

# МЕТОД ФУНКЦИОНАЛЬНОГО ПРЕДСТАВЛЕНИЯ КАК СРЕДСТВО СОЗДАНИЯ ИЕРАРХИИ ВЗАИМОСВЯЗАННЫХ МАКРОМОДЕЛЕЙ В ИНТЕЛЛЕКТУАЛЬНЫХ САПР

Григорьев А.В.

Кафедра ПМИИ, ДонНТУ  
grigorie@r5.dgtu.donetsk.ua

## **Abstract**

*Grigoriev A.V. Method of functional conception as hierarchy creation means associated macromodels in intellectual CAD. In article consider the expansion questions of being conceptuals model of object domain for intellectual CAD. In model bring the consolidates relations. Describes a method modification of functional conception. A Method provides system construction of associated functional levels of diverse complication.*

## **Введение**

Общепринятым подходом в искусственном интеллекте является создание инструментальных оболочек (ИО) для создания экспертных систем на основе уже зарекомендовавших себя экспертных систем (ЭС) [1]. Т.о., принципы, заложенные при построении конкретной проблемно-ориентированной ЭС, применяются уже для решения более широкого класса задач. Совокупность принципов построения ЭС, отвечающих на вопросы «что ..» и «как ..», составляют положения концептуальной модели предметной области (далее просто КМ) как ЭС, так и ИО. Ранее автором разработаны две интеллектуальные системы в области проектирования вычислительных средств (ВС): специализированная ЭС - САПР EMULAT и построенная на ее основе ИО ИнтерНАМ [2]. Соответствующая КМ EMULAT была ориентирована на секционные микропроцессорные комплекты (СМПК), а ИнтерНАМ - на обычную элементную базу, т.е. ИС малой и средней интеграции. Главным достоинством данных систем было наличие функционального представления (ФП) – метода, позволяющего надстраивать над моделью аппаратуры ряда модельных уровней абстракции, например, микропрограмм, системы команд процессора. Анализ теории построения современных САПР ВС, включая концепцию виртуальных машин, КМ современных САПР РЭА, аппаратного интерфейса JTAG для тестирования ПЛИС позволяет судить, что метод ФП *актуален* для современной элементной базы и имеет смысл и в условиях современных САПР. Исходя из этого, автором была поставлена задача разработки новой ИО, наследующей принципы ИнтерНАМ, так называемой мета-эвристической оболочки (МЭО), для построения интеллектуальных САПР, но ориентированной на различные предметные области, в том числе - и на современную элементную базу для построения

ВС, т.е. ПЛИС. Соответственно, «старая» КМ должна быть модифицирована под новую задачу, с сохранением и расширением ее возможностей. Главные недостатки «старой» КМ: 1) отсутствие общей теории построения ФП, т.е. общих путей решения задачи построения ФП; 2) отсутствие общей теории построения взаимосвязанных уровней абстракции, связанные с переходом на новые уровни представлений, со скачками в точности представления моделей. Унифицированная (У) КМ [3] должна обеспечить: представление ВС на основе как «классической», так и новейшей элементной базы, представление объектов не только ВС, но и других предметных областей; наличие средств поддержки ФП. Ранее автором был выполнен ряд работ по построению такой У КМ [4-7]. Недостатком данной У КМ являлось отсутствие определений отношений агрегации, что по существу означало отсутствие реализации метода ФП.

*Цель данной работы* - определить отношения агрегации в У КМ, что фактически означает: ввести массивы и типы в У КМ; обеспечить функциональную многоуровневость модели объекта проектирования средствами, имеющимися в У КМ и на основе метода ФП. При этом отношения агрегации, описывающие массивы и типы в У КМ, должны соответствовать типичным подходам, принятым в САПР. Необходимо определить роль ФП в современных условиях, т.е. доказать: 1) Актуальность применения метода ФП в современных условиях путем анализа текущего состояния и тенденций развития КМ современных САПР РЭА; 2) Возможность и эффективность применения метода ФП при моделировании (проектировании) ВС на основе современной элементной базы построения РЭА в условиях современных САПР РЭА. Вышесказанное означает, что необходимо выполнить следующие задачи: выполнить детальный анализ принципов построения «старой» КМ; сделать детальный анализ «новой» У КМ; предложить решения пополнения «новой» КМ средствами и методами «старой» КМ в новой интерпретации.

## **1. Проблема преемственности моделей различных уровней абстракции в САПР**

Рассмотрим детальнее проблему построения общей теории взаимосвязанных уровней абстракции, отражающих переходы на новые уровни представлений, скачки в точности представления моделей.

### **1.1. Уровни абстракции и формы представления моделей**

Как известно, САПР сложных объектов представляет собой цепочку процедур, предназначенных для последовательного взаимного отображения моделей **системы возрастающих уровней абстракции** и выполняемых под воздействием технического задания. Различают уровни моделей: структурные, системные, функционально-логические, а так же - количественные, включая макро и микро-подуровни [8,9]. Независимо от уровня абстракции, наличие или отсутствие в модели объекта отображения реальной структуры системы позволяет в САПР различать две формы

построения моделей - **микро и макромодел**и [8,9]. Фактически данные формы определяют как подуровни различных абстрактных уровней, так и собственно уровни абстракции. **Макромодель** - это грубая функциональная модель, задающая алгоритм работы объекта (блока) и оперирующая только внешними реальными данными системы. «Грубость» макромодел

и состоит в том, что в ней отсутствует привязка структуры алгоритма к реальной внутренней структуре объекта и внутренним переменным, являющимися характеристиками подсистем. В структуре алгоритма макромодел

и имеются блоки, но не модельно-структурные, а чисто алгоритмические блоки. **Микромодель** – это точная функциональная модель объекта, включающая модель реальной структуры объекта. В зависимости от способа задания модели структуры микромодель может иметь две формы. В первом случае, назовем его *структурной микромоделью*, модель структуры микромодел

и иерархическая, с некоторым числом уровней иерархии более одного. Все базовые подблоки структурной модели объекта, входящие в структуру алгоритма, имеют свои функциональные модели, заданные как *макромодел*и. Т.о., рассматривается: 1) модель структуры как совокупность блоков, а так же определенных над ними структурных, иерархических, множественных и прочих отношений; 2) совокупность функциональных макромодел

и базовых блоков, фактически задающих причинно-следственные отношения над внешними полюсами базовых подблоков. Во втором случае микромодель не имеет четкого деления на модель структуры и функциональные модели базовых блоков, т.е. является смешанной. Модель структуры в этом случае - усеченная, одноуровневая, а модель функций может быть задана просто как совокупность причинно-следственных отношений над множеством свойств-потенциалов, характеризующих некоторые реальных подсистем объекта. Причинно-следственные отношения в этом случае выполняют и роль структурных отношений. Т.о., модель структуры и алгоритм работы базовых блоков задается агрегировано. В зависимости от детальности описания возможно наличие прочих отношений структурной модели (принадлежности к множествам, территориальных и т.д.). Назовем такую микромодель *алгоритмическая микромодель*. Функциональной моделью базового блока может быть макромодель или алгоритмическая микромодель.

### **1.3. Макро и микромодел**и – как уровни адекватности модели

Покажем, что макро- и микромодел

и фактически соответствуют различным уровням достоверности модели. Достоверность или адекватность модели реальной системе определяется такими уровнями достоверности модели [8,9]:

- калиброванная модель: параметры модели установлены в соответствии с имеющимися реальными данными о системе;

- обоснованная: поведение калиброванной модели схоже с поведением реальной системы на уровне ее входов/выходов;
- реалистичная: поведение обоснованной модели схоже с поведением реальной системы на уровне ее элементов, внутренних связей и отдельных подсистем.

Все модели, применяемые в САПР, имеют уровень адекватности выше первой, т.е. калиброванной модели. Макромодель соответствует 2-му уровню адекватности – *обоснованная модель*. Микромодель или полная модель соответствует 3-му уровню адекватности – *реалистичная модель*.

#### **1.4. Проблема преемственности моделей**

При декомпозиции модели объекта, предпринимаемой с целью увеличения уровня адекватности модели, возникает **иерархия «погибших»** базовых функциональных моделей, настроенных над текущей микромоделью объекта. Однако, имеется необходимость обеспечить совместное функционирование «погибших» моделей и текущей микромодели, т.к. подобный подход позволяет получить массу дополнительных преимуществ, делая процесс моделирования и отладки моделей более гибким, обозримым, понятным пользователю. Т.о., возникает проблема восстановления системы функциональных моделей. В этом случае «погибшая» макромодель может быть эмулирована некоторой моделью, являющейся агрегацией базовой микромодели. Известно старое решение данной проблемы, полученное в рамках интеллектуального САПР вычислительных средств на базе СМПК серии АМ2900, называемый **метод функционального представления**.

#### **2. Особенности реализации функционального представления**

Ранее автором были разработаны две программные системы, ориентированные на проектирование ВС [2] и поддерживающие ФП:

- 1) Проблемно-ориентированный интеллектуальный САПР EMULAT, предназначенный для проектирования ВС на базе секционных микропроцессоров серии Am2900 (K1804);
- 2) Инструментальная оболочка ИнтерНаМ, предназначенная для создания интеллектуальных САПР ВС типа САПР EMULAT, но ориентированных на произвольную элементную базу, т.е. БИС малой и средней интеграции.

*Функциональное представление в ИнтерНаМ и EMULAT* – это совокупность средств, предназначенных для решения **трех задач**:

- 1) Прямого задания алгоритмов управления объектом на различных уровнях функционирования как совокупности команд, изменяющих состояние той или иной совокупности элементов памяти в модели объекта.
- 2) Совместной отладки алгоритмов различных уровней иерархии.
- 3) Контроля функционирования алгоритма, способ визуализации функционирования объекта на различных уровнях представления алгоритмов управления.

Перечислим общие возможности САПР EMULAT и ИО ИнтерНаМ [3]. В частности, интеллектуальный САПР EMULAT позволял:

- синтезировать модель структуры ВС и определить тем самым структуру микрокоманды;
- обеспечить прямой ввод микропрограмм в шестнадцатиричной форме представления и косвенный - через интерфейс с системой автоматизации программирования типа AMDASM;
- определять состав и общую структуру команд системы команд вычислительного устройства как специализированного процессора;
- обеспечить прямой ввод программ в системе команд процессора ВС в шестнадцатиричной форме представления и косвенный - через интерфейс с системой автоматизации программирования типа AMDASM;
- обеспечивать совместную отладку моделей ВС как на уровне микропрограмм так и на уровне системы команд ВС.

Условия проверки окончания этапов функционирования микромоделей строились с учетом регулярности функционирования микромоделей, т.е. наличия задающего генератора тактовой частоты. Это можно отнести к специфике метода ФП, принятого в EMULAT. ИнтерНаМ не мог создавать систему макромоделей, а только - одну макромоделю, настроенную над микромоделью, но без агрегации данных. Однако достоинство ИнтерНаМ - это создание библиотеки ФП.

*Достоинства «старого» метода ФП:*

- позволял сохранить модельный уровень, соответствующий некоторому уровню представлений о системе моделирования заданной сложности, в итоговой, самой сложной, адекватной модели; дать возможность пользователю контролировать, анализировать поведение объекта именно на этом уровне представлений;
- возможность визуализации функционирования точной модели на некотором более грубом уровне абстракции;
- начальные значения данных, т.е. условия функционирования системы, могут быть заданы на «старом» уровне абстракции, а – отслеживание функционирования данной модели может выполняться на точной, адекватной модели нижнего уровня.

**Недостатки «старой» реализации ФП:** 1) Необходимость использования больших вычислительных мощностей; 2) Ориентация либо только на СМПК К1804 (EMULAT) при наличии трех уровней ФП, либо наличие только одного макро-уровня ФП в ИнтерНаМ при ориентации на БИС малой и средней интеграции.

**Главные недостатки «старой» КМ относительно ФП:**

- отсутствие общей теории построения ФП, т.е. системы определений, общих путей решения задачи и т.д.;
- отсутствие общей теории взаимосвязанных уровней абстракции, связанные с переходом на новые уровни представления моделей, скачками

в точности представления моделей.

Вывод: актуально и необходимо построить новый метод ФП, лишенный недостатков «старого» ФП.

### ***3. Анализ актуальности метода ФП с точки зрения тенденций развития КМ современных САПР РЭА***

Метод ФП в настоящее время можно явно сравнить с концепцией виртуальных машин, отыскать его положения в КМ современных САПР РЭА, а так же в JTAG - аппаратном встроенном интерфейсе тестирования для ПЛИС.

#### ***3.1. Концепция виртуальных машин как отражение положений метода функционального представления***

Комплекс ФП, имеющийся в ИнтерНАМ, может трактоваться как система альтернативных виртуальных машин (ВМ), надстроенных над одной реальной. Существуют средства аппаратной [10] и программной [11] реализация виртуального компьютера как средства взаимосвязи физического компьютера и виртуального, функционирующего в его среде; Аппаратная реализация включает карты: идентификации виртуальных устройств, виртуальных процессов и т.д. Известно, по крайней мере, две программные реализации концепции ВМ: в ОС Linux [10,11] и - в программных пакетах от компании Microsoft. ВМ может трактоваться как средство кроссового построения программного обеспечения на другой ЭВМ и на другой программной платформе, т.е. задача проектирования РЭА. Покажем, что ВМ – это явно выраженный аналог метода ФП. ВМ ОС Linux — специальная программа, запускаемая в ОС Linux и моделирующая физический компьютер Intel x86. ВМ «строится» из набора виртуальных устройств: виртуальные жесткие диски IDE и т.д. Набор виртуальных устройств не зависит от набора реального компьютера, на котором запускается виртуальная машина. Построение набора аппаратуры ВМ наиболее наглядно происходит при использовании «редактора конфигурации». Если операционная система устанавливается непосредственно внутри ВМ, то в процессе установки все эти устройства определяются корректно. Компания Microsoft открыла свободный доступ к программному продукту Virtual PC 2007, являющийся ответом на ВМ ОС Linux. ВМ Microsoft позволяет запускать виртуальную копию одной или нескольких различных операционных систем одновременно на одном и том же физическом компьютере и при этом позволяет симулировать нужное аппаратное обеспечение. Ориентирована на тестировщиков программного обеспечения перед внедрением продуктов на реальные системы. Существует концепция т.н. "структур" как средства взаимосвязи для различных когнитивных уровней представления модели объекта [12,13], являющаяся вариантом ВК для когнитивных моделей. Сделаем сравнительный анализ концепции виртуального компьютера и метода ФП:

- 1) Базовый метод ФП жестко ориентирован как средство моделирования (проектирования) средств вычислительной техники, т.е. на класс задач проектирования РЭА, включая кроссовую разработку программного обеспечения.
- 2) Концепция ВМ ориентирована только на кроссовое построение программного обеспечения на другой ЭВМ и на другой программной платформе, т.е. - на одну из задач проектирования РЭА;
- 3) ВМ частично отражает метод ФП, ФП ориентирован на более глубокое отражение идеи ВМ и более широкий класс задач;
- 4) Факт появления ВМ говорит об актуальности построения общей теории ФП, способной как обобщить идею ВМ, так и обеспечить приложение данного подхода к другим предметным областям.

### ***3.2. Отражение положений унифицированной концептуальной модели в современных САПР РЭА***

Магистральный путь современного развития САПР РЭА – это формирование моделей структур и функций объектов как текстов на некоторых модификациях широко распространенных алгоритмических языков. Можно назвать языки VHDL (аналог Бейсик), Verilog (аналог С), AHDL и прочие [14]. Сделаем анализ типичного языка проектирования на примере VHDL. Особенности элементной базы VHDL показаны в табл. 1. Сравнительный анализ особенностей КМ VHDL и систем проектирования ИнтерНАМ и EMULAT показаны в табл. 2.

Табл. 1. Сравнительный анализ предметных областей САПР VHDL и ФП.

<b>Особенности предметной области языка проектирования аппаратуры VHDL</b>	<b>Особенности предметной области базового метода ФП</b>
ПЛИС ориентированы большей частью на агрегированное представление логики, выполняемой ранее на ИС малой и средней интеграции, необходимые для помощи процессору.	Ориентация либо только на СМПК К1804 (EMULAT) при наличии трех уровней ФП, либо наличие только одного макро-уровня ФП в ИнтерНАМ при ориентации на БИС малой и средней интеграции.

Общий вывод – КМ VHDL в целом соответствует КМ названных систем. Частные выводы из сравнительного анализа:

- 1) Развитие языков проектирования аппаратуры идет в направлении появления средств и методов, аналогичных методу ФП;
- 2) Метод ФП **актуален** как перспективный путь развития средств и методов языков проектирования аппаратуры на системном и ФЛ-м уровне.
- 3) Неизбежным представляется появление в VHDL иерархии декомпозиции структур данных.

Рассмотрим постановку задачи построения ФП в базовом языке VHDL. VHDL явно рассматривает множество микромоделей, заданных

совокупностью архитектур.

Табл. 2. Сравнительный анализ КМ САПР VHDL и У КМ

<b>Особенности КМ САПР VHDL</b>	<b>Комментарии к У КМ</b>
Имеется тип данных - синхросигнал, т.е. рассматривается регулярный алгоритм с облегченным выделением этапов функционирования объекта.	Базовый метод ФП ориентирован на регулярные функции.
Наличие в VHDL явного деления моделей на два типа - структурной и поведенческой модели.	Поведенческая модель соответствует либо макромодели либо структурной микромодели.
Наличие искусственных методик декомпозиции проекта на подчасти, имеющие более оптимальные пути физической реализации. Является недостатком структурной модели объекта, принятой в VHDL.	При необходимости данная структурная модель должна включать элементы топологической модели, как это принято в предлагаемой КМ.
Имеется механизм типов и подтипов как подмножеств значений исходного дискретного типа.	Данный механизм есть зачаточный механизм декомпозиции данных.
Имеются средства условной трассировки состояния модели.	Нет возможности объединения структур данных и агрегации условий.

Микромодель VHDL - это:

- некоторая сущность, определенная названием, входами-выходами (портами) и рядом (классом) возможных именованных архитектур;
- каждая архитектура есть описание моделей структур на основе других сущностей (как вариант – подфункций);
- на нижнем уровне архитектуры – сущности, связанные с рядом возможных архитектур, заданных как модели поведения.

Т.о., возможно создавать на верхнем уровне архитектуру как модель поведения, т.е. макромодель, а затем уже – разбивать ее на микромодель. Данная ситуация не достаточно четко прописана в языке VHDL, но это - прямая возможность создавать ФП, эмулирующую исходную макромодель. Т.о. ФП в явном виде отсутствует, хотя в наличии – явные предпосылки для его построения. Ответим на вопрос: можно ли, пользуясь существующими средствами языка, что бы эмулировать хотя бы в простом варианте метод ФП. Т.е. реализовать: средства потери точности (декомпозиция данных); отражение вышележащей макромодели – через нижележащую микромодель.

Рассмотрим средства и способ возможного решения:

1) *Средства*: типы и подтипы данных; механизм условной трассировки; механизм декомпозиции сущностей на различные архитектуры; задание



архитектур либо как модели поведения, либо как модели структуры.

2) *Предлагаемый способ реализации ФП*: типы и подтипы есть средство агрегации (декомпозиции) данных, обеспечивающие построение системы моделей с потерей точности; механизм условной трассировки есть простейшее средство условной визуализации данных в методе ФП; совокупность механизма сущностей, как совокупности входов-выходов, архитектур, деления моделей на модели поведения и модели структур есть средство построения единичной микромоделей.

*Общий вывод*: ФП реализовать можно, но в очень урезанном виде.

### **3.3. Сравнительный анализ интерфейса JTAG и ФП**

Интерфейс JTAG есть средство создания встраиваемых инструментов в ПЛИС для физического доступа тестирующей аппаратуры к плате. К возможностям интерфейса относятся: фиксация значений контролируемых контактов в требуемые моменты времени с возможностью контроля их проектировщиком; наличие языка для управления процессом тестирования аппаратуры внутри ПЛИС (внутрисхемное программирование) и т.д. Языки, включающие интерфейс JTAG (HSDL) позволяют: создавать определяемые пользователем имена для описания команд сканирования и групп данных; объединять последовательности битов с назначаемыми пользователем именами и т.д. Можно провести такие параллели с ФП: 1) просматриваются средства агрегации данных; 2) предусматриваются условия (когда) и места съема информации (что) для тестирования.

## **4. Общая постановка задачи построения «эмуляции» погибших макромоделей в рамках нового ФП**

Т.о., ФП в простейшем случае рассматривает:

1) На входе:

– «старая» функциональная модель блока, рассматриваемого как базовый, т.е. либо - грубая макро модель блока, либо - более точная алгоритмическая микро модель блока;

- «новая» функциональная блока, заменившая собой старую и заданная как более точная структурная микро модель, имеющая ряд структурных уровней и с новые макро модели базовых блоков;

2) На выходе – эмуляция «старой» функциональной модели, алгоритм которой задан неявно и отражает алгоритм функционирования ниже лежащей микро модели или – более точной макро модели.

Пользователь должен отразить этапы «старого» алгоритма в этапы функционирования ниже лежащей модели. Имеется информация:

1) По «старому» алгоритму: состав и идентификация этапов старой макро модели; состав и идентификация данных блоков старого алгоритма; причинно-следственные отношения между блоками.

2) По «новому» алгоритму: состав и идентификация блоков базовых макро моделей; состав и идентификация данных каждого блока нового

алгоритма; причинно-следственные отношения между блоками.

Главная задача – синхронизировать окончание некоторого этапа «старого» алгоритма с конкретной ситуацией, сложившейся в процессе функционирования нижележащего «нового» алгоритма.

## **5. Модификация метода ФП**

В работах [3-7] изложены основные принципы построения новой, унифицированной КМ, являющейся модификацией и расширением КМ указанных старых систем проектирования. Рассмотрим ее основные положения и методы и средства реализации метода ФП.

### **5.1. Состав новых средств У КМ:**

1) Построения модели объекта: система построения иерархии моделей путем декомпозиции на основе НЕ-факторов [15,6] и физической семантики ПрОб [7]; принцип доступной когнитивной сложности моделей [16];

2) Построения модели САПР:

- теоретико-множественные операции (ТМО), определенные над контекстно-свободными грамматиками, эквивалентными И-ИЛИ-дереву, с дополнительными явными или неявными продукционными зависимостями, задающими семантику языка [17];
- модель САПР как комплекс двух грамматик – пространства решений (ЦПС) и технических заданий (ПОС) [18];
- система макро-продукций и методов организации вывода над макро-продукциями [19].

### **5.2. Развитие, модификация ФП на основе средств У КМ**

Предлагается следующее обобщение метода ФП:

1) Проверять не одно ожидаемое событие, а ряд событий, идущих во времени друг за другом (различные этапы жизненного цикла (ЖЦ)), либо одновременно (в одном этапе). Событие связывается отношениями типа «возможно» (в будущем) и «необходимо» (одновременно).

2) Проверка событий выполняется путем выполнения ТМО над текстами модели. При этом:

- для сокращения объема текста, подвергаемого ТМО, выполнять проверку в пределах одного подблока, идентифицируемого по вертикали вложенных блоков (доступной когнитивной сложности);
- чтобы убрать несущественные детали из условия, вводится неопределенность (НЕ-факторы), которая касается как необходимой части текста (известной, определенной), так и – не нужной части текста (неизвестной, неопределенной); т.е. предполагается использование неопределенностей в ТМО текстов.

3) Рассматривать ЖЦ объектов для проверки условий, построенных различным способом, т.е. как:

- обобщение ряда текстов существующих ЖЦ, с привлечением процедуры П4 семиотической модели как средства вывода [20];
  - ЖЦ, введенный экспертным путем, т.е. функциями базовых блоков, с привлечением временной логики как средства вывода.
- 4) Рассматривать функции базовых структурных блоков, построенные различным способом, т.е. как:
- обобщение ряда границ данного блока в существующих ЖЦ, т.е. использовать табличный путь построения функций [21];
  - функции, построенные экспертным путем, т.е. использовать механизм программирования в ограничениях.

## **6. Методы и средства обеспечения функциональной многоуровневости в среде У КМ**

Задача обеспечения функциональной многоуровневости в среде У КМ распадается на следующий ряд подзадач:

- введение типичных отношений агрегации в микромодели;
- построение модельных структурных компонент вышележащей модели по структурным элементам нижележащей модели;
- определение условий визуализации состояний макромодели в моменты окончания функционального блока некоторого типа.

Рассмотрим способы решения данных задач.

### **6.1. Введение отношений агрегации в микромодели У КМ**

Состав отношений, определенных в среде У КМ, включает в себя, согласно принятой автором классификации, отношения: принадлежности; передачи информации, обработки информации, декомпозиции, агрегации, порядка [3,6,7]. Ранее были определены: отношения декомпозиции и порядка; отношения передачи информации, обработки информации; отношения принадлежности. Задача данной работы – *определить недостающие отношения агрегации, включая отношения ФП*. Рассмотрим специфику отношений декомпозиции и порядка на примере множества свойств и значений [6]. Декомпозиция структур в данной модели выполняется путем расширения состава свойств блоков за счет декомпозиции «черных», т.е. явно введенных неопределенных свойств (и блоков), а так же путем переопределения значений свойств как свойств с новыми значениями. Т.о., механизм декомпозиции предполагает *скачкообразное увеличение точности модели структур* в процессе декомпозиции.

**Предлагаемое решение:** строить структурные отношения агрегации путем, обратным декомпозиции структурной модели блоков с увеличением точности, т.е. – с потерей точности. Рассмотрим одноуровневую микромодель некоторого уровня с отношениями агрегации первого типа (массивы, типы). Информация о декомпозиции - отсутствует. Построим вышележащую макромодель, используя отношения агрегации. При этом:

- агрегация и декомпозиция строятся совершенно аналогично и включают

тот же набор средств, отличаясь в порядке их применения;

- декомпозиция с увеличением точности означает появление новых данных и блоков, не связанных наследованием с вышележащей моделью;
- агрегация с потерей точности означает - агрегировать не все внешние данные, характеризующие блок на текущем уровне абстракции, а только самые важные; не вошедшие - есть потеря точности;
- при декомпозиции идентификация блока сохраняется, а данные - детализируются и расширяются;
- построить блок агрегацией означает определить идентификацию блока и его внешние данные.

### **6.2. Определение отношений агрегации как механизма создания структурных блоков и свойств вышележащей макромоделей**

Для восстановления макромоделей язык описания микромоделей, должен отражать состояние модели объекта во времени, т.е. состояние всех данных (значение свойств). На уровне ФЛМ это может быть модель синхронного или асинхронного типа. Агрегацией создается язык описания новой макромоделей того или иного уровня. Семантика языка описания макромоделей соответствует структурной модели вида и формы, аналогичной микромоделей. Этапы процесса восстановления: 1) идентификация блока алгоритма, его типа и принадлежности к некоторому массиву некоторого размера, номера в массиве; 2) определение состава входных и выходных данных блока по формату – тип, массив, размер массива, номер в массиве; 3) определение связей между блоками по формату – «данное источник -> данное приемник». Т.о., причинно-следственные связи считаются известными, однако их инициализация не отслеживается в процессе моделирования. Отслеживается только конец срабатывания блока по факту инициирования причинно-следственной связи. Возможные результаты срабатывания, исходя из типа микромоделей: 1) при асинхронном моделировании – запланированное срабатывание выхода при измененном ранее входе; 2) при синхронном срабатывании – конец всех процессов изменения на момент окончания некоторого такта синхросигнала. Отслеживание срабатывания блока предполагает идентификацию блока и состояний его входов-выходов. Т.о., блок фактически рассматривается как совокупность данных. Каждое данное есть пара: "агрегируемые данные из нижележащей модели" + "условие". На первом уровне ФП агрегации может подвергаться состояние микромоделей. Потеря точности – принципиальная черта ФП, т.к. на каждом уровне мы теряем данные, не нужные для пользователя. В макромоделей нижнего уровня теряем данные из микромоделей, в небазовой макромоделей – данные из нижележащей макромоделей. Т.о., ФП есть «агрегация структур с потерей точности» плюс «условия синхронизации».

### **6.3. Порядок построения отношений агрегации**

Рассмотрим механизм реализации агрегации с потерей точности.

Классическая агрегация предназначена для создания одномерных, двумерных или многомерных списков элементов в структурной модели. В нашем случае есть две области формирования отношений агрегации:

1) В **микромодели** агрегация, как способ расширения описания структурной модели, исходит из предположения о наличии двух сред блоков - внешней и внутренней [7], что является спецификой У КМ. Агрегация выполняется по таким признакам:

- принадлежности к одному *типу внутренней структуры*, например, типу блока, типу свойства;
- принадлежности к совокупности элементов, выполняющих одну и ту же роль, т.е. имеющих одно и то же *назначение во внешней среде*, например - массив сигналов одного типа и назначения у блока.

2) В **макромодели** агрегация, как способ построения структуры вышележащей макромодели, выполняется либо в пределах микромодели, либо – в пределах нижележащей «эмулированной» макромодели. Выполнение агрегации идет по признакам:

- принадлежности к *данной структурной компоненте* вышележащего уровня, например, вышележащему блоку или свойству;
- принадлежности данной структурной компоненты вышележащей функциональной модели к *типу структурной компоненты* функциональной модели вышележащего уровня;
- принадлежности данного структурного компонента к совокупности элементов, имеющих одно и то же *назначение*.

*Агрегация первого типа* не содержит ничего принципиально нового и решает задачу внесения массивов и типов в микромодель, принятую в У КМ. *Агрегация второго типа* есть средство построения системы макромоделей. Предполагает такие действия:

- определение структурного элемента некоторого типа вышележащей модели как агрегации конкретного фрагмента модели нижележащего уровня;
- определение типа блока или свойства как пример агрегации блоков и свойств нижележащей модели;
- агрегация выделенных структурных элементов некоторого типа в массивы элементов одного назначения.

Агрегация макромодели в вышележащую макромодель требует упрощения алгоритма, т.е. сокращения числа блоков и их данных. *Примечание:* агрегацией строятся только собственные данные блоков. Чужие, т.е. данные влияния (см. [7]) – не рассматриваются.

### **Заключение**

В статье решены вопросы дополнения У КМ отношениями агрегации, цель которых - внести ФП, т.е.: ввод массивов и типов в У КМ; обеспечение функциональной многоуровневости модели объекта проектирования средствами, имеющимися в новой У КМ. В предлагаемой работе выполнен

детальный анализ принципов, на которых была выполнена «старая» КМ, сделан детальный анализ «новой» У КМ и предложен ряд решений, пополняющих «новую» КМ средствами и методами «старой» КМ в новой интерпретации. Показано, что «новая», У КМ обеспечивает: представление ВС на основе как «классических», так и новейшей элементной базы, например ПЛИС; представление объектов проектирования не только ВС, но и других предметных областей. Предложенная модифицированная У КМ применялась на практике для построения САПР паро-газовых установок [22], включая описания систем трубопроводов, что доказывает ее применимость к предметным областям, отличным от проектирования ВС. Как перспективную задачу назовем практическое доказательство эффективности метода ФП в практике современных САПР ВС.

### **Литература**

1. Интеллектуальные системы принятия проектных решений / А.В.Алексеев, А Н Борисов, Э.Р.Вилломс, Н.Н.Слядзь, С.А.Фомин. - Рига: Зинатне, 1997. - 320 с.
2. Реуцкий В.А., Гельман А.Ю., Григорьев А.В. и др. Пакет программ функционально-логического проектирования микропроцессорных систем // Техника средств связи. 1989.- Вып. 6.-С.63-66.
3. Григорьев А.В. Унифицированная концептуальная модель предметной области. В кн. Информатика, кибернетика и вычислительная техника (ИКВТ-97). Сборник трудов ДонГТУ, Выпуск 1. Донецк: ДонГТУ, 1997. С.218-224.
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. Григорьев А.В. Теоретико-множественные операции над грамматиками как механизм работы со знаниями в интеллектуальных САПР. Вісник Східноукраїнського національного університету імені Володимира Даля, N 2(48). Луганск, ВУТУ, 2002. С. 186-194.
6. Григорьев А.В. Представление недоопределенности знаний в инструментальной оболочке для построения САПР. Искусственный интеллект. N 6, 1999 , С. 56-66.
7. Григорьев А.В. Комплекс моделей САПР как система взаимосвязанных уровней о действительности. Научные труды Донецкого государственного университета. Серия «Информатика, кибернетика и вычислительная техника», (ИКВТ-2000) выпуск 10. - Донецк, ДонГТУ, 2000. - С. 155-167.
8. Норенков И.П. Введение в автоматизированное проектирование технических устройств и систем. М.: Высш. шк., 1986. - 304 с.
9. Петренко А.И., Семенов О.И. Основы построения систем автоматизированного проектирования. - К.: ВШ, 1984. - 296 с.

10. Jim, Jr. Smith, Ravi Nair, James E. Smith , *Virtual Machines: Versatile Platforms For Systems And Processes*, Publisher Morgan Kaufmann Publishers, May 2005, 656 p.
11. В.Костромин, «Linux для пользователя» изд. БХВ-Петербург», 2002 г., серия «Самоучитель».
12. David Nunez. *A constructionist cognitive model of virtual presence*. University of Cape Town. 2004.
13. *Orchestrating multi-level knowledge integration in an organization: a social-cognitive perspective*. Chi-Wei Liu. HongKuang University, Kaohsiung, Taiwan. 2004.
14. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – СПб.: БХВ-Петербург, 2002.
15. Нариньяни А.С. Недоопределенность в системах представления и обработки знаний // Изв. АН СССР. Техн. кибернетика, 1986. -№ 5. – С. 3 – 28.
16. Григорьев А.В. Ограничение когнитивной сложности моделей. В кн. *Прогрессивные технологии и системы машиностроения:: Международный сб. научных трудов*. - Донецк: ДонГТУ, 2000. Выпуск. 10 - С. 49-58.
17. Григорьев А.В. Теоретико-множественные операции над грамматиками как механизм работы со знаниями в интеллектуальных САПР. *Вісник Східноукраїнського національного університету імені Володимира Даля*, N 2(48). Луганск, ВУТУ, 2002. С. 186-194.
18. Григорьев А.В. Упорядочивание обликов в семиотической модели САПР. *Искусственный интеллект*, 2005, N 4, 2005.
19. Григорьев А.В. Принципы организации вывода решений в базе знаний инструментальной оболочки для создания интеллектуальных САПР. // *Практика і перспективи розвитку інституційного партнерства*». *Вісник ДонГТУ – ТРТУ*. Донецьк: РВА ДонНТУ, 2003 – С.96-106.
20. Григорьев А.В. Управление движением объектов в семиотической модели предметной области. *Наукові праці національного технічного університету*. Серія «Обчислювальна техніка та автоматизація». Выпуск 48: Донецк: ДонНТУ, 2002. - С.280-287.
21. А.В. Григорьев. Методы построения функций в специализированной оболочке для создания интеллектуальных САПР. *Искусственный интеллект*. N 3, 2001, С. 40-53.
22. Григорьев А.В. Методы и средства работы с графическими моделями в САПР парогазовых установок СПРУТ / *Моделирование и компьютерная графика: Материалы 1-й международной научно-технической конференции*, г Донецк, 04-07 октября 2005 г. — Донецк, ДонНТУ, Министерство образования и науки Украины, 2005. – С. 77-86.