

УДК 681.3.06.14

Л.С. Файнзильберг

Синтез информационных технологий обработки сигналов в задачах косвенного контроля и диагностики

Рассмотрены проблемы синтеза информационных технологий обработки сигналов («интеллектуальных» датчиков и индикаторов) в условиях ограниченной априорной информации. На основе системного и объектно-ориентированного анализа предложена архитектура инструментальной системы для синтеза прикладных ИТ.

Problems of synthesis of information technologies for signal processing («intelligent» sensors and indicators) under conditions of the limited a priori information are considered. An architecture of the tool system for synthesis of application IT is suggested on the basis of system and object-oriented analysis.

Розглянуто проблеми синтезу інформаційних технологій обробки сигналів («інтелектуальних» датчиків та індикаторів) в умовах обмеженої априорної інформації. На основі системного та об'єктно-орієнтованого аналізу запропоновано архітектуру інструментальної системи для синтезу прикладних ІТ.

*Линия жизни не только твоя,
Всех предыдущих твоих поколений,
Ты — как итог этих чудных ветвлений,
В данный момент заслужил бытия.*

*С миром тобою они говорят,
Ты — слабый шанс в этом свете остаться,
Будь же гуманным, сумей постараться,
Не обрывай на себе этот ряд.*

Т.П. Поганова (1948–1997)

Введение. В последнее время информационные технологии (ИТ) получают все большее распространение для решения задач косвенной оценки параметров и диагностики текущего состояния различных объектов, если непосредственное определение этих характеристик невозможно из-за отсутствия соответствующих датчиков и индикаторов [1, 2]. Синтез ИТ для каждой конкретной прикладной задачи основан на интеграции знаний двух специалистов — конструктора ИТ и эксперта в соответствующей предметной области (рис. 1).

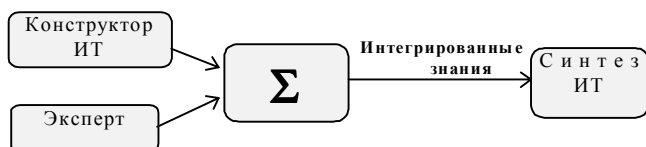


Рис. 1

Опыт показывает, что сам процесс интеграции знаний конструктора и эксперта представляет со-

бой далеко не простую задачу, поскольку профессиональные языки этих специалистов существенно различны. При этом конечного пользователя, который чаще всего и является экспертом в прикладной области, интересуют лишь потребительские свойства разрабатываемой ИТ (достоверность и точность оценки параметров, форма представления информации и т.п.), а конструктора ИТ прежде всего интересует то, как на основе обработки входных данных получить интересующую пользователя информацию.

В тех случаях, когда прикладная задача достаточно хорошо формализована, для интеграции знаний конструктора и эксперта требуется лишь определенное время: в конце концов конструктор получит от эксперта требуемые ему знания в виде алгоритмов, моделей, характеристик входных данных и на основе этих знаний сможет синтезировать ИТ, а затем уже вместе с экспертом оценить ее потребительские свойства.

Однако довольно часто на практике встречаются такие прикладные задачи, когда одной лишь интеграции априорных знаний конструктора и эксперта недостаточно для синтеза ИТ: знания в соответствующей предметной области слабо формализованы, плохо структурированы и противоречивы [3]. В таких случаях конструктор может получить от эксперта лишь некоторые гипотезы о взаимосвязи конечной информации с входными данными, а также экспериментальный материал в виде обучающей выборки наблюдений с указаниями об истинных значениях параметров объекта и его функциональном состоянии.

Синтез ИТ в таком более общем случае представляет собой уже достаточно серьезную проблему. Традиционно разрешение этой проблемы сводится к выполнению этапа исследований, связанного с получением дополнительной информации. При этом осуществляется разработка и реализация ряда альтернативных вычислительных процедур, направленных на анализ и интерпретацию входных данных с последующей сравнительной оценкой качества этих процедур по имеющейся обучающей выборке. И только после того, как будет определена на основании такой сравнительной оценки наилучшая в некотором смысле совокупность вычислительных алгоритмов, можно приступить к синтезу конкретной прикладной ИТ.

Такой путь сопряжен со значительными материальными и временными затратами, что в условиях существующего экономического кризиса препятствует решению ряда актуальных прикладных задач.

На первый взгляд может показаться, что применение какого-либо универсального пакета программ обработки данных позволит существенно сократить эти затраты. Действительно, в известных зарубежных пакетах *ASSYST* фирмы *MacMillan*, *STATGRAF* фирмы *Statistical Research*, *DataEngine* фирмы *Management Intelligenten Technologien GmbH* и многих других не только включен широкий набор программных средств анализа сигналов, написанных на высоком профессиональном уровне, но и реализован современный интерфейс пользователя, обеспечивающий удобство при проведении необходимых исследований.

В то же время большинство таких пакетов закрыты для пользователя в том смысле, что он, по существу, лишен возможности «извлечения» под-

ходящей процедуры обработки сигнала, не говоря уже о возможности ее модифицирования. Другими словами, даже если на этапе экспериментальных исследований с помощью подобных пакетов удастся выбрать оптимальные вычислительные процедуры обработки, то на этапе синтеза ИТ все равно придется писать собственные программы.

При этом перед конструктором ИТ неизбежно возникают многочисленные дополнительные проблемы, вызванные недостаточным пониманием отдельных тонкостей вычислительных процедур, которые обычно скрыты от пользователя. Довольно часто такие непредвиденные проблемы могут оказаться неразрешенными, и попытка синтеза прикладной ИТ собственными силами терпит неудачу.

В настоящей статье развивается альтернативный подход, основанный на синтезе ИТ обработки сигналов с использованием специальной инструментальной системы. Особенность этого подхода состоит в том, что с помощью инструментальной системы конструктор ИТ совместно с экспертом должен иметь возможность не только провести необходимые исследования и сравнительную оценку вычислительных процедур обработки входных данных, но и реализовать оптимальную в некотором смысле технологическую цепочку таких процедур из готовых и доступных ему компонентов самой инструментальной среды и тем самым синтезировать прикладную ИТ обработки сигналов.

Постановка задачи. Следуя [1], будем рассматривать две задачи синтеза информационных технологий обработки сигналов.

Задача 1. Пусть значение N -мерного вектора $z^{(N)} = (z_1, \dots, z_N)$ отражает некоторую количественную характеристику текущего состояния объекта исследования, интересующую пользователя, однако все или какие-либо компоненты этого вектора не могут быть непосредственно измерены ввиду отсутствия соответствующих датчиков. Пусть далее в процессе своего функционирования объект порождает на своем выходе сигнал $x = x(t)$, который может быть зарегистрирован на интервале $t \in [0, T]$. Предполагается, что этот сигнал $x = x(t)$ несет информацию о неизвестных компонентах вектора $z^{(N)}$.

В этом случае под синтезом ИТ будем понимать выбор на основе имеющейся априорной информации совокупности вычислительных проце-

дур, позволяющих путем обработки конкретной реализации сигнала $x = x(t)$ косвенно оценивать значение вектора $z^{(N)}$. При этом точность такой оценки характеризует некоторая мера отклонения от истинных значений вектора $z^{(N)}$, известных для контрольной выборки наблюдений на этапе синтеза ИТ.

Задача 2. Пусть *качественная* характеристика текущего состояния объекта определяется его принадлежностью к одному из M классов конечного множества состояний $W = \{W_1, \dots, W_M\}$ и считается, что наблюдаемый на интервале $t \in [0, T]$ сигнал $x = x(t)$, который порождается этим объектом, несет информацию о текущем состоянии объекта.

В этом случае под синтезом ИТ будем понимать выбор на основе имеющейся априорной информации совокупности вычислительных процедур, позволяющих путем обработки конкретной реализации $x = x(t)$ принимать решение о текущем состоянии объекта W_d ($1 \leq d \leq M$). При этом достоверность принимаемых решений характеризует частота ошибок, допускаемых при обработке контрольной выборки наблюдений $x(t)$ с известной принадлежностью к классам состояний объекта.

Предлагаемый подход. Таким образом, решение сформулированных задач синтеза ИТ сводится к оптимальному в некотором смысле преобразованию исходного сырья (реализации сигнала $x(t)$) в конечный продукт (оценку вектора $z^{(N)}$ и/или решение о текущем состоянии объекта W_d).

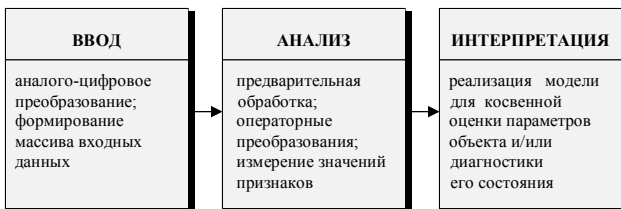


Рис. 2

Естественно, при решении конкретной прикладной задачи такое преобразование должно осуществляться с учетом специфики решаемой задачи. Вместе с тем, на основе системного анализа существующих ИТ можно выделить общие этапы такого преобразования, присущие практически любой ИТ обработки сигналов (рис. 2):

- Ввод наблюдаемого сигнала $x(t)$;

- Анализ сигнала, включающий его предварительную обработку, преобразование и измерение характерных признаков;
- Интерпретация сигнала в соответствии с математической моделью, в некотором смысле оптимальным образом связывающей $z^{(N)}$ и W_d с измеренными значениями признаков наблюдаемой реализации $x(t)$.

Какие же знания нужны для реализации этих этапов при синтезе конкретной информационной технологии?

Для организации ввода $x(t)$ конструктору достаточно получить сведения об уровне и диапазоне возможных значений обрабатываемого сигнала, выбрать интервал наблюдения и на основании частотных характеристик сигнала определить оптимальный шаг квантования по времени. Будем считать, что такие знания имеются, и тогда конструктору остается лишь выбрать параметры средств организации ввода (параметры нормирующего преобразователя, разрядность и быстродействие АЦП и т.п.) или при отсутствии стандартных — сформировать техническое задание на их разработку. Разумеется, на этом этапе возникает целый ряд других вопросов, обсуждение которых выходит за рамки настоящей статьи.

Более серьезные проблемы возникают при реализации второго и третьего этапов. Поэтому остановимся на них подробнее.

Будем считать, что реализация входного скалярного сигнала $x(t)$ представлена в виде временного ряда — конечного массива дискретных значений x_1, \dots, x_K , зафиксированных с некоторым постоянным шагом квантования в дискретные моменты времени $1, \dots, K$. Естественно, что именно в векторе $x^{(K)} = (x_1, \dots, x_K)$ сосредоточена вся доступная нам информация об исследуемом объекте. Отсюда может показаться, что проблема синтеза ИТ сводится всего лишь к идентификации математической модели, устанавливающей зависимость компонент векторов $z^{(N)}$ со значениями векторов $x^{(K)}$ в виде

$$z_j = F_j(x^{(K)}, \theta_j), j = 1, \dots, N, \quad (1)$$

т.е. выбору на основе имеющейся обучающей выборки для каждого j наилучших в некотором смысле структур F_j и определению оптимальных значений вектора параметров $\theta_j = (\theta_{j1}, \dots, \theta_{jL})$ по

алгоритмам, известным из теории распознавания образов и восстановления зависимостей по экспериментальным данным [4–6]. Однако в общем случае такой подход не является правомерным, причем дело даже не только в том, что множество всевозможных структур F_j бесконечно. Даже если попытаться каким-либо способом ограничить это множество, например, рассматривать только линейные по параметру θ_j модели (линейные регрессии)

$$z_j = \theta_j^T G_j(x^{(k)}), \quad j = 1, \dots, N, \quad (2)$$

где $G_j(x^{(k)}) = (g_j^{(1)}(x^{(k)}), \dots, g_j^{(K)}(x^{(k)}))$ — вектор-функция, T — знак транспонирования, а в качестве базисных функций $g_j^{(1)}(x^{(k)}), \dots, g_j^{(K)}(x^{(k)})$ выбрать полиномиальные функции и провести структурную и параметрическую идентификацию на основе известных комбинаторных [7] или итерационных [8] алгоритмов, то и в этом случае непосредственное использование дискретных отсчетов сигнала x_1, \dots, x_K в качестве аргументов таких моделей в ряде случаев может не привести к желаемому результату.

Поясним сказанное на следующем примере. Пусть физическая природа объекта такова, что значение его параметра z_q связано линейной зависимостью только с одним из значений наблюдаемого сигнала $x(t)$ в характерной точке, например с максимальным значением x_{\max} , т.е.

$$z_q = c_0 + c_1 x_{\max}, \quad (3)$$

где c_0, c_1 — некоторые константы, однако сам момент времени t_{\max} , когда $x = x_{\max}$, зависит от значения z_q , т.е. $t_{\max} = \Psi(z_q)$, где Ψ — неизвестная функция. Совершенно очевидно, что без такой дополнительной априорной информации невозможно при сколь угодно большом объеме обучающей выборки восстановить истинную зависимость (3), опираясь на полиномиальные модели (2): для различных реализаций из обучающей выборки аргумент истинной зависимости (3) — максимальное значение сигнала x_{\max} — будет соответствовать различным аргументам модели (2), т.е. различным номерам дискретных отсчетов x_1, \dots, x_K . К тому же при больших K возникают непреодолимые вычислительные проблемы в использовании известных алгоритмов структурной идентификации моделей (2) [7, 8].

Именно поэтому важным этапом синтеза прикладной ИТ является выбор методов перехода от пространства наблюдений $x^{(k)}$ к пространству признаков y_1, \dots, y_R , которые разумно использовать в качестве аргументов интерпретирующих моделей, причем число R таких признаков должно быть существенно меньше числа K .

В ряде случаев интеграция априорных знаний конструктора и эксперта достаточна для решения этой задачи: эксперт знает, какие характеристики сигнала потенциально могут быть использованы в качестве аргументов модели, а конструктор знает, как эти характеристики могут быть вычислены на основе обработки исходных данных. Однако на практике такие полные априорные знания являются скорее редким исключением, чем правилом. Обычно синтез ИТ осуществляется в условиях неполных или противоречивых знаний о том, какие именно характеристики сигнала следует использовать в качестве признаков y_1, \dots, y_R .

В этом общем случае, по-видимому, не остается ничего иного, как методом проб и ошибок провести сравнительную оценку различных методов перехода от реализаций $x^{(k)}$ к признакам $y^{(R)} = (y_1, \dots, y_R)$ по имеющимся экспериментальным данным. Поэтому для синтеза ИТ предлагается следующая обобщенная архитектура инструментальной системы (рис. 3).

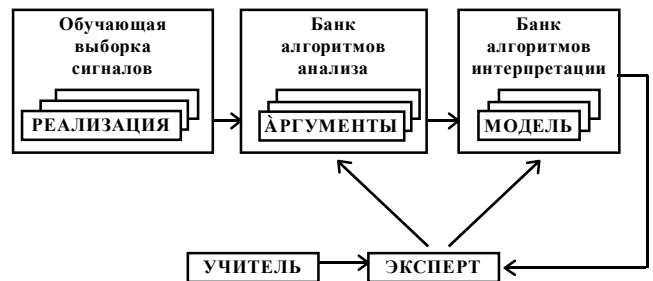


Рис. 3

В данной архитектуре под указаниями учителя понимаются сведения об истинных значениях оцениваемого параметра z_j и/или классе состояния объекта W_d для каждой из реализаций сигнала в обучающей выборке, а под указаниями эксперта — экспертная оценка или же некоторый формальный критерий качества продукта ИТ.

Возникает естественный вопрос: можно ли реализовать такую универсальную архитектуру инструментальной системы для синтеза прикладных

ИТ обработки сигналов, включив в банки алгоритмов анализа и интерпретации все известные и апробированные на практике методы? По-видимому, в настоящее время это практически нереализуемо, поскольку таких алгоритмов огромное множество. В то же время, рассматривая эту цель как «дальнюю», можно в качестве «ближней» цели поставить задачу построения инструментальной системы, которая позволит синтезировать ИТ обработки сигналов для некоторого подмножества актуальных прикладных задач косвенного контроля и диагностики.

Будем рассматривать задачи, для которых на основании априорной информации известно, что потенциально пригодными признаками являются характеристики формы входного сигнала $x(t)$. К такого рода постановкам сводятся многие задачи технической и медицинской диагностики, в частности — диагностика сердечно-сосудистой системы путем интерпретации электрокардиограмм, реограмм, сфигмограмм, косвенный контроль характеристик материалов методом термического анализа и целый ряд других актуальных в практическом плане задач.

Инструментальная система. Ограничимся рассмотрением класса объектов, для которых имеется следующая априорная информация:

- на интервале наблюдения $t \in [0, T]$ сигнала $x(t)$ объект может находиться только в одном из своих состояний W_1, \dots, W_M и сохраняет неизменными свои параметры z_1, \dots, z_N ;
- на интервале $t \in [0, T]$ в объекте происходят физические процессы, которые порождают появление на наблюдаемой реализации $x = x(t)$ одного или нескольких специфических по форме фрагментов (связных участков сигнала), однако заранее неизвестны как моменты появления таких фрагментов $t_v \in [0, T]$, так и их протяженности во времени τ_v ;
- каждый тип подобного фрагмента может быть формально описан и идентифицирован символом Φ_i ($1, \dots, G$) некоторого конечного алфавита $\Phi = \{\Phi_1, \dots, \Phi_G\}$.
- при переходе объекта из одного состояния в другое и/или существенном (с точки зрения пользователя ИТ) изменении его параметров z_1, \dots, z_N закономерным, но неизвестным нам образом изменяются моменты t_v появления

фрагментов из Φ , их протяженности τ_v во времени и другие вычислимые характеристики фрагментов.

Для синтеза ИТ в условиях такой ограниченной информации и предназначена предлагаемая инструментальная система.

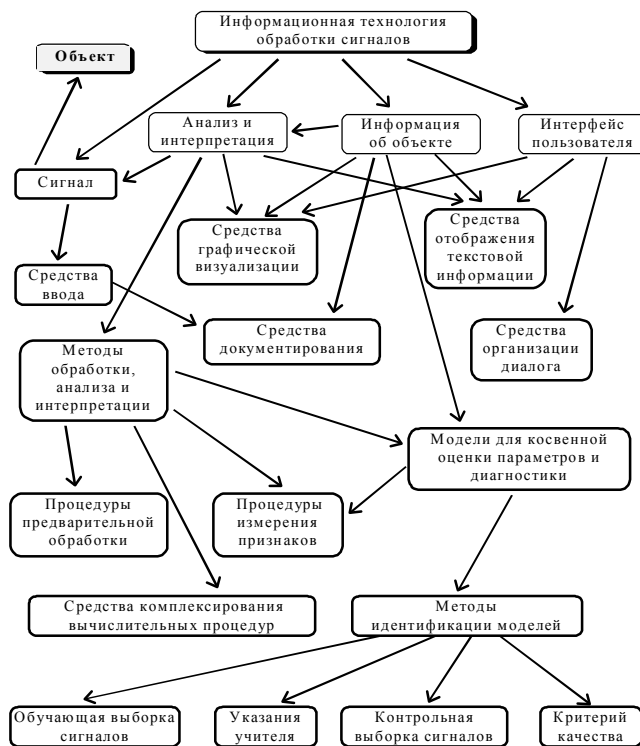


Рис. 4

На рис. 4 представлена информационная модель (структура) инструментальной системы, в которой с позиций объектного анализа [9, 10] идентифицированы ее основные сущности (домены). Направление связи обозначает, что домен более высокого уровня использует средства, обеспечиваемые более низким доменом. Дадим краткие пояснения к некоторым из этих доменов.

Процедуры предварительной обработки. Обычно порождаемый объектом сигнал $x(t)$ искажен случайными и частотными помехами, причем чаще всего характеристики этих помех заранее неизвестны. Разумеется, можно по имеющимся реализациям попытаться восстановить эти характеристики и затем воспользоваться формальными методами синтеза оптимальных в некотором смысле фильтров. Но есть и другой путь — заранее реализовать в инструментальной системе широкий на-

бор процедур фильтрации, ориентированных на различные модели помех, и методом проб и ошибок в интерактивном режиме выбрать процедуры фильтрации, которые, с точки зрения эксперта, адекватны имеющимся данным, и настроить их параметры.

Именно такой подход, который, на наш взгляд, быстрее приводит к цели, и реализован в предлагаемой инструментальной системе, в частности — в домене процедур предварительной обработки. При этом в системе организован удобный язык диалога с пользователем, реализующий известный принцип *Point-and-Select* («Укажи и выбери») — для обеспечения доступа к данным достаточно указать мышью на соответствующий график в рабочем окне, а для вызова вычислительной процедуры — выбрать пункт меню и, если это необходимо, в раскрывающемся диалоговом окне задать требуемые параметры процедуры. В результате на экране будет представлена вся информация, отражающая результаты обработки, в частности — график отфильтрованного сигнала.

Методы предварительной обработки исходных данных x_1, \dots, x_k реализованы в виде библиотеки процедур, фрагмент которой иллюстрирует рис. 5.



Рис. 5

Процедуры измерения признаков реализованы в виде библиотеки алгоритмов, обеспечивающих обнаружение фрагментов сигнала определенной формы на фоне возможных искажений и вычисление характеристик фрагментов, в том числе

моментов t_v , их появления, протяженностей во времени τ_v , соответствующих уровней сигнала x_v , производных сигнала в моменты времени t_v и ряд других характеристик, численные значения которых используются в качестве аргументов математических моделей для косвенной оценки параметров объекта и диагностики его функционального состояния. Иерархическая структура этой библиотеки ориентирована на поиск и обнаружение фрагментов следующих трех уровней:

- **фрагмент <свойство>** — последовательность дискретных значений, удовлетворяющая заданному свойству, в частности — ε -монотонности, ε -выпуклости и т.п., где ε — заданный порог нечувствительности к изменению этого свойства в пределах фрагмента, вызванному ограниченной по модулю помехой $|\eta| \leq \varepsilon$;
- **фрагмент <функция>** — последовательность дискретных значений, для которых может быть обеспечена ε -аппроксимация функцией $f_i(t, \theta)$, $i = 1, \dots, I$, из заданного множества функций с точностью до вектора параметров $\theta = (\theta_1, \dots, \theta_p)$;
- **фрагмент <параметр>** — последовательность дискретных значений, для которых может быть обеспечена ε -аппроксимация функцией $f_i(t, \theta)$, $i = 1, \dots, I$, из заданного множества функций при ограничениях на значения параметра θ_p ($p = 1, \dots, P$) вида $\theta_p^{\min} \leq \theta_p \leq \theta_p^{\max}$.

Средства комплексирования вычислительных процедур позволяют в режиме интерактивного взаимодействия синтезировать комплексные алгоритмы обработки сигнала на основе базовых вычислительных процедур. При этом сама инструментальная система открыта для расширения: в любой момент в ее состав может быть введена новая вычислительная процедура. Такие возможности обеспечиваются взаимодействием трех универсальных компонент (объектов) инструментальной системы — обобщенной модели сигнала, обобщенной модели обработки и графа, отображающего структуру взаимодействия вычислительных процедур обработки на этапе синтеза конкретной ИТ.

Обобщенная модель сигнала выполняет роль «носителя» информации в инструментальной сис-

теме. Поля этого объекта представляют собой четверку

$$Q = \langle D, G, A, S \rangle, \quad (4)$$

где D — поля данных (одно-, двух- и трехмерные динамические массивы дискретных отсчетов), G — текущие графические атрибуты этих данных (цвет графика, его расположение на экране монитора, коэффициенты масштабирования и т.п.), A — атрибуты данных (наименование, тип используемой процедуры обработки, параметры настройки вычислительного алгоритма и т.п.). В объекте Q также инкапсулированы свойства S , реализующие набор сервисных методов работы с объектом, в том числе ряд графических процедур (перемещение и масштабирование графика по экрану монитора на фоне остальных неподвижных графиков, просмотр его фрагментов в увеличенном масштабе, отображение координат указанной пользователем точки графика и т. п.).

Обобщенная модель обработки реализует последовательность виртуальных методов, обеспечивающих взаимодействие носителей информации Q входных и выходных данных с конкретной вычислительной процедурой обработки, задание в режиме диалога требуемых параметров вычислительной процедуры и т. п.

Рис. 7 иллюстрирует схему взаимодействия объектов Q в процессе синтеза ИТ. При вызове очередной вычислительной процедуры порождается новый экземпляр объекта Q_{i+1} , который частично наследует соответствующие поля объекта-прародителя Q_i . Объединение (инкапсуляция) в объектах Q полей (4) позволяет обеспечить, с одной стороны, доступ к соответствующим полям данных через текущие экранные координаты графика и, с другой стороны — привязку графика, отображающего результат обработки, к графику исходных данных.

Практические результаты. Предложенный подход апробирован при синтезе ряда прикладных ИТ обработки сигналов. Разработанная инструментальная система позволила достаточно быстро провести необходимые исследования и синтезировать конкретные прикладные ИТ [1, 2], в том числе ИТ для косвенного контроля технологических параметров железоуглеродистых расплавов по кривым охлаждения, ИТ диагностики патологий организма по кривым биофлуоресценции и ИТ для оценки функционального состояния операторов по

биоциклическому сигналу, отображаемому в фазовом пространстве координат.

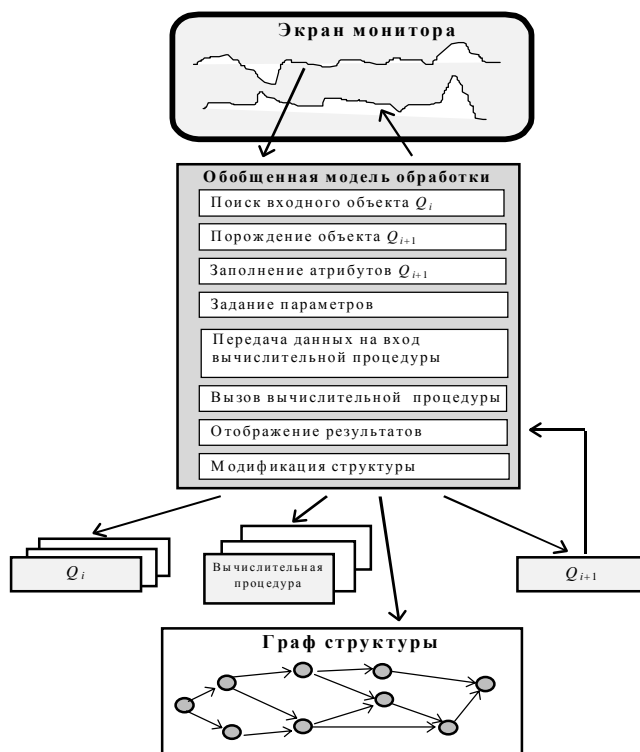


Рис. 7

Заключение. Реализованный в инструментальной системе подход к синтезу ИТ обработки сигналов базируется на двух главных принципах: модульности базовых компонент и единства формальной и неформальной технологии синтеза.

Первый из указанных принципов сводится к тому, что конструктору ИТ предоставляется широкий набор универсальных компонент (вычислительных и сервисных процедур), которые реализуют отдельные элементарные стадии ИТ. Вторым принципом, предполагающим формализацию при разработке базовых вычислительных методов, допускает возможность неформальных действий конструктора ИТ при объединении вычислительных процедур в сложные структуры обработки сигнала. При этом в условиях ограниченной априорной информации конструктор действует методом проб и ошибок, полагаясь на свой опыт и интуицию, анализируя результаты текущей стадии обработки в виде текстовой и графической информации, а также руководствуясь мнением эксперта в рассматриваемой предметной области.

Следует еще раз подчеркнуть, что объектная архитектура инструментальной системы, подобно общей шине современного компьютера, открыта для расширения: для интеграции в систему новых вычислительных процедур, в которых может появиться необходимость при синтезе конкретной ИТ, достаточно просто при написании соответствующих программных модулей выполнить ряд не очень обременительных ограничений, налагаемых на интерфейс вызываемых процедур.

Таким образом инструментальная система находится в постоянном развитии: существующие средства помогают быстро и эффективно провести необходимые исследования при синтезе новой ИТ, а возникающие при этом проблемы неизбежно порождают новые вычислительные модули, которые расширяют ее функциональные возможности. Сама же система используется в качестве «полигона» в процессе разработки, исследований и сравнительной оценки оригинальных алгоритмов обработки сигналов.

1. *Скурихин В.И., Файнзильберг Л.С., Потапова Т.П.* Инструментальная система СИДИГРАФ для синтеза информационных технологий обработки сигналов // УСиМ. — 1996. — № 6. — С. 36—45.
2. *Скурихин В.И., Файнзильберг Л.С., Потапова Т.П.* Инструментальная система для синтеза информационных технологий обработки сигналов различной физической природы // Сб. науч. тр. междунар. конф. «ЗНАНИЕ–ДиАЛОГ–РЕШЕНИЕ». — Ялта, 1995. — Т. 2. — С. 380–391.

3. *Белкин А.В., Левин М.Ш.* Принятие решений: комбинаторные модели аппроксимации информации. — М.: Наука, 1990. — 160 с.
4. *Ивахненко А.Г., Юрачковский Ю.П.* Моделирование сложных систем по экспериментальным данным. — М.: Радио и связь, 1987. — 120 с.
5. *Алгоритмы и программы восстановления зависимостей* / Под ред. В.Н. Вапника. — М.: Наука, 1984. — 814 с.
6. *Нильсон Н.* Обучающиеся машины. — М.: Мир, 1967. — 180 с.
7. *Степашко В.С.* Комбинаторный алгоритм МГУА с оптимальной схемой перебора моделей // Автоматика. — 1981. — № 3. — С. 31–36.
8. *Юрачковский Ю.П.* Восстановление полиномиальных зависимостей на основе самоорганизации // Автоматика. — 1981. — № 4. — С. 15–20.
9. *Шлеер С., Меллор С.* Объектно-ориентированный анализ: моделирование мира в состояниях. — Киев: Диалектика, 1993. — 240 с.
10. *Буч Г.* Объектно-ориентированное проектирование с примерами применения. — М.: Конкорд, 1992. — 519 с.

Поступила 23.01.98
Тел. для справок: (044) 267-60-23 (Киев)
© Л.С. Файнзильберг, 1998