



УДК 621.01219

Э. М. Фархадзаде, д-р техн. наук,
А.З. Мурадалиев, канд. техн. наук, **Ю.З. Фарзалиев**
Азербайджанский научно-исследовательский
и проектно-изыскательский ин-т энергетики
(Азербайджанская Республика, Az 1012, Баку, пр-т Зардаби, 94,
тел. (99412) 431-64-07, E-mail: fem1939@rambler.ru)

Разработка интерфейса автоматизированной системы анализа надежности энергоблоков ГРЭС

(Статью представил чл.-кор. НАН Украины В. В. Васильев)

Рассмотрены вопросы организации и характеристика дружественного интерфейса автоматизированной системы анализа и контроля технического состояния оборудования и устройств энергоблоков ГРЭС.

Розглянуто питання організації та характеристику дружнього інтерфейсу автоматизованої системи аналізу і контролю технічного стану устаткування та пристроїв енергоблоків ГРЕС.

К л ю ч е в ы е с л о в а: энергоблок, автоматизированная система, интерфейс.

Снижение эксплуатационных затрат неразрывно связано с проблемами объективной оценки технического состояния оборудования и устройств электроэнергетических систем (ЭЭС), организации их эксплуатации, восстановления износа и своевременной замены. Непременным условием решения этих проблем является систематизация ретроспективной информации, включающей паспортные, нормативные и справочные данные, данные измерения диагностических параметров, данные перехода в нерабочие, в частности аварийные состояния, планируемые и фактические значения основных производственных показателей и др. Большие объемы статистических данных, возможные ошибки при их систематизации вручную, наукоемкость современных методов статистического анализа требуют разработки специализированных автоматизированных систем, позволяющих оперативно представлять необходимую информацию и решать конкретные задачи, связанные с повышением надежности и эффективности работы. Результаты расчетов и полученные рекомендации являются информационной поддержкой для специалистов.

Переход от интуитивного подхода, применения метода «проб и ошибок», к использованию систематизированной ретроспективной информации в процессе принятия решения, несомненно, повышает эффективность эксплуатации, снижает риск принятия ошибочных решений.

Сведения об используемых на практике компьютерных технологиях [1], в том числе и сведения о разработанной в АзНИПИИЭ автоматизированной системе [2], традиционно включают перечень и алгоритмы решения конкретных эксплуатационных задач, иллюстративные примеры. Существенно меньше сведений о достоверности информационной поддержки, нарушаемой вследствие:

ошибок при вводе данных с клавиатуры компьютеров, при программных сбоях ЭВМ и влиянии внешних факторов (например, качество электроэнергии);

случайного характера оценок показателей надежности;

традиционных предположений, на основе которых получены асимптотические табличные данные критических значений критериев согласия.

Соответствующим специалистам предприятий энергосистемы в конечном итоге нужны лишь рекомендации при решении конкретных задач. Однако наряду с этим (в частности, при различии рекомендаций с интуитивными решениями) в компьютерных технологиях должна предусматриваться возможность (при подтверждении запроса) получить в табличной или графической форме с комментариями лаконичную информацию, обосновывающую принятие рекомендации. Поэтому одним из основных требований, предъявляемых к автоматизированным системам, является дружественный интерфейс (ДИ), выполняющий функции путеводаителя при решении конкретных задач, в частности, позволяющий получить ответы на возникающие вопросы (например, выявить наименее надежные узлы объекта) и сомнения (например, безошибочность данных о техническом состоянии объекта). Именно ДИ обеспечивает удобство и успех практического использования компьютерных технологий при эксплуатации оборудования и устройств на предприятиях энергосистем.

Рассмотрим вопросы организации интерфейса автоматизированной системы, характеристика которой изложена в [2]. Возможности организации ДИ непосредственно зависят от используемой системы управления базой данных (СУБД). Многолетний опыт работы показывает, что наиболее полно требования к создаваемой базе данных о техническом состоянии оборудования и устройств ЭЭС и решаемым задачам соответствуют возможности СУБД PARADOX (язык программирования — PAL) фирмы BORLAND [3].

Достоинства СУБД PARADOX следующие:

база данных может содержать любую информацию (в виде списка, таблицы, формы); элементами информации могут быть слова, числа или даты;

обладает мобильной системой запроса, позволяющей оперативно получать необходимую информацию; каждый запрос остается активным до тех пор, пока не удален; запрос, затрагивающий множество таблиц, обрабатывается так же быстро, как и запрос по одной таблице;

наличие встроенного языка программирования, позволяющего создавать сложные системы обработки данных, ориентированные на конкретные практические задачи;

обладает средствами диагностики информации, включающими автоматический контроль календарных дат, просмотр таблиц для контроля их целостности, автоматический контроль формата данных (последовательность записи данных), автоматический контроль неполноты данных, автоматический контроль формата данных;

обеспечивает, при необходимости, избирательный доступ к информации.

Основные трудности организации ДИ. Несмотря на указанные выше возможности, разработка ДИ требует преодоления ряда трудностей, ограничивающих формализацию управления. Наиболее значимыми из них являются следующие:

однородность форм исходного меню и необходимость разработки специальных программ, обеспечивающих удобный для пользователя интерфейс;

большой расход оперативной памяти при предусмотренном в СУБД принципе организации меню, что существенно сокращает возможности увеличения уровней иерархии меню;

по завершении каждого этапа расчета осуществляется автоматический возврат к исходному автоматизированной системы, хотя оперативность решения требует передачи управления начальному блоку меню решаемой задачи;

недостаточное число уровней иерархии СУБД ограничивает перечень решаемых задач и, в частности, возможности детализации конструкции объекта;

при одновременной работе с несколькими взаимосвязанными эмпирическими таблицами недостаточный учет особенностей взаимосвязи ведет к срыву программы вычислений;

отсутствие возможности оперативного контроля достоверности ряда взаимосвязанных вычислений.

Формирование структуры ДИ. Характерной особенностью решения большей части эксплуатационных задач и, в частности задач оценки технического состояния оборудования и устройств, является использование

выборку данных из эмпирических таблиц. Эта особенность обуславливает необходимость поэтапной классификации ретроспективных данных по ряду заданных признаков. Набор фрагментов меню ограничивается размерами экрана монитора и зависит от множества задач, решаемых автоматизированной системой. Процесс формирования этой системы — динамический, так как увеличение перечня решаемых задач находится в обратной связи с заинтересованностью заказчика в результатах анализа ретроспективных данных.

На рис. 1 приведены фрагменты меню, позволяющие получить технические характеристики пароводяного тракта поверхности нагрева паровой котельной установки блока 300 МВт на газомазутном топливе. Для сокращения числа фрагментов из рассмотрения исключен перечень электростанций (ЭС) блочного типа.

Будем рассматривать отдельные узлы пароводяного тракта котла. Изложенное выше подтверждает возможность использования одних и тех же фрагментов меню при решении различных информационных задач, что, в свою очередь, определяет формирование структуры каждого меню, число фрагментов меню и определяет уровень иерархии в глубину. Множество пунктов меню обуславливает множество возможных направлений маршрута. Задача определения минимальной протяженности маршрута визуальна по структуре меню достаточно проста. Однако процесс движения по выбранному маршруту требует разработки специальных программ управления и, прежде всего, формализации самого процесса.

На рис. 1 каждый блок меню обозначен $X_{i,j}$, где $i = 1, 2, 3, \dots$ — уровень иерархии в глубину, а $j = 1, 2, 3, \dots$ — порядковый номер блоков меню на каждом уровне иерархии. Последовательность движения по маршруту целесообразно визуализировать средствами теории графов. Для этого достаточно идентификаторы $X_{i,j}$ ($i = 1, n, j = 1, n_i$, где n — уровень иерархии маршрута, а n_i — число блоков меню на i -м уровне иерархии) каждого уровня иерархии представить вершинами графа и соединить хордами соответствующие вершины каждого уровня.

Структурная схема маршрута по фрагментам меню, представленным на рис. 1, приведена на рис. 2. На каждом уровне маршрут начинается со строго фиксированной вершины с порядковым номером, равным единице. Из каждой вершины возможно движение в шести направлениях. Первые четыре осуществляются на плоскости каждого уровня: налево, направо, вверх и назад. Пятое направление — возвращение на предыдущий уровень иерархии и шестое — движение на последующий уровень иерархии.

Как следует из рис. 2, граф для выбранного маршрута достаточно сложен:

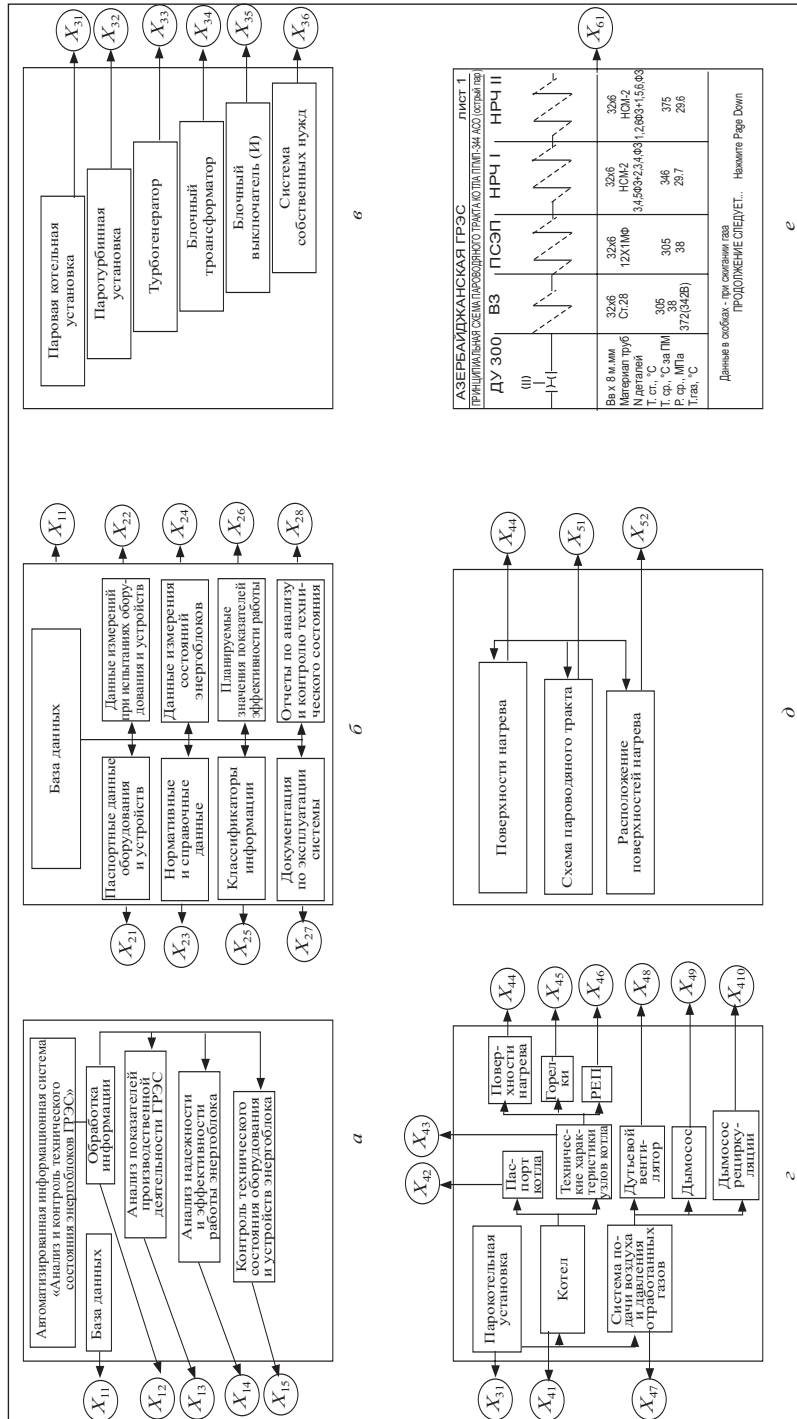


Рис. 1. Фрагменты меню, характеризующие поиск информации о технических характеристиках узлов пароводяного тракта поверхности нагрева котлоагрегата: *а* — структурная схема автоматизированной системы с базой данных, где размещена требуемая информация; *б* — структурная схема базы данных; *в* — установка энергоблока, включающая паровую котельную установку; *г* — совокупность основных узлов паровой котельной установки, включающая паровую котельную установку; *д* — меню, включающее основную схему пароводяного тракта со всеми узлами и их параметрами и компоновку этих узлов; *е* — фрагмент схемы пароводяного тракта котла

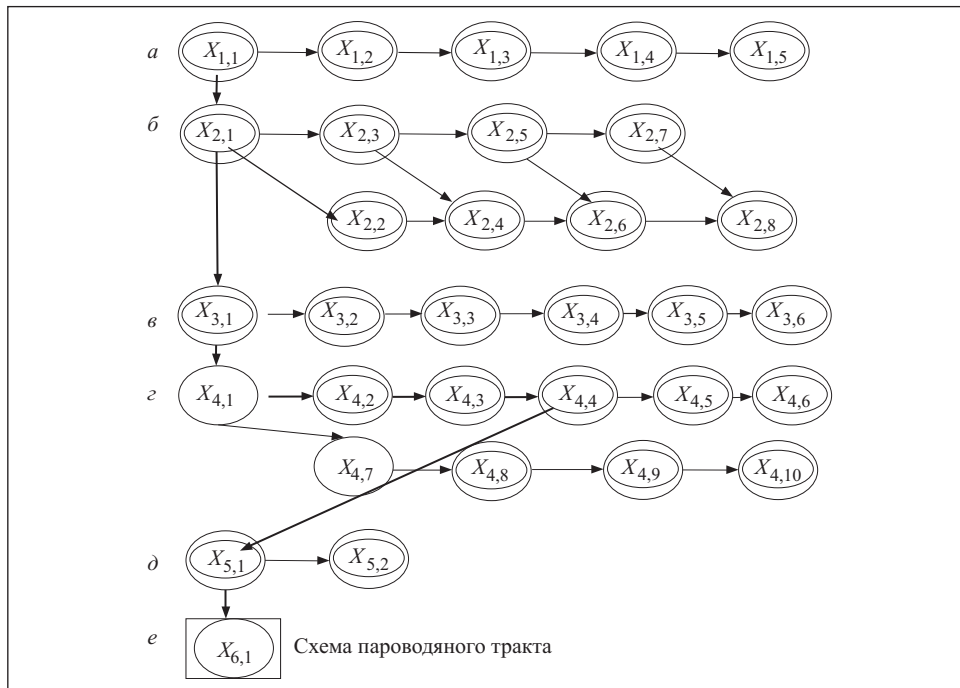


Рис. 2. Граф маршрута движения по фрагментам меню, представленным на рис. 1, а — е

прежде всего, он ориентирован, так как движение возможно в обоих направлениях;

он связан, так как любая пара вершин может быть соединена простой цепью;

на отдельных уровнях иерархии он может быть представлен как плоский граф;

плоские графы могут включать параллельные простые цепи с различным числом вершин.

Основное назначение графа — формирование алгоритма движения по маршруту при решении конкретной задачи, связанной с оценкой технического состояния оборудования. Однако указание возможных направлений движения и увязка с топологией меню существенно затрудняют непосредственный переход от графа к алгоритму и программе управления движением. Существенную поддержку в этом процессе оказывает применение математического аппарата алгебры логики.

Обозначим индексом конъюнкции Λ направление движения от одного пункта меню к другому, а индексом дизъюнкции V — возможные варианты движения. Направление движения по уровням иерархии осуществляется по-

следовательным увеличением (уменьшением) порядкового номера уровня по формуле

$$X_{(i-1),\xi} \leftarrow X_{i,j} \rightarrow X_{(i+1),1}^{(j)},$$

где ξ — условный номер блока меню $(i - 1)$ уровня, характеризующего предшествующую вершину маршрута; $X_{(i+1),1}^{(j)}$ — идентификатор, обозначающий первый пункт меню $(i + 1)$ уровня для j -го пункта меню предыдущего i -го уровня. При $i = n$ эта формула имеет вид $X_{i,j} \rightarrow X_{(i-1),\xi}$.

В качестве примеров приведем алгоритмы возможного перемещения на плоских графах первого, второго и четвертого уровней иерархии, соответствующих различным по структуре меню, представленным на рис. 1, а, б и г:

- 1 : $(\overline{X_{1,1} \wedge \text{ВыборЭС}})V(X_{1,1} \wedge X_{2,1})V(\overline{X_{1,1} \wedge X_{1,2}});$
 $(\overline{X_{1,2} \wedge \text{ВыборЭС}})V(X_{1,2} \wedge X_{2,1})V(\overline{X_{1,2} \wedge X_{1,1}})V(X_{1,2} \wedge X_{1,3});$
 $(\overline{X_{1,3} \wedge \text{ВыборЭС}})V(X_{1,3} \wedge X_{2,1})V(\overline{X_{1,3} \wedge X_{1,2}})V(X_{1,3} \wedge X_{1,4});$
 $(\overline{X_{1,4} \wedge \text{ВыборЭС}})V(X_{1,4} \wedge X_{2,1})V(\overline{X_{1,4} \wedge X_{1,3}})V(X_{1,4} \wedge X_{1,5});$
 $(\overline{X_{1,5} \wedge \text{ВыборЭС}})V(X_{1,5} \wedge X_{2,1})V(\overline{X_{1,5} \wedge X_{1,4}})V(\overline{X_{1,5} \wedge X_{1,2}});$
- 2 : $(\overline{X_{2,1} \wedge X_{1,\xi}})V(X_{2,1} \wedge X_{3,1})V(\overline{X_{2,1} \wedge X_{2,3}})V(X_{2,1} \wedge X_{2,2});$
 $(\overline{X_{2,2} \wedge X_{1,\xi}})V(X_{2,2} \wedge X_{3,1})V(\overline{X_{2,2} \wedge X_{2,1}})V(X_{2,2} \wedge X_{2,4});$
 $(\overline{X_{2,3} \wedge X_{1,\xi}})V(X_{2,3} \wedge X_{3,1})V(\overline{X_{2,3} \wedge X_{2,1}})V(X_{2,3} \wedge X_{2,5})V(\overline{X_{2,3} \wedge X_{2,4}});$
 $(\overline{X_{2,4} \wedge X_{1,\xi}})V(X_{2,4} \wedge X_{3,1})V(\overline{X_{2,4} \wedge X_{2,2}})V(X_{2,4} \wedge X_{2,6})V(\overline{X_{2,4} \wedge X_{2,3}});$
 $(\overline{X_{2,5} \wedge X_{1,\xi}})V(X_{2,5} \wedge X_{3,1})V(\overline{X_{2,5} \wedge X_{2,3}})V(X_{2,5} \wedge X_{2,7})V(X_{2,5} \wedge X_{2,6});$
 $(\overline{X_{2,6} \wedge X_{1,\xi}})V(X_{2,6} \wedge X_{3,1})V(\overline{X_{2,6} \wedge X_{2,4}})V(X_{2,6} \wedge X_{2,5})V(X_{2,6} \wedge X_{2,8});$
 $(\overline{X_{2,7} \wedge X_{1,\xi}})V(X_{2,7} \wedge X_{3,1})V(\overline{X_{2,7} \wedge X_{2,5}})V(X_{2,7} \wedge X_{2,8});$
 $(\overline{X_{2,8} \wedge X_{1,\xi}})V(X_{2,8} \wedge X_{3,1})V(\overline{X_{2,8} \wedge X_{2,7}})V(\overline{X_{2,8} \wedge X_{2,6}});$
- 4 : $(\overline{X_{4,1} \wedge X_{3,1}})V(X_{4,1} \wedge X_{4,7})V(\overline{X_{4,1} \wedge X_{4,2}});$
 $(\overline{X_{4,2} \wedge X_{3,1}})V(X_{4,2} \wedge X_{5,1}^{(2)})V(\overline{X_{4,2} \wedge X_{4,1}})V(X_{4,2} \wedge X_{4,3});$
 $(\overline{X_{4,3} \wedge X_{3,1}})V(\overline{X_{4,3} \wedge X_{4,2}})V(X_{4,3} \wedge X_{4,4})V(\overline{X_{4,3} \wedge X_{4,1}});$
 $(\overline{X_{4,4} \wedge X_{3,1}})V(X_{4,4} \wedge X_{5,1}^{(4)})V(\overline{X_{4,4} \wedge X_{4,3}})V(X_{4,2} \wedge X_{4,5});$
 $(\overline{X_{4,5} \wedge X_{3,1}})V(X_{4,5} \wedge X_{5,1}^{(5)})V(\overline{X_{4,5} \wedge X_{4,4}})V(X_{4,5} \wedge X_{4,6})V(\overline{X_{4,5} \wedge X_{4,3}});$
 $(\overline{X_{4,6} \wedge X_{3,1}})V(X_{4,6} \wedge X_{5,1}^{(6)})V(\overline{X_{4,6} \wedge X_{4,5}})V(\overline{X_{4,6} \wedge X_{4,3}});$
 $(\overline{X_{4,7} \wedge X_{3,1}})V(\overline{X_{4,7} \wedge X_{4,1}})V(\overline{X_{4,7} \wedge X_{4,8}});$

$$\begin{aligned} & (\overline{X_{4,8} \wedge X_{3,1}})V(X_{4,8} \wedge X_{5,1}^{(8)})V(X_{4,8} \wedge X_{4,9})V(\overline{X_{4,8} \wedge X_{4,7}}); \\ & (\overline{X_{4,9} \wedge X_{3,1}})V(X_{4,9} \wedge X_{5,1}^{(9)})V(\overline{X_{4,9} \wedge X_{4,8}})V(X_{4,9} \wedge X_{4,10})V(\overline{X_{4,9} \wedge X_{4,7}}); \\ & (\overline{X_{4,10} \wedge X_{3,1}})V(X_{4,10} \wedge X_{5,1}^{(10)})V(\overline{X_{4,10} \wedge X_{4,9}})V(\overline{X_{4,10} \wedge X_{4,7}}). \end{aligned}$$

Рассмотрим некоторые особенности этих алгоритмов:

- 1) возвращение на предыдущий уровень иерархии осуществляется в пункт ξ меню маршрута движения;
- 2) переход на предыдущий уровень иерархии осуществляется на первую позицию пунктов меню;
- 3) движение в обратном направлении обозначается чертой сверху;
- 4) при многоуровневом меню переход на последующий уровень иерархии (в очередное меню) возможен лишь из пунктов меню, не имеющих разветвлений; на рис. 2 эти пункты (вершины) обозначены одним кружком.

Методы преодоления трудностей организации ДИ. *Метод процедуры* основан на том факте, что процедуры в СУБД PARADOX работают значительно быстрее, чем программы в целом. Для каждого пункта меню составляются процедуры прямого и обратного хода. Возможным направлениям движения сопоставляется понятие «истина» или «ложь», в зависимости от совпадения направления движения с направлением маршрута. При организации интерфейса необходимо учесть следующее:

при каждом изменении направления движения очищается рабочая область (стирается с экрана меню);

одним из основных элементов интерфейса является изменение цвета пункта меню, соответствующего местоположению при движении по маршруту;

все процедуры записываются в библиотеку; активизация процедур проводится вызовом по их наименованию.

Этот метод прост и удобен для реализации при условии, что граф на данном уровне иерархии представляет собой элементарную цепь (например, графы первого, третьего и пятого уровней (см. рис.1)).

Процедура очередного перемещения по маршруту сводится к такой последовательности операций:

- блокировка всех команд с клавиатуры;
- активизация кнопки, определяющей направление маршрута;
- выделение пункта меню, соответствующего направлению движения, изменением цвета;
- сохранение структуры меню.

Если граф меню отличается от элементарной цепи, рассматриваемый метод оказывается недостаточно эффективным, так как приводит к су-

щественным затратам оперативной памяти и необходимости изменения структуры управления при добавлении к меню дополнительных блоков

Метод выбора маршрута в многоуровневом меню. Отличительной особенностью этого метода является предварительное фиксирование не только идентификатора ($X_{i,j}$), но и координат каждого пункта меню.

Алгоритм перемещения по маршруту сводится к такой последовательности операций:

блокировка всех команд с клавиатуры;

в соответствии с целесообразным направлением движения осуществляется поиск соответствующего направления движения пункта по координатам;

выделение пункта меню, соответствующего направлению движения, изменением его цвета;

сохранение структуры меню;

обратный ход упрощается «метками», оставленными по выбранному маршруту.

В качестве примера реализации этого метода рассмотрим меню для рис.1, 2. При использовании метода процедур переход от пункта меню «Горелки» к пункту «Поверхности нагрева» невозможен, так как слева от пункта «Горелки» на том же уровне находится пункт меню «Котел» (совпадают координаты строки). Поскольку проверка проводится сначала по координатам строки, PARADOX автоматически ошибочно выбирает следующую вершиной направления пункт «Котел».

В рекомендуемом методе правильный выбор направления движения обеспечивается поиском координат заранее фиксированной метки. Система возврата аналогична модели поиска направления движения в лабиринте по меткам.

Метод искусственного увеличения числа уровней иерархии. Существующие ограничения экрана монитора ограничивают число уровней иерархии. Можно было бы указать все уровни на одной плоскости, но при этом существенно сокращается наглядность и повышается сложность программ поиска маршрута. СУБД PARADOX допускает восемь уровней иерархии, что в ряде случаев вносит ограничения как в возможности паспортизации данных, так и в решение конкретных эксплуатационных задач. Ограничения обусловлены величиной ресурса оперативной памяти, выделенной для СУБД PARADOX.

По данным электронных средств информации фирмой BORLAND дополнительно разработан алгоритм, обеспечивающий 12 уровней иерархии интерфейса.

Первоначально была рассмотрена возможность отображения меню девятого и последующих уровней путем освобождения в оперативной

памяти информации последнего уровня. Суть алгоритма заключается в следующем:

в восьмом (конечном) уровне иерархии некоторому идентификатору присваивается значение $S = 1$;

из восьмого уровня иерархии осуществляется возврат в седьмой уровень;

в седьмом уровне предусматривается условие: если $S = 1$, то осуществляется переход на девятый уровень;

в девятом уровне предусматривается присвоение идентификатору значения $S = 2$;

из девятого уровня осуществляется возврат на седьмой уровень, где очередным условием является проверка условия $S = 2$, и если это условие выполнено, то осуществляется изображение меню десятого уровня и т. д.

Возможности СУБД PARADOX ограничивали глубину иерархии по этому алгоритму до 12-ти. В последующем был разработан алгоритм, позволяющий выходить на произвольную глубину иерархии маршрута поиска информации. Идея этого алгоритма заключается в совмещении принципов предшествующего алгоритма, записи необходимой информации на внешнюю память и использовании «маяков» для распознавания информации.

Выводы. Одной из основных составляющих автоматизированной системы анализа и контроля надежности энергоблоков является разработка интерфейса. Специализированный характер таких систем, рассчитанных на узкий круг специалистов, обуславливает повышенные требования к оперативности, определенности и наглядности решаемых задач. Эти требования выполняются при использовании следующих методов:

поэтапной классификации ретроспективных данных по ряду признаков и их разновидностей на основе сочетания теории графов и алгебры логики;

выбора маршрута в многоуровневом меню на основе поиска координат заранее фиксированной «метки»;

произвольного увеличения числа уровней иерархии, что не предусмотрено в других СУБД; такое увеличение вызвано характером решаемых информационных задач для сложных энергетических объектов, к которым относятся энергоблоки ГРЭС.

Problems of architecture and characteristics of the friendly interface for the computer-aided check-out and control system for the equipment of the state district power station power-generating units are considered.

1. Васин В. П., Лоскутов В. Ф., Старшиков В. А., Поляков А. М. К развитию компьютерных технологий в эксплуатации электрооборудования электростанций // Электрические станции. — 2005. — № 4. — С. 33—40
2. Фархадзаде Э. М., Сафарова Т. Х., Мурадалиев А. З. и др. Автоматизированная система анализа индивидуальной надежности и эффективности энергоблоков ГРЭС// Электрические станции. — 2005. — № 11. — С. 38—47.
3. Тиней Д. Программирование в Paradox for Windows на примерах / Пер.с англ. — М. : БИНОМ, 1994. — 752 с.

Поступила 10.12.07;
после доработки 01.04.08

ФАРХАДЗАДЕ Эльмар Мехти оглу, д-р техн. наук, проф., руководитель лаборатории «Надежность энергетического оборудования» АзНИПИИ Энергетики. В 1961 г. окончил Азербайджанский ин-т нефти и химии (АЗИНЕФТЕХИМ). Область научных исследований — надежность и эффективность электроэнергетических систем.

МУРАДАЛИЕВ Айдын Зураб оглу, канд. техн. наук, ст. науч. сотр. лаборатории «Надежность энергетического оборудования» АзНИПИИ Энергетики. В 1982 г. окончил Азербайджанский ин-т нефти и химии (АЗИНЕФТЕХИМ). Область научных исследований — количественная оценка индивидуальной надежности оборудования и устройств электроэнергетических систем.

ФАРЗАЛИЕВ Юзиф Зейни оглу, инженер-математик, вед. инж. лаборатории «Надежность энергетического оборудования» АзНИПИИ Энергетики. В 1985 г. окончил Азербайджанский госуниверситет. Область научных исследований — точность и достоверность оценок показателей индивидуальной надежности энергоблоков ГРЭС.