

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИДЕНТИФИКАЦИИ СОСТОЯНИЙ БАРАБАННЫХ МЕЛЬНИЦ

Розглядається програмне забезпечення, яке розроблене для ідентифікації станів барабаних млинів через моментні характеристики супутніх сигналів.

Рассматривается программное обеспечение, которое разработано для идентификации состояний барабанных мельниц через моментные характеристики сопутствующих сигналов.

The software is **considered**, which is developed for authentication of **states** of drum mills through the momentnie descriptions of concomitant signals.

Для реализации математического аппарата вычисления моментных характеристик сигналов [1-4], используемых для идентификации технологических и технических состояний барабанных мельниц (БМ) разработано специальное программное обеспечение, которое содержит две основные формы. Главная форма служит для задания параметров моментной характеристики рассчитываемой на данном этапе и визуализации входного сигнала с полученными характеристиками. Форма сравнения служит для сравнения и анализа информационных характеристик однотипных моментных функций нескольких сигналов.

Пункт главного меню *Файл* → *Загрузить данные* (рис.1) предназначен для загрузки в программу измеренной последовательности значений входного сигнала. Первым значением в файле данных идет шаг дискретизации сигнала в секундах, за ним количество значений сигнала и далее сами значения реализации сигнала.

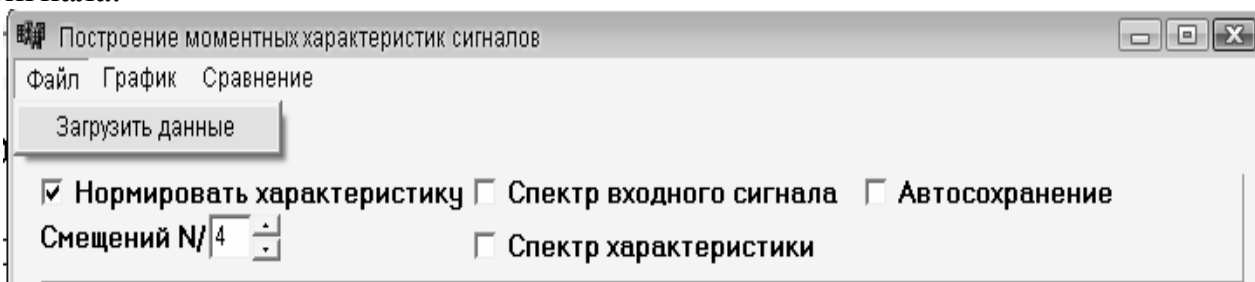


Рис. 1. Загрузка вектора значений сигнала из файла

В случае, если структура файла не совпадает с вышеописанной, выводится сообщение об ошибке. Для загрузки файла данных используется функция модуля `main.cpp` `LoadFunctions`. Вектор значений реализаций исходного информационного сигнала хранится в динамическом массиве `input`, шаг дискретизации – в переменной `dt`, количество значений сигнала – в переменной `N`.

В случае успешной загрузки в видовом окне первого графика отображается визуальное представление измеренной реализации анализируемого входного сигнала БМ. При работе в интерактивном режиме пользователю предлагается

задать параметры рассчитываемой моментной характеристики и базовые особенности построения ее визуализации.

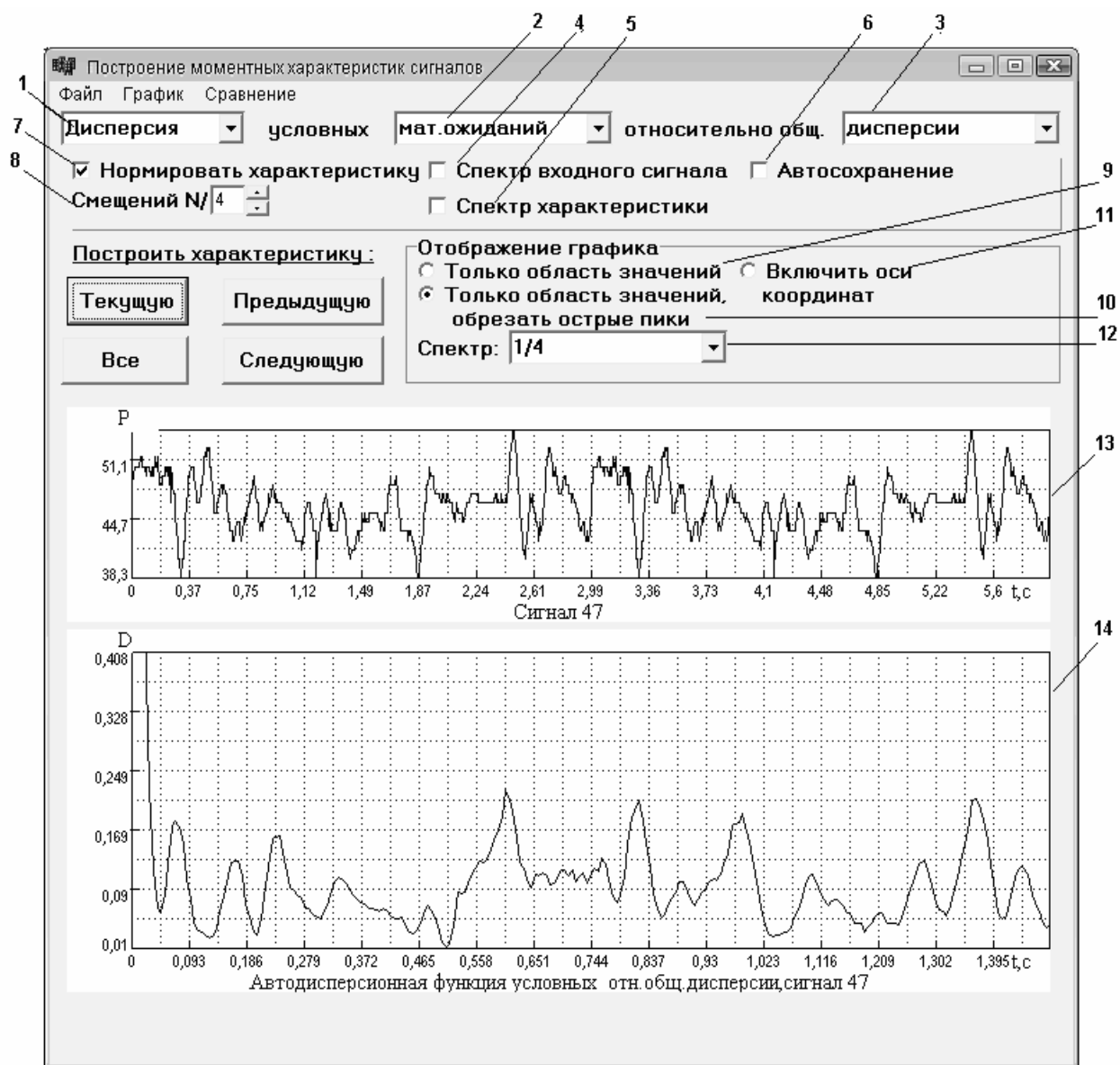


Рис. 2. Общий вид главной формы программы идентификации состояний барабанных мельниц

Элементы управления, обозначенные на рис. 2 номерами 1, 2 и 3 позволяют выбрать тип моментной функции которую требуется вычислить и тип ее модификации. Активизация элементов 4 или 5 позволяет рассчитать спектральную плотность соответственно входного сигнала или моментной характеристики. Отметив фокусом элемент 6, пользователь может определить что построенная визуализация моментной характеристики будет сразу же сохранена в *.bmp файл в текущей папке с именем по умолчанию, которое включает аббревиатуру рассчитанной моментной характеристики и название файла данных. Для того чтобы рассчитанная характеристика была нормирована, элемент 7 должен быть активизирован. Элемент 8 позволяет указать количество временных сдвигов m ,

которые должны быть выполнены при расчете моментной характеристики. Блок параметров *Отображение графика* служит для задания вида графика, при этом пункт 9 определяет выбор видового окна таким образом, чтобы график был максимально визуально приближен и при этом не выходил за рамки видимой части координатной плоскости, а пункт 10 работает аналогично, но активизирует функцию при которой чрезмерно острые пики графика, затрудняющие просмотр мелких деталей визуализации, будут обрезаться. Активизация пункта 11 задает функцию при которой график также должен быть максимально приближен, но при этом в видимую область должны быть включены оси координат. Элемент 12 позволяет выбрать необходимую часть графика спектральной плотности по оси частот, которая будет отображена на экране.

При нажатии на любую кнопку построения моментной характеристики все заданные параметры влияющие на процесс расчета записываются в объект `mOptions`. Чтобы построить моментную характеристику с параметрами заданными на форме необходимо нажать кнопку *Текущую*. Кнопки *Следующую* и *Предыдущую* служат для построения соответственно следующей и предыдущей моментной характеристики из общего множества доступных модификаций. Кнопка *Все* позволяет рассчитать все доступные моментные характеристики для загруженного сигнала, автоматически сохранив их визуализации в текущей папке в *.bmp файлах с именами по умолчанию.

Когда визуализация моментной характеристики получена, можно произвольно изменять видовое окно графиков 13, 14. Захватив левой кнопкой мыши, можно перемещать график в видовом окне, а зажав при этом клавишу Shift можно масштабировать видовое окно, что позволяет рассмотреть мелкие ранее не заметные детали графика. Щелчок правой кнопкой мыши на графике приведет к возвращению параметров видового окна в исходное состояние. При помощи щелчка правой кнопкой на графике с зажатой клавишей Shift можно сохранить текущий вид графика в *.bmp файл с именем по умолчанию. Кроме того графики исходного информационного сигнала и рассчитанной моментной характеристики можно сохранить в произвольный *.bmp файл при помощи пункта меню *График* → *Сохранить как*. Пункт меню *График* → *Сохранить* дает альтернативную возможность сохранения графиков в файл с именем по умолчанию (рис. 3.)

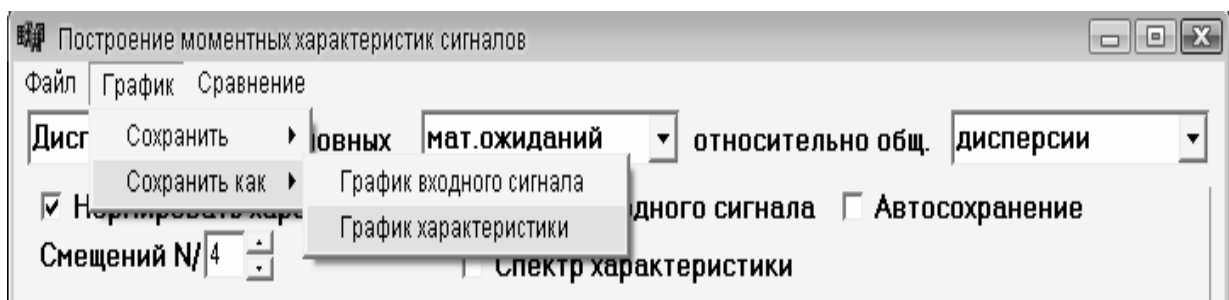


Рис. 3. Сохранение графиков моментных функций в файл с именем по умолчанию

С помощью пункта меню *Добавить* формы сравнения можно добавлять на форму визуализации моментной характеристики выбранного сигнала. При каждом использовании этого пункта меню параметры расчета моментной характеристики загружаются из главной формы, таким образом можно получить на форме сравнения произвольное количество визуализаций различных моментных характеристик для различных сигналов. Пункт меню *Удалить* служит для удаления последней добавленной на форму визуализации. Пункт меню *Графики* → *Сохранить все* позволяет сохранить все добавленные на форму графики в *.bmp файлы с шаблонными именами (например, test_1, test_2, test_3 и т.д.). При помощи пункта меню *Графики* → *Сохранить все как* можно сохранить все графики, которые добавлены на форму в *.bmp файлы с именами по умолчанию. Пункт меню *Сравнить* открывает форму сравнения (рис. 4).

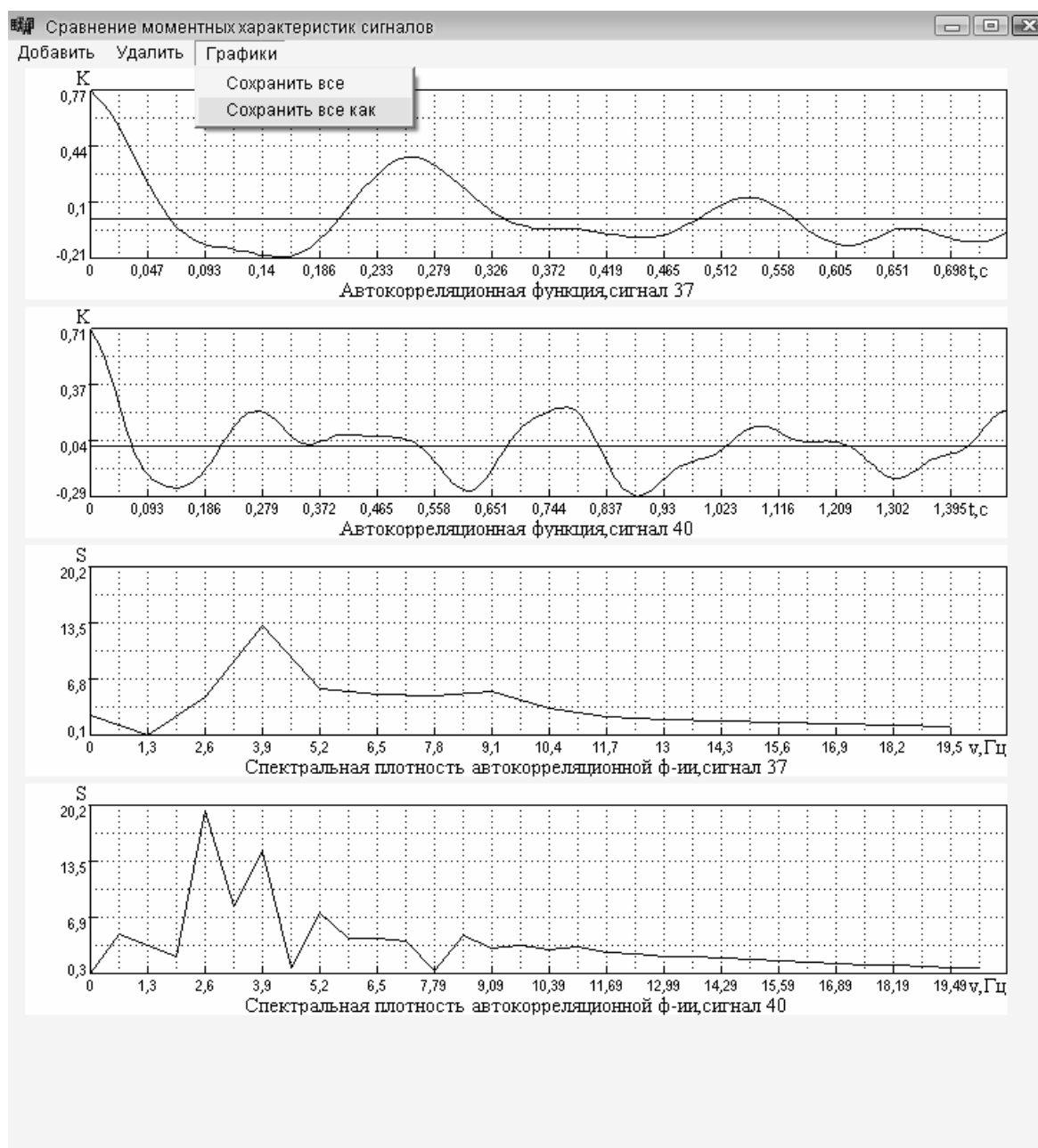


Рис. 4. Общий вид формы сравнения моментных функций

Основные функциональные модули программы следующие. Вычислительный модуль Disperse.cpp – содержит класс CDisperse реализующий математический аппарат моментных характеристик сигналов. Основные атрибуты класса CDisperse: InputF – сечение входного сигнала X_s при $\tau=0$; OutputF – сечение входного сигнала X_t при $\tau=0$; MX – общее математическое ожидание сечения входного сигнала X_s ; MY – общее математическое ожидание сечения входного сигнала X_t ; DX – общая дисперсия сечения входного сигнала X_s ; DY – общая дисперсия сечения входного сигнала X_t ; TMatrix – корреляционная матрица для расчета моментных характеристик; dU – ширина 1 интервала значений сечения входного сигнала X_s (шаг квантования); divs – количество интервалов значений сечения входного сигнала X_s ; kT – длина реализации n; Norm – булево значение, показывает необходимо ли нормализовать результаты расчета; Divisor – параметр равный связанный с количеством временных сдвигов m при расчете моментных характеристик равенством $m=n/\text{Divisor}$; MyFloat – структура используемая для создания динамического списка вещественных чисел, содержит значение вещественного числа val и указатель на следующий элемент списка link.

Модуль Graphic.cpp – содержит класс CGraphic, реализующий построение графиков дискретных функций, заданных вектором значений и шагом дискретизации. Основные атрибуты класса CGraphic: Sx, Dx – начальное значение аргумента функции и приращение аргумента функции (шаг дискретизации); Func – вектор значений функции; ArgCol – количество значений функции; px, py – цена деления (пикселя) в математических единицах; AreaLeft, AreaRight, AreaTop, AreaBottom – параметры определяющие область координатной плоскости, которая будет отображаться на графике; NumbVis, NetVis – булевы значения, указывают на необходимость отображения соответственно числовой шкалы и сетки; NetX, NetY – цена деления (клетки на графике) в математических единицах; NetSlotsX, NetSlotsY – количество клеток по горизонтали и вертикали; mGName – имя графика. Модуль fft.cpp – реализует вычисление спектральной плотности с использованием алгоритма быстрого преобразования Фурье (FFT). Модуль main.cpp – реализует интерфейс пользователя и использует вышеперечисленные модули для расчета и визуализации моментных характеристик. Модуль использует визуально сконструированный класс формы TMainForm. Модуль compare.cpp – функционально связан с main.cpp, реализует вычисление и визуализацию заданной в главной форме информационной характеристики для нескольких выбранных сигналов с целью сравнительного анализа.

При нажатии одной из кнопок построения характеристики происходят вызовы функций: TMainForm::GetOptions, которая сохраняет параметры расчета характеристики в объекте mOptions; TMainForm::DrawSignals модуля main.cpp, которая передает вектор значений входного сигнала в объект mGraphics[0] класса CGraphic и вызывает функцию CGraphic::DrawGraph для отрисовки графика; TMainForm::CalcDrawFunction, которая реализует создание объекта mDisperse класса CDisperse, который в свою очередь реализует матема-

тический аппарат расчета моментных функций. Затем функция инициирует расчет вектора (вызов `mDisperse->CreateDisperse`) значений моментной характеристики и его передачу в объект `mGraphics[1]` класса `CGraphic` для отрисовки.

При помощи функции `PushModel` в объект `mDisperse` передается вектор значений входного сигнала. Затем при помощи функции `TMainForm::ApplyOptions` в объект передаются параметры вычисления моментной характеристики, заданные пользователем в основной форме.

При помощи функций `CDisperse::SetFunction`, `CDisperse::SetArgument`, `CDisperse::SetCenter` устанавливается тип модификации моментной характеристики которую необходимо рассчитать, например для автодисперсионной функции условных асимметрий относительно общей дисперсии $\frac{DA(X_t | X_s)}{DX_t}$

вызываются функции: `mDisperse->SetFunction (CDisperse::GenDisp)` – устанавливается тип моментной функции (дисперсионная); `mDisperse->SetArgument (CDisperse::Assimetry)` – устанавливается тип условной оценки, оценка распределения которой вычисляется (асимметрия); `mDisperse->SetCenter(CDisperse::Disperse)` – устанавливается способ нормирования (значением общей дисперсии для сечения X_t при заданном временном сдвиге). Данные вычислительные функции присваивают значения атрибутам класса `CDisperse`: `Function`, `Argument`, `Center`, которые по сути являются указателями на функции-члены класса `CDisperse`.

Для атрибута `Function` возможны следующие значения: `GenCor` – функция реализует вычисление одного значения автокорреляционной функции сигнала при заданном сдвиге τ ; `GenDisp` – функция реализует вычисление одного значения автодисперсионной функции сигнала при заданном сдвиге τ ; `GenAsm` – функция реализует вычисление одного значения автоасимметрионной функции сигнала при заданном сдвиге τ ; `GenExs` – функция реализует вычисление одного значения автоэксцессионной функции сигнала при заданном сдвиге τ ; `GenFifth` – функция реализует вычисление одного значения моментной функции 5-го порядка функции сигнала при заданном сдвиге τ .

Для атрибута `Argument` в перечне возможны следующие значения: `MExpect` – функция реализует вычисление математического ожидания; `Disperse` – функция реализует вычисление дисперсии; `Assimetry` – функция реализует вычисление асимметрии; `Excess` – функция реализует вычисление эксцесса; `Fifth` – функция реализует вычисление момента 5-го порядка.

Для атрибута `Center` возможны следующие значения: `MExpect` – функция реализует вычисление математического ожидания; `Disperse` – функция реализует вычисление дисперсии; `Assimetry` – функция реализует вычисление асимметрии; `Excess` – функция реализует вычисление эксцесса; `Fifth` – функция реализует вычисление момента 5-го порядка.

Перечисленные функции (`MExpect`, `Disperse`, `Assimetry`, `Excess`, `Fifth`) нахождения статистических оценок используют функцию нахождения центрального момента произвольного порядка `CMoment`. Каждая из них выполнена в трех технических реализациях: для динамического массива вещественных чи-

сел (используется при вычислении общих моментных характеристик сечения сигнала), для динамического списка элементов типа MyFloat (используется при вычислении условных моментных характеристик) и для динамического массива элементов типа MyFloat (используется для нахождения оценки распределения условных моментных характеристик).

При помощи функция CDisperse::SetParam определяются атрибуты объекта mDisperse Norm и Divisor. После установки этих параметров выполняется вызов функции CDisperse::CreateDisperse, в результате работы которой создается вектор значений выбранной моментной характеристики. Если при помощи элемента интерфейса была указана необходимость вычисления спектральной плотности, этот вектор обрабатывается алгоритмом быстрого преобразования Фурье, реализованного в функции FFT модуля fft.cpp. Затем сформированный вектор передается в объект mGraphics[1] для визуализации моментной характеристики. Настройка видового окна объекта mGraphics[1] выполняется с помощью функции CGraphic::InitDefaultCoords, которой передается код выбранного видового окна.

Процесс расчета моментной функции в объекте класса CDisperse, начиная с вызова функции CreateDisperse выполняется следующим образом. Вначале вычисляется количество сдвигов, которые будут использованы для расчета (переменная $\text{maxShift} = \text{kt} / \text{Divisor}$) и выделяется память для массива вычисляемых значений. Далее выполняется maxShift шагов вычисления значений моментной характеристики для каждого временного сдвига от 0 до $m-1$. На каждом шаге вызывается расчетная функция, определяемая атрибутом Function – GenCor, GenDisp, GenAsm, GenExs, GenFifth. Процесс вычисления значения моментной характеристики для фиксированного временного сдвига на примере функции GenDisp пошагово выполняется так. Вначале определяются вектора значений входного сигнала, соответствующего его сечениям X_s и X_t . Используя зависимость $t = s + \tau$, функция Shift записывает в переменные MyInputF и MyOutputF указатели на динамические массивы, соответствующие сечениям X_s и X_t . Далее вычисляется используемое количество значений входного сигнала (переменная $\text{MykT} = \text{kT} - \text{shift}$, где переменная shift – текущий временной сдвиг). Для полученных векторов сечений измеренного сигнала функция GenTMatrix составляет корреляционную матрицу и вычисляет условные статистические оценки для каждого из интервалов сечения X_s . В алгоритме ее работы вначале вычисляется количество интервалов сечения X_s (переменная divs) по стандартной формуле $k = 1 + 3,32 \lg n$, где $\text{divs} = k$, $n = \text{MykT}$, а также ширина интервала (переменная dU). Далее создается вектор значений условных статистических оценок TMatrix (тип оценки определяется атрибутом Argument), типа MyFloat. Количество элементов сформированного вектора равно вычисленному количеству интервалов. Указатель link каждого из элементов этого массива будет указывать на начало списка значений сечения X_t , соответствующих данному интервалу сечения X_s при данном временном сдвиге shift(τ). Далее функция формирует эти списки значений, проходя по вектору значений сечения X_t и распределяя

их соответственно между интервалами сечения X_s (каждому интервалу соответствует 1 элемент вектора TMatrix).

Если в данный момент обрабатываются элемент сечения X_s MyInputF[i] и элемент сечения X_t MyOutput[i] то необходимый интервал сечения X_s определяется по формуле: $u = (MyInputF[i] - \min) / dU$, где переменная u – номер интервала, переменная \min – минимальное значение сечения X_s , переменная dU – ширина уровня. Далее добавляется в динамический список значений элемента TMatrix[u] (который соответствует интервалу u) текущее значение сечения X_t MyOutput[i], соответствующее значению элемента сечения X_s MyInput[i]. По полученным спискам значений вычисляются условные статистические оценки (условное математическое ожидание, условная дисперсия, условная асимметрия, условный эксцесс или условный момент 5-го порядка – в зависимости от атрибута Argument) для каждого интервала. В случае если длина списка для какого-либо интервала равна 0, он исключается из расчета.

Итак, в результате работы функции CDisperse::GenTMatrix формируется вектор значений условных статистических оценок. Далее в функции CDisperse::GenDisp вычисляется оценка распределения этих оценок, используя функцию, заданную атрибутом Function. Нормировав полученное значение с помощью общей моментной характеристики сечения X_t (тип которой определяется атрибутом Center) функция возвращает значение моментной характеристики для текущего количества сдвигов.

Вывод. Разработанное программное обеспечение позволяет произвести оперативную идентификацию технических и технологических состояний барабанных мельниц различных конструкций на основе информационных характеристик моментных функций. Имея данные о наиболее характерных отклонениях технического состояния барабанных мельниц от нормального, например по степени износа футеровки в мельницах самоизмельчения, можно выполнить процедуру раннего обнаружения формируемого предаварийного состояния с соответствующим своевременным его блокированием, исключив значительные экономические потери.

Список литературы

1. Мещеряков Л.И. Идентифікація параметрів об'єктів автоматизованого управління в задачах АСУТП ексцесійними моделями // Сб. науч. тр. Національний гірничий університет. – 2006. – № 24. – С. 182–186.
2. Мещеряков Л.И. Базова форма дисперсійної моделі гірничих технологічних комплексів // Сб. науч. тр. НГАУ. – 2004. – № 20. – С. 209–214.
3. Марюта А.Н., Автоматическая оптимизация процесса обогащения руд на магнито-обогажительных фабриках. – М.: Недра, 1975. – 218 с.
4. Мещеряков Л.И. Системная оценка идентификации барабанных мельниц // Сб. науч. трудов НГАУ. – Днепропетровск, 1998. – Т. 6. – № 3. – С. 255-259.

*Рекомендовано до публікації д.т.н. Слесаревим В.В.
Надійшла до редакції 16.02.10*