В. Анохин

MATLAB для DSP. Идентификация линейных систем

Изучение свойств и особенностей объектов с помощью современных методов обработки информации основывается на построении модели изучаемого объекта по наблюдаемым данным — входным и выходным сигналам. Построение модели объекта по наблюдаемым данным называется идентификацией и включает определение его структуры и параметров. Объектом идентификации может быть устройство, явление или процесс.

Задачи индентификации возникают во многих научных и прикладных сферах человеческой деятельности: медицине, сейсмологии, гидро- и радиолокации, обработке изображений, анализе и синтезе электрических цепей и др. Построение модели начинается с формирования входных воздействий и выбора структуры модели, определяющей взаимосвязь наблюдаемых данных через совокупность параметров. После этого входные воздействия подаются на объект, и измеряются отклики на эти воздействия (выходные сигналы). Затем входные и выходные сигналы и выбранная структура используются для оценки значений параметров в соответствии с принятым критерием качества. Критерий качества идентификации характеризует степень адекватности модели объекту в рамках согласованных допущений и ограничений. Очень часто используется среднеквадратичный критерий, в соответствии с которым ищутся такие оценки параметров, которые обеспечивают минимальный средний квадрат разности выходных сигналов модели и объекта при одном и том же входном воздействии. Оценивание параметров выполняется на основе алгоритма идентификации, определяющего правила поиска оценок. Наконец, для того, чтобы проверить, насколько точно построенная модель имитирует или предсказывает данные наблюдений, необходимо сравнить их при одинаковых воздействиях. Эта процедура называется верификацией модели. Таким образом, для решения задачи идентификации [1-3] необходимо выбрать или сформировать:

- входные сигналы;
- структуру модели изучаемого объекта;
- критерий качества идентификации;
- алгоритм идентификации;
- критерии и методы верификации (подтверждения) модели.

Для понимания и практического использования описываемой ниже графической интерактивной программы ident (по сути дела GUI — graphic user interface), входящей в состав инструментария ident пакета Matlab, необходимо кратко напомнить основные понятия и способы математического описания систем, используемые для построения моделей в процессе работы с этой программой. Изложение всего материала сопровождается простым примером, иллюстрирующим основные положения идентификации и правила работы со средствами Matlab.

Результаты, изложенные в статье, получены в среде Matlab 5.3.

Дискретные модели линейных систем

Подобно тому, как непрерывные линейные системы описываются дифференциальными уравнениями, описание дискретных линейных систем основывается на разностных уравнениях вида

$$\begin{aligned} y(n) = {} & b_1 u(n-1) + \ldots + b_{n_b} u(n-n_b) - \\ & -a_1 y(n-1) - \ldots - a_{n_a} y(n-n_a) \quad (1а) \end{aligned}$$

где последовательности u(n) и y(n) представляют собой входной и выходной сигналы, соответственно; n — индекс времени (здесь и далее для удобства используются обозначения, принятые в документации Matlab). В некоторых приложениях (например, в теории систем управления) разностные уравнения записывают в компактном виде с помощью оператора сдвига q; такая же форма записи разностных уравнений используется и в справочной информации для функций и программ, входящих в состав инструментария ident. Действие оператора сдвига на произвольную последовательность x(n) основано на соотношениях

$$\begin{aligned} qx(n) &= x(n+1), \\ q^{-1}x(n) &= x(n-1), \\ q^{m}x(n) &= x(n+m), \\ q^{-m}x(n) &= x(n-m), \end{aligned}$$

где запись вида qx(n) следует понимать именно как действие оператора, а не как умножение q на x(n).

Если определить операторы A(q) и B(q) в виде полиномов от q:

$$A(q) = 1 + \sum_{i=1}^{n_a} a_i q^{-i},$$

$$B(q) = \sum_{i=1}^{n_b} b_i q^{-i},$$

то действие этих операторов на последовательность x(n) заключается в сдвигах и умножениях отсчётов последовательности на соответствующие весовые коэффициенты $\{a_i\}$ и $\{b_i\}$:

$$\begin{aligned} A(q)x(n) &= \left[1 + \sum_{i=1}^{n_a} a_i q^{-i}\right] x(n) = \\ &= x(n) + a_1 q^{-1}x(n) + a_2 q^{-2}x(n) + \ldots + a_{n_a} q^{-n_a}x(n) = \\ &= x(n) + a_1 x(n-1) + a_2 x(n-2) + \ldots + a_{n_a} x(n-n_a), \end{aligned}$$

$$\begin{aligned} B(q)x(n) &= \left[\sum_{i=1}^{n_b} b_i q^{-i}\right] x(n) = \\ &= b_1 q^{-1}x(n) + b_2 q^{-2}x(n) + \ldots + b_{n_b} q^{-n_b}x(n) = \\ &= x(n) + b_1 x(n-1) + b_2 x(n-2) + \ldots + b_{n_b} x(n-n_b). \end{aligned}$$

Следовательно, уравнение (1а) можно записать в операторном виде следующим образом:

$$A(q)y(n) = B(q)u(n). \quad (1б)$$

Отношение B(q)/A(q), определяющее связь входного и выходного сигналов через операторы сдвига, будем обозначать G(q), то есть

$$G(q) = B(q)/A(q)$$

Следуя [1], в справочной системе Matlab это отношение называют передаточной функцией системы (модели), хотя в классической радиотехнике передаточной функцией называют отношение

$$G(z) = B(z)/A(z),$$

где A(z) и B(z) являются z-преобразованиями последовательностей {a} и {b}:

$$A(z) = \sum_{n=0}^{\infty} a_n z^{-n}$$

$$B(z) = \sum_{n=0}^{\infty} b_n z^{-n}$$

Это замечание читатель должен иметь в виду, когда в последующем к G(q) будет применяться термин “передаточная функция”.

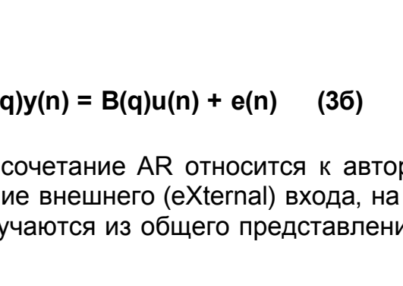

Рис. 1. Модель системы в общем виде

В реальных условиях система подвержена влиянию внешних помех v, однако часто полагают, что действие этих помех можно отобразить аддитивной случайной составляющей сигнала y и являющейся результатом прохождения белого шума e через фильтр с неизвестной передаточной функцией H. Детерминированная же составляющая выходного сигнала формируется с помощью передаточной функции G, определяющей динамические свойства системы. Тогда в общем виде модель системы можно представить так, как показано на рис. 1а, и описать с помощью уравнения

$$y(n) = G(q)u(n) + H(q)e(n), \quad (2)$$

где

$$\begin{aligned} G(q) &= g_1 q^{-1} + g_2 q^{-2} + \ldots, \\ H(q) &= 1 + h_1 q^{-1} + h_2 q^{-2} + \ldots. \end{aligned}$$

Одна из наиболее простых и часто используемых моделей показана на рис. 2а. Она определяет связь между сигналами системы с помощью уравнения

$$\begin{aligned} y(n) = {} & b_1 u(n-1) + \ldots + b_{n_b} u(n-n_b) - \\ & -a_1 y(n-1) - \ldots - a_{n_a} y(n-n_a) + e(n) \quad (3а) \end{aligned}$$

или в операторном виде

$$A(q)y(n) = B(q)u(n) + e(n) \quad (3б)$$

и называется ARX-моделью, где сочетание AR относится к авторегрессионной (AvtoRegressive) части A(q)y(n), а X показывает наличие внешнего (eXternal) входа, на который поступает сигнал u(n). Как видно из уравнений (3), они получаются из общего представления (2) при H(q) = 1/A(q) и G(q) = B(q)/A(q).

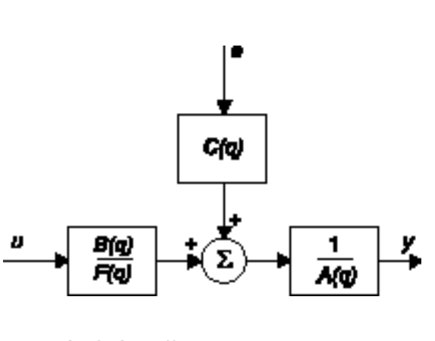

Рис. 2а. Общий вид ARX-модели

Если помеха v(n) моделируется как скользящее среднее белого шума, то получаем ARMAX-модель:

$$\begin{aligned} y(n) = {} & b_1 u(n-1) + \ldots + b_{n_b} u(n-n_b) - \\ & - a_1 y(n-1) - \ldots - a_{n_a} y(n-n_a) + \\ & + e(n) + c_1 e(n-1) + \ldots + c_{n_c} u(n-n_c) \quad (4а) \end{aligned}$$

$$A(q)y(n) = B(q)u(n) + C(q)e(n). \quad (4б)$$

MA-часть (Moving Average) в названии модели указывает на скользящее усреднение C(q)e(n), где

$$C(q) = 1 + \sum_{i=0}^{n_c} c_i z^{-i}$$

Из сравнения (2) и (4) видно, что для ARMAX-модели справедливо равенство H(q) = C(q)/A(q). Эта модель показана на рис. 2б.

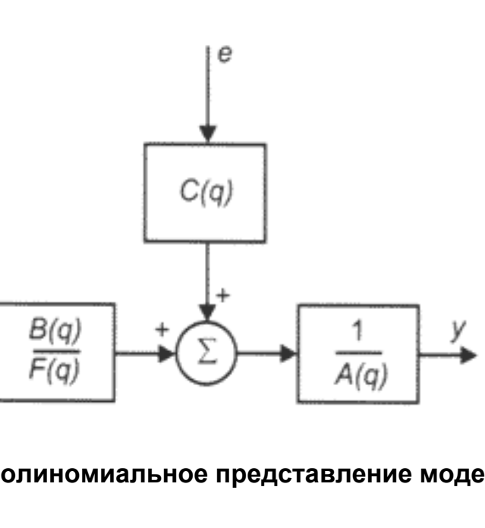

Рис. 2б. Общий вид ARMAX-модели

Общее представление (2) может быть записано в развернутом виде через соответствующие полиномы:

$$A(q)y(n) = \frac{B(q)}{F(q)}u(n) + \frac{C(q)}{D(q)}e(n)$$

Модель (5) показана на рис. 2в. Она включает модели (3) и (4) как частные случаи. Строго говоря, каждое из уравнений (3), (4) и (5) описывает не одну, а множество моделей и называется структурой модели, или модельной структурой. Конкретная модель получается из выбранной структуры, если определить её порядок (порядок соответствующих полиномов A(q), B(q) и так далее) и численные значения параметров (коэффициентов этих полиномов). Таким образом, общая модельная структура (5) включает в себя более простые модельные структуры, например, (3) и (4). В теории идентификации систем [1] множество моделей, описываемое уравнением (5), называется семейством моделей на основе передаточной функции.

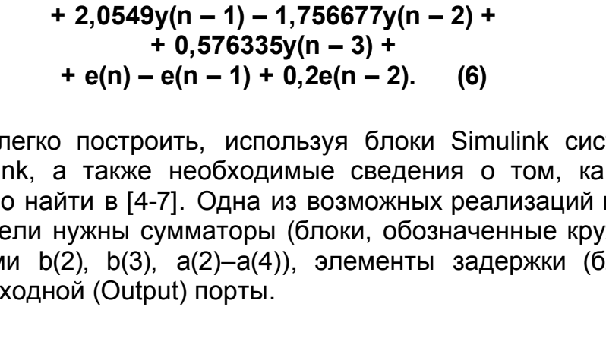

Рис. 2в. Полиномиальное представление модели

Для того, чтобы осуществить идентификационный эксперимент, необходимо выполнить следующие действия:

- построить объект идентификации;
- определить входные сигналы и подать их на вход объекта;
- рассчитать реакцию (выходные сигналы);
- вызвать функцию ident;
- полагая, что нам неизвестны параметры объекта, оценить их по имеющимся входным и выходным сигналам.

Исполнение перечисленных пунктов рассмотрим на конкретном примере.

Объект идентификации и подготовка данных

Пусть дана линейная система, подверженная действию внешнего шума и описываемая разностным уравнением

$$\begin{aligned} y(n) = {} & 0{,}12u(n-1) + 0{,}055752u(n-2) + \\ & + 2{,}0549y(n-1) - 1{,}756677y(n-2) + \\ & + 0{,}576335y(n-3) + \\ & + e(n) - e(n-1) + 0{,}2e(n-2). \quad (6) \end{aligned}$$

Модель такой системы легко построить, используя блоки Simulink системы Matlab. Правила построения моделей Simulink, а также необходимые сведения о том, как задавать параметры используемых блоков, можно найти в [4-7]. Одна из возможных реализаций модели (6) показана на рис. 3. Для построения модели нужны сумматоры (блоки, обозначенные кружками), умножители на константу (блоки с именами b(2), b(3), a(2)–a(4)), элементы задержки (блоки с надписью 1/z), входные (Input и Noise) и выходной (Output) порты.

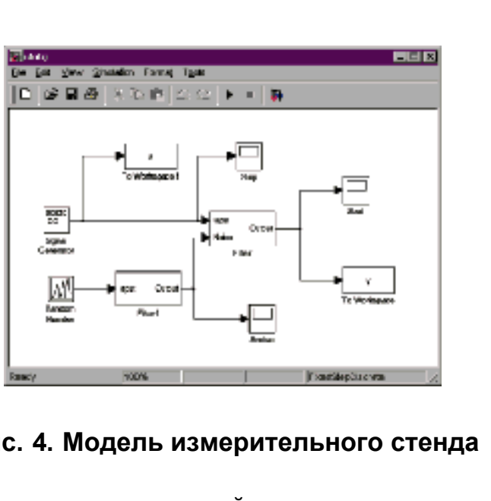

Рис. 3. Реализация модели линейной системы

Перед тем, как к входам построенной модели подключить источники сигналов, а к выходу — регистрирующие блоки, представим модель объекта в виде подсистемы, для чего необходимо выполнить следующие действия:

- позиционировать курсор мыши в левом верхнем углу окна модели;
- нажать левую клавишу мыши и, удерживая её в нажатом положении, переместить курсор в правый нижний угол окна так, чтобы появившаяся в виде прямоугольника область выделения охватила все блоки;
- отпустить удерживаемую клавишу (в результате все блоки в окне будут выделены с помощью прямоугольных чёрных маркеров);
- открыть меню Edit, выделить строку “Create Subsystem” и щёлкнуть левой клавишей мыши.

В результате появится другое изображение модели системы, на котором она представлена как подсистема с подключенными входными и выходными портами.

Далее следует удалить всё, оставив только блок “Subsystem”. Если необходимо, его можно поворачивать на 90°, предварительно выделив его и используя комбинацию клавиш “Ctrl+R”. Раскрыть блок можно, дважды щёлкнув по нему мышью, при этом появится окно с содержимым подсистемы. Чтобы заменить имя блока, присвоенное по умолчанию, щёлкнем по тексту “Subsystem” и наберём текст “Filter”.

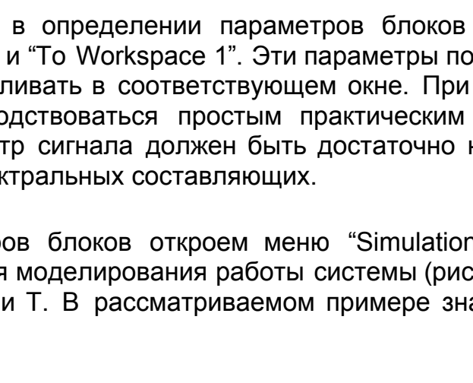

Рис. 4. Модель измерительного стенда

Для завершения процедуры построения полной модели измерительного стенда, подключим к подсистеме Filter необходимые блоки источников и приёмников сигналов, а также фильтр Filter1, на выходе которого формируется шум в соответствии с (6). Построенная модель измерительного стенда показана на рис. 4, а подсистема Filter1 — на рис. 5. Построение подсистемы Filter1 выполняется так же, как и Filter.

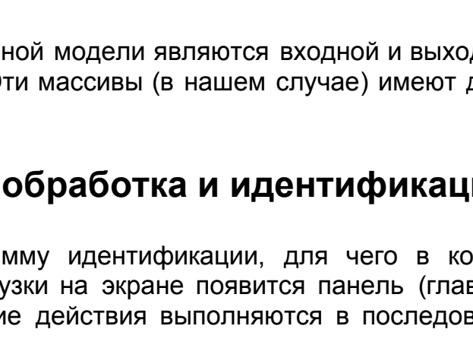

Рис. 5. Модель подсистемы Filter1

Следующий шаг заключается в определении параметров блоков модели “Signal Generator”, “Random Number”, “To Workspace” и “To Workspace 1”. Эти параметры пользователь может выбирать по своему усмотрению и устанавливать в соответствующем окне. При выборе вида и параметров входных сигналов можно руководствоваться простым практическим соображением: в рабочей полосе частот амплитудный спектр сигнала должен быть достаточно насыщенным, то есть иметь как можно больше ненулевых спектральных составляющих.

После определения параметров блоков откроем меню “Simulation”, раздел “Parameters…” и введём данные, необходимые для моделирования работы системы (рис. 6). Теперь осталось только определить период дискретизации T. В рассматриваемом примере значение T = 1e-4 (вводится в командном окне Matlab).

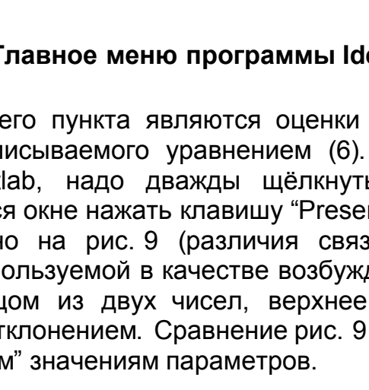

Рис. 6. Задание параметров моделирования

Результатами работы построенной модели являются входной и выходной сигналы, записанные в массивы u и y, соответственно. Эти массивы (в нашем случае) имеют длину 2001 и сохраняются в рабочем пространстве Matlab.

Предварительная обработка и идентификация параметров

Загрузим графическую программу идентификации, для чего в командном окне необходимо набрать ident. В результате загрузки на экране появится панель (главное окно) программы ident, показанная на рис. 7. Дальнейшие действия выполняются в последовательности, изложенной на вставке.

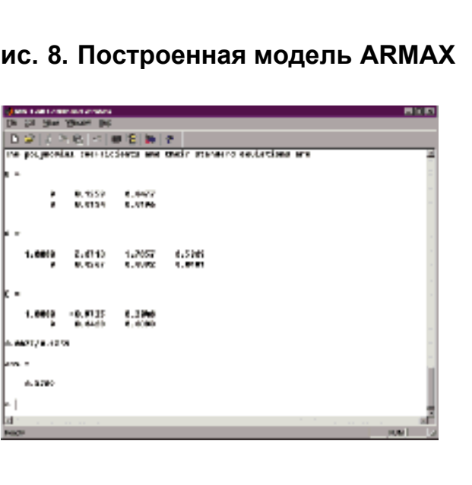

Рис. 7. Главное меню программы Ident

Результатом выполнения последнего пункта являются оценки a, b и c параметров a, b и c ARMAX-модели (4) для объекта, описываемого уравнением (6). Для того, чтобы эти оценки отобразить в командном окне Matlab, надо дважды щёлкнуть правой клавишей мыши по пиктограмме amx3221 и в открывшемся окне нажать клавишу “Present”. Окно Matlab будет выглядеть приблизительно так, как изображено на рис. 9 (различия связаны с разными реализациями случайной последовательности e, используемой в качестве возбуждающего шума). Оценка каждого из параметров представлена столбцом из двух чисел, верхнее из которых является оценкой среднего, а нижнее — стандартным отклонением. Сравнение рис. 9 и уравнения (6) показывает, что полученные оценки близки к “истинным” значениям параметров.

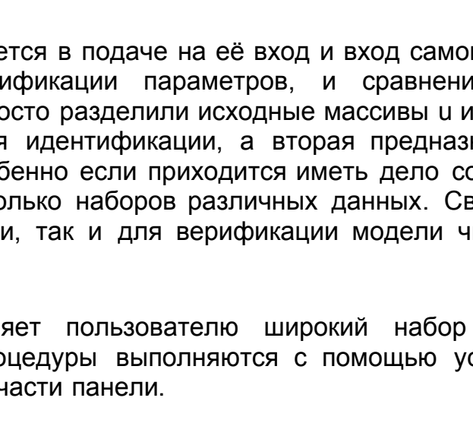

Рис. 8. Построенная модель ARMAX

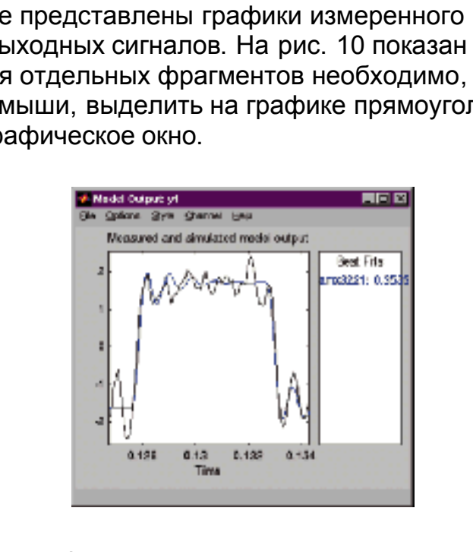

Рис. 9. Данные "оценки" параметров ARMAX-модели

Однако на практике эти “истинные” значения неизвестны, и для того, чтобы убедиться, насколько точно построенная модель описывает объект, необходимо проанализировать её поведение. Эта процедура называется подтверждением или верификацией модели.

Верификация модели

Верификация модели заключается в подаче на её вход и вход самого объекта данных, которые не использовались для идентификации параметров, и сравнении выходных сигналов. В рассматриваемом примере мы просто разделили исходные массивы u и y на две половины, одна из которых была использована для идентификации, а вторая предназначена для подтверждения модели. В реальной работе, особенно если приходится иметь дело со сложными объектами, для верификации используются несколько наборов различных данных. Сведения по выбору входных сигналов, как для идентификации, так и для верификации модели читатель может получить из монографии [1].

Программа ident предоставляет пользователю широкий набор средств для анализа и подтверждения модели. Эти процедуры выполняются с помощью установки флажков в окнах, расположенных в правой нижней части панели.

1. Для сравнения наблюдаемых и моделируемых выходных сигналов установим флажок “Model output”. В открывшемся окне представлены графики измеренного (линия чёрного цвета) и рассчитанного по модели выходных сигналов. На рис. 10 показан увеличенный фрагмент этих сигналов. Для рассмотрения отдельных фрагментов необходимо, удерживая в нажатом положении левую клавишу мыши, выделить на графике прямоугольную область, которая будет отображена на всё графическое окно.

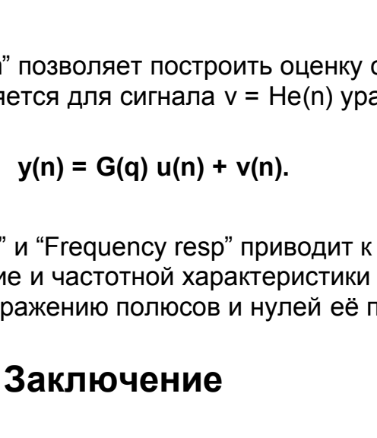

Рис. 10. Сравнение наблюдаемых и моделируемых выходных сигналов

Меню Options позволяет пользователю строить либо сами сигналы, как это сделано выше, либо их разность (пункт меню “Error plot”), устанавливать a-процентный доверительный уровень и строить доверительные интервалы (пункты “Set confidence level” и “Show a% confidence intervals”), строить одношаговые и многошаговые прогнозы выходных процессов (“Set prediction horizon” и “n step ahead predicted output”). В правой части окна указана среднеквадратичная ошибка моделирования, рассчитываемая по формуле

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}\left(y(n) - \hat{y}(n)\right)^2}$$

где N — объём выборки (всей по умолчанию). Параметр N можно менять выбором пункта “Customized time span for fit…” меню “Options”.

2. Флажок “Model resids” устанавливается для выполнения статистического анализа моделируемых данных с использованием информации, находящейся в области “Walidation Data”. Анализ статистических свойств моделируемых данных предполагается рассмотреть в последующих публикациях.
3. Установка флажка “Noise Spectrum” позволяет построить оценку спектральной плотности мощности шума, которая определяется для сигнала v = He(n) уравнения (2):

$$y(n) = G(q)\,u(n) + v(n).$$

4. Установка флажков “Transient resp” и “Frequency resp” приводит к построению реакции на единичное ступенчатое воздействие и частотной характеристики модели, соответственно, флажка “Zeros and poles” — к отображению полюсов и нулей её передаточной функции.

Заключение

Идентификация систем (включая построение моделей временных рядов) находит всё большее применение в различных приложениях. Для использования результатов теории идентификации на практике необходимы знания в различных научных и прикладных областях (регрессионного, корреляционного и спектрального анализа, планирования эксперимента, моделирования систем и др.). В рамках нашей статьи рассмотрены лишь основные средства и возможности для решения задач идентификации, предоставляемые графической интерактивной программой ident, однако и приведённое рассмотрение позволяет отметить следующее.

Простая на первый взгляд процедура идентификации динамической линейной системы, описанная в настоящей статье, на самом деле включает решение целого ряда задач (оптимизации, корреляционного и регрессионного анализа и др.), которые автоматически решаются по мере необходимости путём вызова соответствующих функций Matlab.

В зависимости от объекта идентификации (системы линейные и нелинейные, с одним и многими входами и выходами, с постоянными и переменными параметрами и так далее), значительно изменяется количество, разнообразие и сложность таких “внутренних” задач, однако для пользователя это практически незаметно.

В программе ident экспорт и импорт данных являются тривиальными процедурами, что позволяет легко сочетать работу этой программы с другими программами Matlab (в нашем случае — с моделью Simulink).

Таким образом, простота, гибкость и универсальность делают описанную программу ident одинаково полезной как для начинающих пользователей, так и для опытных специалистов в области моделирования и идентификации.

Литература

1. *Льюнг Л. Идентификация систем. — М.: Наука. Гл. ред. физ.-мат. лит. — 1991. — 432 с.*
2. *Цыпкин Я.З. Основы информационной теории идентификации. — М.: Наука. — 1984. — 320 с.*
3. *Современные методы идентификации /Под ред. П. Эйкхоффа. — М.: Мир. — 1983. — 400 с.*
4. *Дьяконов В.П., Абраменкова И.В. Matlab 5.0/5.3. Система символьной математики. — М.: Нолидж. — 1999. — 633 с.*
5. *Гультяев А.К. Имитационное моделирование в среде Windows. — СПб.: КОРОНА принт. — 1999. — 288 с.*
6. *Анохин В.В. **Моделирование аналого-цифрового преобразования. Часть 1 // Chip News. — 2000. — № 2. — С. 4–7.***
7. *Анохин В.В. **Моделирование аналого-цифрового преобразования. Часть 2 // Chip News. — 2000. — № 3. — С. 26–29.***
8. *Семенов Н. **Моделирование приемника DTMF // Chip News. — 2000. — № 4, С. 48–52. — № 5, С. 12–16.***
9. *Геппенер В., Ланнэ А., Черниченко Д. **Использование GUI Wavemenu для решения инженерных задач // Chip News. — 2000. — № 6, С. 2–8. — № 7, С. 16–19.***

E-mail: arturlan@robotek.ru

© 2001 "CHIP NEWS".
Разработка сайта: