УДК 519.711

Ю.И. Параев

ВЫЧИСЛИТЕЛЬНАЯ СИСТЕМА «УПРАВЛЕНИЕ»

Приводится описание вычислительной системы «Управление», разработанной для проведения лабораторных работ по курсу «Теория управления» на персональных компьютерах типа PC IBM. Эта система может также широко использоваться в научных исследованиях в области теории автоматического управления и смежных вопросах.

Вычислительная система «Управление» (ВСУ) первоначально разрабатывалась для проведения лабораторных, курсовых и дипломных работ по курсу «Теория управления» на персональных компьютерах типа PC IBM и совместимых с ними. Позднее она широко использовалась и в научных исследованиях.

Работы по указанному курсу связаны с исследованием динамических систем управления, поведение которых задается в пространстве состояний, т.е. описывается дифференциальными или разностными уравнениями, в правые части которых кроме фазовых координат могут входить управляющие и случайные воздействия. Кроме того, имеется наблюдаемый или измеряемый процесс, который связан с управляемым процессом и несет информацию о его текущем состоянии. Процесс наблюдений или измерений может сопровождаться случайными ошибками или помехами в измерительном канале.

Тематика работ связана с решением таких задач, как

– моделирование систем (в том числе и стохастических),

– исследование систем на устойчивость,

– синтез регуляторов и компенсаторов различных типов (ПИД, оптимальных, динамических, модальных и т.п.),

– синтез наблюдателей и фильтров (Калмана, Луенбергера и т.п.),

– частотные методы исследования.

В математическом отношении эти задачи связаны с решением дифференциальных или разностных уравнений, решением матричных уравнений Ляпунова, Риккати, Сильвестра, вычислением определителей Гурвица и рядом других вычислительных задач.

ВСУ позволяет проводить эти вычисления в диалоговом режиме Все числовые данные оформляются в виде матриц, которым присваиваются свои имена, причем размерности матриц, в которые заносятся результаты счета, определяются автоматически. Размерности матриц могут быть произвольными. Пользователь работает только с кодами операций и именами матриц и векторов, причем в режиме «подсказки» коды операций вводятся автоматически. Пользователь имеет дело только с формулами уравнений, которые определяют решение той или иной задачи. Поскольку ВСУ содержит большой набор (около 250) операций линейной алгебры и операций, связанных с решением алгебраических, дифференциальных и разностных уравнений, то она может использоваться при изучении других учебных дисциплин и проведении научных исследований, математически связанных с указанными процедурами.

ВСУ содержит основные группы операций:

– операции управления системой, в том числе установка параметров вычислительных процедур, включая машинный ноль;

– операции ввода различных матриц, в том числе случайных:

– операции компоновки и формирования матриц: перестановка, удаление, выделение, дублирование блоков столбцов или строк, компоновка матриц из блоков, вписывание одной матрицы в другую, формирование матриц управляемости и наблюдаемости, выделение линейно независимых строк и столбцов и т.д.;

– арифметические операции над матрицами: сложение, вычитание, умножение, обращение, вычисление квадратного корня из симметрической матрицы и т.п.; поэлементные арифметические операции;

– вычисление характеристик матриц: ранга, определителя, главных миноров, характеристического многочлена, собственных чисел и векторов и т.д.;

– приведение матриц к каноническим формам – жордановой, естественной, Шура, Хессельберга и др., нахождение инвариантных многочленов матриц;

– решение алгебраических уравнений (включая псевдообращение), решение матричных прямых и обратных уравнений Ляпунова и Риккати, уравнения Сильвестра для непрерывных и дискретных систем;

– решение дифференциальных и разностных уравнений, в том числе и стохастических при случайных воздействиях гауссовского и пуассоновского типов;

– приведение непрерывных систем управления к дискретным;

– операции с многочленами: сложение и перемножение многочленов, введение биномиальных многочленов и многочленов Батерворда, вычисление матрицы Рауса и определителей Гурвица для многочлена, вычисление корней многочлена, вычисление матричных многочленов;

– вычисление определенных интегралов и табуляция функций (в том числе матричных);

– ввод функций, определяемых пользователем;

– частотные методы исследования систем управления: вычисление годографа Михайлова, амплитудных и фазовых частотных характеристик, перемножение и сложение передаточных функций, вычисление передаточной функции для замкнутой системы, построение передаточной функции для динамической системы и обратная задача;

– представление графического материала, в том числе построение фазовых портретов.

В системе предусмотрен программный режим, позволяющий составлять из операций программы, которым могут присваиваться свои имена и которые затем могут выполняться автоматически. Все выполненные в процессе диалога операции запоминаются и, таким образом, автоматически получается готовая программа. Имеется возможность включения в программы команд условного выполнения операций и команд условного и безусловного перехода по меткам, а также организация циклов. Предусмотрено включение одних программ в другие. Всего возможна организация до 20 программных уровней. Программный режим позволяет одновременно просчитывать несколько вариантов задачи с изменением какого-либо параметра. Для набора и отладки программ в системе предусмотрен специальный редактор. Предусмотрена операция автоматического изменения в программах одних имен матриц на другие, что позволяет использовать имеющиеся программы как стандартные, т.е. использовать их при решении разных задач.

Числовые результаты и созданные программы можно записывать в файлы и, таким образом, создавать второй уровень системы. Система позволяет обрабатывать любые числовые массивы, записанные в кодах ASCII.

По функциональным возможностям и интерфейсу система «Управление» несколько напоминает универсальную интерпретирующую среду матричных вычислений MatLab («Матричная лаборатория»). Но в отличие от системы MatLab система «Управление» специально ориентирована на решение задач моделирования, управления и обработки информации в динамических системах, описываемых в пространстве состояний. Поэтому она имеет значительно большее число встроенных специальных операций для данной предметной области.

Более того, организация системы позволяет включать в нее новые операции без каких-либо затруднений, поэтому она постоянно совершенствуется. Удачный интерфейс системы «Управление» и простая организация программного режима существенно упрощают работу пользователя.

1. Организация системы

1.1. Запуск

1.1.1. ВСУ работает автономно в среде MS DOS (или WINDOWS). Имеется возможность работы в сети с общим жестким диском.

1.1.2. В файле DIM.BAS содержатся параметры системы: величина числового массива (не может превышать 16380), максимальное число используемых матриц, максимальное число программ, максимальное число программных уровней и т.д. Эти параметры в файле DIM. BAS можно изменять с учетом возможностей компьютера (в частности, с величиной строковой памяти). Текущее количество свободной памяти указывается в конце списка введенных матриц (операция ctm).

1.1.3. Причина сбоев при запуске или работе:

– нехватка строковой памяти – внесите изменения в файл DIM.BAS, уменьшив число матриц, максимальное число операций в программах, максимальное число программ, максимальное число программных уровней и т.д., и затем снова запустите ВСУ;

– нехватка зарезервированных массивов для числа матриц, программ и операций – удалите лишние матрицы;

– при работе в сети с общим жестким диском в файле DIM.BAS неправильно задан соответствующий параметр.

1.1.4. При работе может создаваться файл ERR. DAT, в который заносятся случившиеся ошибки (с указанием их кода и соответствующей операции). Его можно удалять в любое время.

1.1.5. После запуска на экране появляется заставка, в которой кратко изложены общие правила работы с системой. Этого достаточно даже для начинающего пользователя. Признаком готовности системы к работе или к выполнению следующей операции служит появление на экране надписи «ОПЕРАЦИЯ:».

1.2. Организация данных

1.2.1. Все числовые данные оформляются в виде матриц порядка $n \times m$. Векторам соответствуют матрицы порядка $n \times 1$ или $1 \times n$, отдельным числам – матрицы порядка 1×1. Размерности исходных матриц определяются при их вводе. Размерности матриц, в которые заносятся результаты счета, определяются автоматически. Размерности матриц могут быть произвольными. Нужно только учитывать, что величина числового массива не должна превышать 16000 чисел. Максимальное количество различных матриц задается в файле DIM.BAS и практически не должно превышать 1000.

Матрицам присваиваются имена, которые могут состоять из латинских букв, цифр и символов (не допускаются ЗАПЯТАЯ и символы +, –, /, \, :, ;, ^, *, (,), [,], !, так как они используются в арифметических операциях). Лучше имена матриц начинать с букв. Это же относится и к именам программ.

1.2.2. Многочлен

$$f(\lambda) = a_{n+1}\lambda^n + a_n\lambda^{n-1} + \ldots + a_2\lambda + a_1,$$

где $a_{n+1} \neq 0$, задается в виде вектора. В разных операциях используются две формы многочленов и соответствующих векторов. Многочлен называется нормированным (сопровождается отметкой [n]), если ему соответствует n-мерный вектор a, составленный из чисел $a_1, a_2, \ldots, a_n$. При этом предполагается, что $a_{n+1} = 1$. Так, например, представляется характеристический многочлен матрицы. Многочлен называется ненормированным (далее в тексте и в меню подсказки сопровождается отметкой [p]), если ему соответствует n + 1-мерный вектор, составленный из чисел $a_1, a_2, \ldots, a_n, a_{n+1}$. Нужно обратить внимание на порядок нумерации коэффициентов многочлена и элементов вектора. Если в какой-то операции возможно использование обеих форм задания многочленов, то ставится отметка [s]. С помощью операции lnp можно переходить от одной формы представления многочленов к другой.

1.2.3. Передаточная функция

$$W(p) = \frac{b_1 + b_2 p + \ldots + b_n p^{n-1} + b_{n+1} p^n}{a_1 + a_2 p + \ldots + a_n p^{n-1} + a_{n+1} p^n},$$

где в числителе или знаменателе коэффициенты при старших степенях p могут равняться нулю, но не одновременно, задается с помощью матрицы порядка $2 \times (n+1)$ вида $W = \begin{bmatrix} b_1 & b_2 \cdots b_n & b_{n+1} \\ a_1 & a_2 \cdots a_n & a_{n+1} \end{bmatrix}$.

Передаточная функция считается физически реализуемой, если степень многочлена в числителе меньше степени многочлена в знаменателе, т.е. по крайней мере $b_{n+1} = 0$, но $a_{n+1} \neq 0$.

1.3. Диалоговый режим

1.3.1. Для работы с системой после появления на экране надписи «ОПЕРАЦИЯ:» следует ввести в виде одного слова саму операцию. Общая структура этого слова следующая:

<u>КОД операции</u> <u>фрагмент1</u> <u>фрагмент2</u> … .

Если пользователь не знает кода операции и структуру слова, то он может получить подсказку в МЕНЮ ОПЕРАЦИЙ, куда можно войти с помощью ENTER. Вообще, при всех затруднениях нажимайте ENTER!

Код операции состоит из трех латинских букв. В меню коды операций разбиты на группы по смысловому признаку. Коды операций из одной группы начинаются с одинаковых букв. Выбрав группу операций и нажав ENTER, попадаем в список операций данной группы. Выбрав нужную операцию и нажав ENTER, получаем окно для набора слова. В этом окне приводится шаблон структуры слова. Если пользователь знает код операции и шаблон структуры слова, то он может вводить это слово сразу после надписи «ОПЕРАЦИЯ:», не входя в МЕНЮ.

После введения кода следует ввести необходимые фрагменты слова, между которыми всегда должен стоять ПРОБЕЛ. Внутри фрагмента пробелы недопустимы. Буквы, набранные в верхнем и нижнем регистрах, считаются разными. Исключение относится только к кодам операций и именам стандартных функций.

В меню операций указываются необходимые аргументы операции. Эти аргументы обозначаются прописными (большими) или строчными (малыми) буквами. Большими буквами отмечены аргументы, которые должны быть именами матриц или для скалярных задач – числами. При обработке каждого фрагмента сначала проверяется наличие указанной матрицы, а если ее нет, то проверяется – является ли фрагмент числом. Если окажется, что фрагмент не является ни тем, ни другим, то выдается сообщение об ошибке «Матрицы нет».

Строчными буквами обозначаются скалярные параметры, которые можно вводить в виде каких-то арифметических выражений. Эти арифметические выражения могут содержать элементы имеющихся матриц, числа и стандартные или вводимые пользователем функции. В арифметических выражениях можно использовать символы +, –, *, /, ^ и круглые скобки ( ). Квадратные скобки используются только для обозначения аргументов стандартных и вводимых функций. Элементы матриц указываются в виде $A(i, j)$, элементы векторов – в виде $A(i)$. Исключением являются элементы $A(1, 1)$ или $A(1)$, когда достаточно указать только имя матрицы. Для вычисления индексов i и j допустимо также использовать арифметические выражения. При этом при получении дробных значений они округляются до ближайшего целого снизу.

При описании операций, а также в подсказках необязательные имена, выражения или признаки заключены в СКОБКИ. Если при вводе операции такие параметры не указываются, то они принимают значения по умолчанию. При вводе операции СКОБКИ не указываются! Признак / указывает на возможность выполнения другого варианта операции. В МЕНЮ ОПЕРАЦИЙ некоторые операции сопровождаются краткими комментариями, в которых поясняются особенности операции.

1.3.2. Одним словом можно ввести выполнение нескольких одинаковых операций, разделив группы матриц с помощью знака «;» –

КОД фрагменты; фрагменты и т.д.

(например, при msm *X A B*; *X X C* вычисляется сумма $X=A+B+C$), или разных операций, разделив их знаком «:» –

КОД1 фрагменты; фрагменты; ...

:КОД2 фрагменты; фрагменты и т.д.

(например, при mpr *x A B*; *y C D*: msm *X x y* вычисляется $X=AB+CD$). Это удобно при составлении программ.

1.3.3. В операциях первой (или первыми) стоит матрица, в которую заносится результат вычислений. Ниже при описании операций в большинстве случаев такие матрицы обозначаются через *X*, *Y*, *Z*. Имя этой матрицы может совпадать с именем какой-либо уже введенной матрицы, в том числе и с используемой в данной операции. В таком случае последняя матрица удаляется. Это удобно для удаления лишних матриц.

1.3.4. В системе имеются следующие стандартные функции: abs[ ] – абсолютное значение, ath[ ] – арктангенс, cos[ ] – косинус, exp[ ] – экспонента, fix[ ] – удаление десятичной части числа, int[ ] – округление до целого, log[ ] – натуральный логарифм, sgn[ ] – знаковая функция, sin[ ] – синус, sqr[ ] – квадратный корень, tan[ ] – тангенс.

Можно ввести дополнительные функции с помощью операции ppn. Вводимые функции должны обязательно начинаться с буквы *F*. Аргументы функций могут быть любыми арифметическими выражениями, включая любые стандартные и введенные функции. Все эти арифметические выражения не должны содержать ПРОБЕЛЫ, иначе они будут восприниматься как разные фрагменты. При обращении к этим функциям допустимо в аргумент вставлять другие функции, а также и саму функцию. Например, допустимо выражение *F*[*F*[*x*+1]]. С помощью операции ppf эти введенные функции можно оформить в программу, чтобы сохранить их для последующей работы.

1.3.5. В диалоговом режиме активны следующие функциональные клавиши (клавиши, активные в данный момент, высвечиваются в нижней строке экрана).

F1 – «Матрицы». При нажатии на F1 появляется окно, где приводится список всех введенных матриц. Выбрать нужную матрицу можно с помощью маркера. При этом становятся активными следующие клавиши:

F5 – «Переименование матриц» – изменение имени матрицы;

F6 – «Изменение матриц» – можно изменить какой-либо элемент матрицы;

F7 – запись выделенных матриц в файл;

F8 – удаление выделенных матриц.

Выделение нужных матриц, программ или файлов осуществляется с помощью INS. Все матрицы можно выделить с помощью правого + .

F2 – «Программы». При нажатии на F2 появляется окно, где приводится список всех введенных программ. При этом становятся активными следующие клавиши:

F5 – «Переименование программ» – изменение имени программы;

F6 – «Изменение программ» – переход в редактор программ;

F7 – запись выделенных программ в файл;

F8 – удаление выделенных программ;

F9 – запуск программы.

Если при работе какие-либо файлы считывались, то после нажатия F7 кроме надписи «Укажите имя файла» внизу экрана высвечиваются имена считанных ранее файлов. При нажатии соответствующей клавиши F происходит запись в указанный файл, т.е. в таком случае не нужно создавать новый файл.

F3 – « Числовые файлы». При нажатии на F3 появляется окно, где приводится список всех имеющихся файлов, содержащих числовые массивы (см. ниже). Выбрав нужный файл, можно считать его с помощью ENTER, причем в ДОПОЛНЕНИЕ к имеющемуся массиву. Кроме того, становятся активными следующие клавиши:

F5 – «Переименование» – изменение имени файла;

F6 – «Считывание файла» вместо имеющегося массива, т.е. имеющиеся до считывания матрицы удаляются;

F8 – удаление выделенных файлов;

F10 – введение новой директории.

F4 – «Программные файлы». При нажатии на F4 появляется окно, где приводится список всех имеющихся файлов, содержащих массивы программ (см. ниже). Выбрав нужный файл, можно считать его с помощью ENTER, причем в ДОПОЛНЕНИЕ к имеющемуся массиву. Кроме того, становятся активными следующие клавиши:

F5 – «Переименование» – изменение имени файла;

F6 – «Считывание файла» вместо имеющегося массива, т.е. имеющиеся до считывания программы удаляются;

F8 – удаление выделенных файлов;

F10 – введение новой директории.

После введения кода операции активными остаются клавиши F1 – «Матрицы» и F2 –«Программы». С помощью этих клавиш можно войти в список матриц или программ. Выбрав с помощью маркера нужное имя, можно, нажав ENTER, ввести это имя в набираемое слово.

1.4. Графический режим

Числовые массивы, задаваемые в виде матриц, можно отразить на экране в виде графиков. Операции построения графиков имеют следующий вид: g** имена матриц (необязательные параметры или признаки). Вторыми буквами кода операции могут быть k, p, l, которые определяют расположение графика на экране: в центре, в правой или в левой части экрана соответственно. Третьи буквы в коде операции g, f, s определяют тип графика.

g – собственно ГРАФИК. Числа откладываются по вертикали, причем элементы каждой строки каждой матрицы располагаются равномерно по горизонтали в интервале [0,1] и соединяются ломаной. Если после набора операции набрать /, то кривая аппроксимируется горизонтальными отрезками. Каждой строке матрицы соответствует своя кривая. Возможно наряду с именами матриц указывать конкретные числа. Например, glg *A B* 5,55. Для изменения интервала [0, 1] на интервал [0, *t*] нужно это сделать либо при вводе операции, набрав в конце слова через пробел = *t*, либо после построения графика нажать F6 и затем ввести *t*. При этом *t* может быть либо нужным числом, либо каким-то выражением.

f – ФАЗОВЫЙ ПОРТРЕТ. Первый элемент каждого столбца каждой матрицы откладывается по горизонтали, а остальные – по вертикали. Элементы каждой строки (начиная со второй) соединяются ломаной. Если после набора операции набрать /, то кривая аппроксимируется горизонтальными отрезками. Каждой строке матрицы (кроме первой) соответствует своя кривая. Выводимые матрицы должны содержать не менее двух строк.

s – СПЕКТР матрицы. Выводимые матрицы должны состоять только из двух столбцов. Элементы первого столбца каждой матрицы откладываются по горизонтали, а соответствующие элементы второго столбца – по вертикали.

Максимальное число выводимых кривых равно 8. Поэтому для вывода на экран массива, содержащего большее число строк, его нужно разбить на блоки. Можно менять порядок вывода строк. Для этого в режиме ГРАФИК после имен матриц нужно набрать \ и номера строк (все без пробелов) от 1 до 9. Например, при *A*\934 на экран выводятся 9, 3 и 4-я строки матрицы *A*. В режиме ФАЗОВЫЙ ПОРТРЕТ после имен матриц можно набрать (без пробелов) \ и далее номер строки, элементы которой откладываются по горизонтали, а затем номера остальных строк (все от 1 до 9). Если после \ указать только один номер, то остальные строки выводятся в порядке возрастания номеров. Например, при *A*\934 по горизонтали откладываются элементы 9-й строки, а на экран выводятся строки 3-я и 4-я; при *A*\3 по горизонтали откладываются элементы 3-й строки, а на экран выводятся строки 1, 2, 4 и т.д.

Неправильно заданные строки игнорируются.

Для операций ГРАФИК и ФАЗОВЫЙ ПОРТРЕТ предусмотрено введение меток для разных кривых. Для введения меток нужно нажать F5 и набрать номер кривой и ENTER. Перемещение меток осуществляется клавишами ← и →, фиксация меток – клавишей ENTER. Одновременно внизу экрана указываются значения координат точки. Выход из режима меток осуществляется с помощью ENTER. В операции ФАЗОВЫЙ ПОРТРЕТ нужно помнить, что первой кривой соответствует вторая строка матрицы и т.д.

Клавиши F1, F2, F3, F10 включают и выключают цвет, сетку, рамку и подписи на рисунке соответственно. Под графиками после нажатия F9 можно набрать текст (не более двух строк).

Переход к выполнению следующей операции осуществляется клавишей ENTER. Если далее вводится графическая операция (начинающаяся с буквы g) без обращения к подсказке, то система остается в графическом режиме. Это позволяет строить графики в правой и левой частях экрана.

Изображение, полученное на экране, может быть отпечатано на принтере или записано в буфер с помощью PrintScreen.

1.5. Программный режим

В системе предусмотрена возможность из выполненных операций составлять программы и работать с этими программами. Каждой программе присваивается свое имя. Операция pio позволяет оформить уже выполненные операции в новую программу. Для набора и отладки программ в системе предусмотрен специальный редактор (см. ниже), хотя это можно делать и с помощью обычных редакторов (NORTON и т.п.). Переход в режим набора и отладки программ осуществляется с помощью операций pip, pio, pvp или клавиш F2 и F6.

В одну программу можно вставлять другие программы (кроме ее самой). В результате возможна организация нескольких программных уровней (число таких уровней устанавливается в файле DIM. BAS). С помощью операции prn можно выполнять программу несколько раз. Таким образом можно организовывать циклические вычисления. С помощью операции ppp можно просчитывать несколько вариантов одной задачи при варьировании какого-либо параметра. Имеется возможность автоматического переименования имен матриц в программе, что позволяет создавать библиотеку стандартных программ.

1.6. Редактор для отладки программ

Вход в редактор осуществляется с помощью операций pip, pio, pvp или клавиш F2 и F6. Выход из редактора – с помощью Esc или Ctrl+Q. Клавиши ENTER, Backpace, Home, End, DEL, INS, PageUp, PageDown, Ctrl+PageUp, Ctrl+PageDown имеют обычные функции.

F3 – переход в МЕНЮ ОПЕРАЦИЙ, где можно выбрать и набрать нужную операцию, которая затем поместится на место, указанное маркером.

F8 – включение и выключение INS.

F4, F5 – выделение блока строк (F4 – начало, F5 – конец).

ENTER или Ctrl+M – разделение строки (в режиме вставки).

Ctrl+Y – удаление строки.

Ctrl+F – запись программы в файл,

Ctrl+N – изменение имен матриц – можно автоматически изменить имена имеющихся в программе матриц или программ на новые.

Ctrl+G – изменение имени редактируемой программы.

Ctrl+K – ввод или изменение комментария к программе.

Ctrl+P – считывание другой программы на место, указываемое маркером.

Ctrl+X – копирование выделенного блока на место, указываемое маркером.

Ctrl+V – перестановка выделенного блока на место, указываемое маркером.

Ctrl+W – удаление выделенного блока.

При выходе из редактора проводится проверка правильности набранных операций: проверяется правильность кода и числа обязательных аргументов. Из редактора можно выйти лишь тогда, когда в программе нет ошибок. Без исправления ошибок можно выйти из редактора, если в начале каждой строки с ошибками поставить знак ', т.е. превратив строку в комментарий.

1.7. Работа с файлами

1.7.1. Система работает с файлами двух типов: файлами, содержащими числовые массивы (им по умолчанию присваивается расширение <d-c>), и с файлами, содержащим массивы программ (им по умолчанию присваивается расширение <d-p>). При записи и считывании таких файлов расширение указывать не следует. Имена файлов не должны содержать знак + (это стандартное ограничение в MS DOS).

1.7.2. Организация файлов.

В числовых файлах для каждой матрицы порядка $n \times m$ отводится строка, имеющая вид:

+ <u>имя матрицы</u> *n m* ' <u>комментарий</u>,

после которой следуют числа. Если комментарий отсутствует, то ' можно не указывать. В самой системе в качестве комментария записывается операция, с помощью которой получена данная матрица, однако в редакторе этот комментарий можно изменять.

В программных файлах для каждой программы, состоящей из n операций, отводится n+1 строка. Первая из этих строк имеет вид: + <u>имя программы</u> ' <u>комментарий</u>. В остальные n строк заносятся операции в виде <u>операция</u> ' <u>комментарий.</u>

Если комментарии отсутствуют, то ' можно не указывать. Допустимы строки с одним комментарием (знак ' обязателен). В самой системе комментарий для программы вводится пользователем, а в качестве комментариев для операций записываются их расшифровки. Метки отмечаются с помощью * и устанавливаются перед соответствующей операцией. Например, **aa* ima *A n m*.

Можно проводить изменения в файлах с помощью обычных редакторов (NORTON и т.п.) и вводить и обрабатывать любые числовые массивы, записанные в кодах ASCII. Для этого нужно организовать числовой файл с расширением <d-c> по приведенной выше схеме.

При считывании файлов нужно учитывать, что массивы могут считываться в ДОПОЛНЕНИЕ к имеющимся или НА МЕСТО имеющихся. В первом случае возможно появление матриц или программ с одинаковыми именами, что может вызвать ошибки при работе.

При записи файлов (расширение не указывается) с помощью операций rpc, rpp записывается весь имеющийся массив. При записи файлов с помощью F1 (или F2), INS и F7 записывается выделенная часть массива. В последнем случае после нажатия F7 кроме надписи «Укажите имя файла» внизу экрана высвечиваются имена считанных ранее файлов. При нажатии соответствующей клавиши F происходит запись в указанный файл, т.е. не нужно создавать новый файл.

2. Список операций

2.1. Операции управления системой

csm – вывод списка всех имеющихся матриц с указанием их имен, числа строк и столбцов, а также операции, с помощью которой введена эта матрица. Если матрица введена в результате прогонки какой-то программы, то указывается и имя этой программы. В конце списка указывается текущее количество свободной памяти.

cum A_1 A_2 ... – удаление матриц A_1, A_2, ... Если указать n_1–n_2, то удаляются все матрицы с номерами от n_1 до n_2 в списке матриц включительно. Если указать *XX**, то удаляются все матрицы, имена которых начинаются с *XX*. При cum* удаляется весь массив.

ctm A_1 A_2 ... – вывод матриц A_1, A_2, ... на экран. Можно указывать и их номера в списке.

clm A_1 A_2 ... – печать матриц A_1, A_2, ... на принтере. Можно указывать и их номера в списке.

csp – вывод на экран списка программ с указанием имен, числа операций и комментария.

cup A_1 A_2 ... – удаление программ с именами A_1, A_2, ...

ctp A_1 A_2 ... – вывод на экран текста программ A_1, A_2, ...

clp A_1 A_2 ... – печать текста программ A_1, A_2, ... на принтере.

Часть из этих операций можно выполнить с помощью функциональных клавиш (см. выше).

cnn *N A* – определение размерности квадратной матрицы *A*.

cnm *N M A* – определение размерности прямоугольной матрицы *A*.

Операции cnn и cnm могут быть полезными при составлении стандартных программ, предназначенных для решения задач разной размерности.

cte *A i j* – вывод элемента $A(i,j)$ на экран в виде числа с плавающей запятой. Если *A* – скаляр, то индексы указывать не нужно. Если *A* – вектор, то можно указать один индекс. Эта операция удобна для просмотра «малых» элементов матрицы, так как при выводе матрицы все ее элементы печатаются в одном формате.

cve (*i s*) – изменение параметров счета: e(1) – машинного ноля, e(2) – точности вычислений, e(6) – фиксированного числа итераций и т.д. В данной операции *i* – номер параметра, *s* – новое значение. Если эти значения не указывать, то выводится список всех параметров.

cuo – удаление выполненных операций. Все выполненные операции запоминаются, и из них можно оформить новую программу. Данная операция удаляет выполненные операции, если они не нужны.

cnp – выдается информация о системе: величине числового массива, допустимого числа матриц, вводимых операций и встроенных программ. Эти характеристики задаются в файле DIM.BAS.

cps – пауза. Может использоваться в программном режиме для контроля за выполнением программ.

cst – конец работы (можно с помощью ESC).

2.2. Операции ввода матриц

ima *A n m* – ввод произвольной матрицы порядка $n \times m$.

ipr *A s* – ввод параметра *s*, т.е. формируется матрица *A* порядка 1×1 с элементом, равным *s*. Значение s может быть каким-то выражением.

im0 *A n m* – ввод нулевой матрицы порядка $n \times m$.

ime *A n m* (*a*) – ввод единичной матрицы порядка $n \times m$.

imn *A n m* (*a*) – ввод наддиагональной матрицы порядка $n \times m$.

imp *A n m* (*a*) – ввод поддиагональной матрицы порядка $n \times m$.

imk *A n m* (*a*) – ввод косодиагональной матрицы порядка $n \times m$.

ied *A n m* (*a*) – ввод матрицы, состоящей из 1.

inr *A n m* (*a/*) – ввод матрицы, состоящей из чисел 1, 2, 3, ... Если справа указывается /, то числа заносятся в обратном порядке.

igs *A n m* (*a*) – формирование случайной матрицы, элементы которой являются гауссовскими случайными величинами с нулевыми средними и единичными дисперсиями.

irs *A n m* (*a*) – формирование случайной матрицы, элементы которой являются равномерно распределенными в интервале [0,1] случайными величинами.

ips *A n m p* – формирование случайной матрицы, элементы которой являются случайными величинами, распределенными по закону Пуассона с параметром *p*.

ior *A n i* (*a/*) – ввод *i*-го орта порядка *n*, т.е. вектора порядка *n*, у которого все элементы равны нулю, кроме *i*-го, равного 1. Если справа указывается /, то образуется вектор-строка, иначе вектор-столбец.

ist *A n w* (*a/*) – ввод столбца, состоящего из степеней числа *w*, начиная с нулевой. Если справа указывается /, то образуется вектор-строка.

В этих операциях числа *n* и *m* могут быть произвольными. Например, при вводе единичной матрицы при n=1 и m=4 создается матрица A=[1 0 0 0]. Результат умножается на число *a*. Если это число не указывается, то a=1. Число *a* можно задавать в виде какого-либо выражения. При вводе слова скобки не указываются. Знак / указывается в конце слова.

iem *A i j s* – присвоение элементу $A(i, j)$ матрицы *A* значения *s*. Если матрицы *A* нет, то она вводится, причем ее размерность определяется числами *i* и *j*. Если эти числа превышают размерность существующей матрицы *A*, то ее размерность увеличивается, причем промежуточные элементы приравниваются нулю.

iev *A i s* (*/*) – присвоение элементу *i* вектора *A* значения *s*. Если вектора *A* нет, то он вводится, причем его размерность равна *i*. При добавлении признака / образуется вектор-строка, иначе – вектор-столбец. Если число *i* превышает размерность существующего вектора *A*, то его размерность увеличивается, причем промежуточные элементы приравниваются нулю.

ism *A n k* – ввод спектральной матрицы *A* порядка $n \times 2$, *k* – число вещественных собственных чисел. В первый столбец включаются вещественные части собственных чисел, во второй – мнимые.

2.3. Операция изменения матриц

kpc *X A i j* – матрица *X* получается из матрицы *A* путем перестановки *i*-го и *j*-го столбцов.

kps *X A i j* – матрица *X* получается из матрицы *A* путем перестановки *i*-й и *j*-й строки.

kuc *X A i j* – матрица *X* получается из матрицы *A* путем удаления столбцов с *i*-го по *j*-й включительно.

kus *X A i j* – матрица *X* получается из матрицы *A* путем удаления строк с *i*-й по *j*-ю включительно.

kic *X A i j* – матрица *X* получается путем выделения из матрицы *A* столбцов с *i*-го по *j*-й включительно.

kis *X A i j* – матрица *X* получается путем выделения из матрицы *A* строк с *i*-й по *j*-ю включительно.

Если в этих операциях указывается только *i*, то по умолчанию $j = i$ и в операциях участвует только одна *i*-я строка (или столбец).

kbc *X A i j k* – перестановка блока столбцов: столбцы с *i*-го по *j*-й переставляются на место, начиная с *k*-го. Промежуточные столбцы передвигаются.

kbs *X A i j k* – перестановка блока строк: строки с *i*-й по *j*-ю переставляются на место, начиная с *k*-й. Промежуточные строки передвигаются.

kdc *X A i j k* – дублирование блока столбцов: столбцы с *i*-го по *j*-й дублируются на место, начиная с *k*-го. Если $k>m$, то промежуточные столбцы заполняются нулями.

kds *X A i j k* – дублирование блока строк: строки с *i*-й по *j*-ю дублируются на место, начиная с *k*-й. Если $k>n$, то промежуточные строки заполняются нулями.

kie *X A i j* – выделение *ij*-го элемента матрицы *A*. Если *A* – вектор, то указываются только один индекс.

kim *X A i j k l* – матрица *X* получается путем выделения из матрицы *A* минора с угловыми элементами (i, j) – (k, l). Если *A* – вектор, то указываются только два индекса.

kpb *A i j* – разделение матрицы *A* на 4 блока, т.е. образуются матрицы с именами A_{11}, A_{12}, A_{21}, A_{22} такие, что $X = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$, причем $i \times j$ – размер блока A_{11}. Если указывается только *i*, то по умолчанию $j = i$.

kkv *X A B i j* – в матрицу *A* вписывается матрица *B* так, что на место элемента (i, j) матрицы *A* ставится элемент (1, 1) матрицы *B*. Числа *i* и *j* могут быть любыми и превышать размеры матрицы *A*. В последнем случае матрица *X* дополняется нулями. Если указывается только *i*, то по умолчанию $j = i$.

2.4. Операция формирования матриц

fmd *X A* (*/*) – формирование квадратной диагональной матрицы *X* с диагональными элементами, равными элементам матрицы (или вектора) A. Если добавить /, то X – вектор, составленный из элементов главной диагонали матрицы A.

fmf X A – формирование сопровождающей матрицы (Фробениуса), последняя строка, которой состоит из элементов вектора A со знаком минус, т.е.

$$X = \begin{bmatrix} 0 & 1 & 0 \cdots 0 \\ 0 & 0 & 1 \cdots 0 \\ \cdots & \cdots & \cdots \cdots \cdots \\ -a_1 & -a_2 & -a_3 \cdots -a_n \end{bmatrix}. \qquad (4.1)$$

fmo X A i – формирование матрицы отражения для столбца A, т.е. $X = I - \alpha v v^{\mathrm{T}}$, где $\alpha = 2/v^{\mathrm{T}}v$, вектор – v строится следующим образом:

$$v(k) = \begin{cases} 0 & \text{при } k < i, \\ s & \text{при } k = i, \\ A(k) & \text{при } k > i, \end{cases}$$

где $s = A(i) \pm [A^2(i) + \ldots + A^2(n)]^{1/2}$, причем + или – выбирается так, чтобы число s было максимальным по модулю.

fmu X A B (/) – X – матрица управляемости, т.е.

$$X = \left[b_1\, Ab_1 \ldots A^{n-1}b_1\, , \ldots , b_m\, Ab_m \ldots A^{n-1}b_m\, ,\right],$$

где $b_1, \ldots, b_m$ – столбцы матрицы B. Если справа добавляется /, то в X заносятся только линейно независимые столбцы матрицы управляемости. Одновременно формируется матрица X_#, в которой указываются номера этих столбцов.

fmn Y A C (/) – Y – матрица наблюдаемости, т.е.

$$Y^{\mathrm{T}} = \left[c_1\, A^{\mathrm{T}} c_1 \ldots A^{\mathrm{T}^{n-1}} c_1 , \ldots , c_m\, A^{\mathrm{T}} c_m \ldots A^{\mathrm{T}^{n-1}} c_m\right]^{\mathrm{T}},$$

где $c^{\mathrm{T}}_1, \ldots, c^{\mathrm{T}}_m$ – строки матрицы C. Если справа добавляется /, то в Y заносятся только линейно независимые строки матрицы наблюдаемости. Одновременно формируется матрица Y_#, в которой указываются номера этих строк.

flc X A – матрица X состоит из линейно независимых столбцов матрицы A.

fls X A – матрица X состоит из линейно независимых строк матрицы A.

fkc X A B – компоновка по столбцам: $X = [A\ B]$ – к матрице A справа приписывается матрица B. Если число строк в этих матрицах разное, то недостающие снизу строки в X заполняются нулями. Если матрицы A нет, то $X = B$.

fks X A B – компоновка по строкам: $X = \begin{bmatrix} A \\ B \end{bmatrix}$ – к матрице A снизу приписывается матрица B. Если число столбцов в этих матрицах разное, то недостающие справа столбцы в X заполняются нулями. Если матрицы A нет, то $X = B$.

fkb X A_{11} A_{12} A_{21} A_{22} – компоновка из блоков, т.е. $X = \begin{bmatrix} A_1 & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$.

Если размерности блоков не совпадают, то недостающие в X столбцы (справа) или строки (снизу) заполняются нулями. Например, при операции fkb X A_{11} 0 A_{21} A_{22} образуется матрица $X = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix}$.

foc X A – формирование столбца из столбцов матрицы A

fos X A – формирование строки из строк матрицы A.

fkd X A_1, A_2 ... , A_n – компоновка блочно-диагональной матрицы, т.е. $X = \begin{bmatrix} A_1 & 0 \cdots 0 \\ 0 & A_2 \cdots 0 \\ \cdots & \cdots\cdots\cdots \\ 0 & 0 \cdots A_n \end{bmatrix}$.

fip X A n m – матрица A переписывается построчно в матрицу X размерности $n \times m$. Если размерность X оказывается больше размерности матрицы A, то недостающие элементы заполняются нулями. С помощью этой операции можно изменять размерность числовых массивов.

2.5. Арифметические операции

mam X A (a) – умножение матрицы на число – $X = aA$. По умолчанию $a = 1$.

mtr X A – транспонирование матрицы – $X = A^{\mathrm{T}}$.

mcm X A –симметризация – $X = (A + A^{\mathrm{T}})/2$.

mob X A – обращение матрицы – $X = A^{-1}$.

mex X A t – матричная экспонента – $X = \exp(At)$.

msv X A B (a b) – сумма матриц – $X = aA + bB$. По умолчанию $a = b = 1$.

msb X A B – разность матриц – $X = A - B$.

mpr X A B (a) – произведение матриц A и B – $X = aAB$. По умолчанию $a = 1$.

mkr X A B – кронекерово произведение матриц, т.е. $X = \begin{bmatrix} b_{11}A \cdots b_{1m}A \\ \cdots\ \cdots\ \cdots \\ b_{n1}A \cdots b_{nm}A \end{bmatrix}$.

mks X A – квадратный корень из симметрической матрицы, т.е. $A = XX^{\mathrm{T}}$.

mkf X A (B) – квадратическая форма – $X = A^{\mathrm{T}}BA$. Если матрица B не указывается, то она полагается единичной.

mvp X A (B) – квадратическая форма – $X = ABA^{\mathrm{T}}$. Если матрица B не указывается, то она полагается единичной.

moh X A m – $X = A_{11} - A_{12}(A_{22})^{-1}A_{21}$. Матрица A разбивается на блоки $\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$, причем блок A_{11} имеет размерность $m \times m$.

mod X A_1 A_2 ... (/) – в X заносится значение максимального по модулю элемента матриц A_1, A_2, ...

min X A_1 A_2 ... (/) – в X заносится значение минимального элемента матриц A_1, A_2, ...

max X A_1 A_2 ... (/) – в X заносится значение максимального элемента матриц A_1, A_2, ...

Если в последних операциях указать /, то дополнительно образуется матрица с именем A*_mod (или A*_min, A*_max), в которую заносятся индексы этого элемента.

mps X A B ... $f(A, B, \ldots)$ – поэлементные вычисления, т.е. $X(i, j) = f(A(i, j), B(i, j), \ldots)$, где $f(A, B, \ldots)$ – некоторое выражение, куда входят имена матриц A, B, ... Эти матрицы должны иметь одинаковые размерности и разные имена. Их число не должно превышать 16. Например, при вводе операции mps X A B A*B образуется матрица X, составленная из поэлементных произведений элементов матриц A и B. Эта операция удобна для табуляции функций. Для этого в качестве A нужно взять матрицу, состоящую из чисел натурального ряда.

2.6. Характеристики матриц

asm X A (a/) – X – сумма элементов матрицы A. Если добавить /, то вычисляется след матрицы A, т.е. сумма элементов главной диагонали. По умолчанию $a = 1$.

arg X A – $X = \operatorname{rang} A$.

adt X A – $X = \det A$.

agm X A – X – вектор, составленный из главных миноров матрицы A.

axm X A – X – вектор [n], составленный из коэффициентов характеристического многочлена матрицы A.

akp X A(/) – вычисление спектральной матрицы: X – матрица порядка $n \times 2$, составленная из собственных значений произвольной матрицы A. Первый столбец состоит из вещественных частей, второй – из мнимых. Если добавить /, то образуются матрицы $X\#B$ и $X\#M$, составленные из соответственно вещественных и мнимых частей собственных векторов матрицы A.

aks X A (/) – X – вектор, составленный из собственных значений симметрической матрицы A. Если добавить /, то образуется матрица $X\#$, составленная из соответствующих собственных векторов.

apm X A P – $X = P^{-1}AP$ – подобная матрица, где P – невырожденная матрица.

arq R Q A – вычисление матриц R и Q для A, где R – верхняя треугольная матрица, Q – ортогональная матрица такая, что $A = QR$.

asr H S A – вычисление матрицы Шура H – верхней блочно-треугольной матрицы с диагональными блоками порядка 1×1 или 2×2. S – ортогональная матрица такая, что $H = S^{T}AS$.

ahs H S A – вычисление матрицы Хессельберга H– верхней треугольной матрицы с ненулевой первой поддиагональю, S – ортогональная матрица такая, что $H = S^{T}AS$.

2.7. Решение алгебраических уравнений для непрерывных систем

eab X A B – решение алгебраического уравнения $AX = B$ при квадратной и невырожденной матрице A, т.е. $X = A^{-1}B$.

esl X A D B – решение уравнения Сильвестра $AX + XD + B = 0$.

ens Y A D B X – невязка уравнения Сильвестра, т.е. $Y = AX + XD + B$.

elp X A C – решение прямого уравнения Ляпунова $AX + XA^{T} + C = 0$.

elo X A C – решение обратного уравнения Ляпунова $A^{T}X + XA + C = 0$.

enl Y A C X (/) – невязка решения прямого уравнения Ляпунова, т.е. $Y = AX + XA^{T} + C$. При добавлении знака / вычисляется невязка решения обратного уравнения Ляпунова $Y = A^{T}X + XA + C$.

erp X A C H (X_0) – решение прямого уравнения Риккати $AX + XA^{T} + C - XHX = 0$.

ero X A C H (X_0) – решение обратного уравнения Риккати $A^{T}X + XA + C - XHX = 0$.

enr Y A C H X (/)– невязка решения прямого уравнения Риккати, т.е. $Y = AX + XA^{T} + C - XHX$. При добавлении знака / вычисляется невязка решения обратного уравнения Риккати $Y = AX + XA^{T} + C - XHX$.

В этих уравнениях матрицы A, B, C, D, H заданы, X – искомое решение. Матрицы A, C, D, H должны быть квадратными, причем C и H – симметрическими. При решении уравнений Сильвестра и Ляпунова применяется предварительное преобразование матриц A и D к форме Шура. Уравнения Риккати решаются методом последовательных приближений Ньютона, когда на каждом шаге решается соответствующее уравнение Ляпунова. В операциях решения уравнений Риккати X_0 – начальное приближение. Если оно не задается, то оно выбирается автоматически. Решение может расходиться, если не выполняются условия управляемости и наблюдаемости, а также если начальное приближение X_0 не обеспечивает устойчивость замкнутой системы.

eac X A B – псевдообращение решения алгебраического уравнения $AX = B$ при произвольной матрице A порядка $n \times m$. Решение вычисляется в виде

$$X = \begin{cases} (A^{T}A + \varepsilon I)^{-1}A^{T}B & \text{при } n \ge m, \\ A^{T}(AA^{T} + \varepsilon I)^{-1}B & \text{при } m > n. \end{cases}$$

Здесь ε – регуляризующий малый параметр.

emf X F Z (C D) – минимизация функционала $J = (Z - FX)^{T}C(Z - FX) + X^{T}DX$ относительно вектора X, т.е. $X = (F^{T}CF + D)^{-1}F^{T}CZ$. Если C и D не указываются, то по умолчанию $C = I$ и $D = 0$.

2.8. Решение алгебраических уравнений для дискретных систем

hab X A B – решение алгебраического уравнения $X = AX + B$ при квадратной матрице A, т.е. $X = (I - A)^{-1}B$.

hsl X A D B – решение уравнения Сильвестра $X = AXD + B$.

hns Y A D B X – невязка уравнения Сильвестра, т.е. $Y = AXD + B - X$.

hlp X A C – решение прямого уравнения Ляпунова $X = AXA^{T} + C$.

hlo X A C – решение обратного уравнения Ляпунова $X = A^{T}XA + C$.

hnl X A C X (/) – невязка решения прямого уравнения Ляпунова, т.е. $Y = AXA^{T} + C - X$. При добавлении знака / вычисляется невязка решения обратного уравнения Ляпунова $Y = A^{T}XA + C - X$.

hrp X A C H (X_0) – решение прямого уравнения Риккати $X = AXA^{T} + C - AXH^{T}(I + HXH^{T})^{-1}HXA^{T}$.

hro X A C B (X_0) – решение обратного уравнения Риккати $X = A^{T}XA + C - A^{T}XB(I + B^{T}XB)^{-1}B^{T}XA$.

hnr Y A C H X (/) – невязка решения прямого уравнения Риккати, т.е. $Y = AXA^{T} + C - AXH^{T}(I + HXH^{T})^{-1}HXA^{T} - X$. При добавлении знака / вычисляется невязка решения обратного уравнения Риккати, т.е. $Y = A^{T}XA + C - A^{T}XB(I + B^{T}XB)^{-1}B^{T}XA - X$.

В этих уравнениях матрицы A, B, C, D, H заданы, I – единичная матрица, X – искомое решение. Матрицы A и C должны быть квадратными, матрица C – симметрической. При решении уравнений Сильвестра и Ляпунова применяется предварительное преобразование матриц A и D к форме Шура. Уравнения Риккати решаются методом последовательных приближений Ньютона, когда на каждом шаге решается соответствующее уравнение Ляпунова. В операциях решения уравнений Риккати X_0 – начальное приближение. Если оно не задается, то оно выбирается автоматически. Решение может расходиться, если не выполняются условия управляемости и наблюдаемости, а также если начальное приближение X_0 не обеспечивает устойчивость замкнутой системы.

hkf $K\ A\ H\ X$ – вычисление коэффициента фильтра Калмана $K = AXH^{\mathrm{T}}(I + HXH^{\mathrm{T}})^{-1}$, где X – решение прямого уравнения Риккати, которое должно быть уже найдено.

hkr $K\ A\ B\ X$ – вычисление коэффициента оптимального регулятора (АКОР) $K = (I + B^{\mathrm{T}}XB)^{-1}B^{\mathrm{T}}XA^{\mathrm{T}}$, где X – решение обратного уравнения Риккати, которое должно быть уже найдено.

2.9. Дискретизация непрерывных систем

Данная группа операций связана с преобразованием непрерывной системы в дискретную. Непрерывная система задается уравнением

$$\dot{x} = Ax + Bu, \qquad (9.1)$$

где $x(t)$ – n-мерный вектор состояний объекта; $u(t)$ – m-мерный вектор входных воздействий; $q(t)$ – стандартный r-мерный белый гауссовский шум; A, B, L – заданные матрицы соответствующих размерностей. Соответствующая дискретная система строится в виде

$$x(k+1) = Fx(k) + Gu(k) + Mr(k),\ \ k = 1, 2, \ldots, \qquad (9.2)$$

где $r(1)$, $r(2)$, ..., $r(k)$, ... – последовательность независимых между собой стандартных гауссовских случайных величин. Здесь

$$F = \exp(At) = U(t) \text{ – переходная матрица}, \qquad (9.3)$$

$$G = G(t) = \int_0^t U(t-s)B\,ds = A^{-1}(U(t)-1)B, \qquad (9.4)$$

M – такая матрица, что

$$MM^{\mathrm{T}} = \int_0^t U(s)LL^{\mathrm{T}}U^{\mathrm{T}}(s)\,ds, \qquad (9.5)$$

где t – интервал дискретизации; I – единичная матрица.

Эти преобразования выполняются с помощью следующих операций. Здесь матрицы A, B, C, D, L считаются заданными, t – задаваемый интервал дискретизации.

dao $F\ A\ t$ – вычисление матрицы F по формуле (9.3).

dab $F\ G\ A\ B\ t$ – вычисление матриц F и G по формулам (9.3) и (9.4).

dar $F\ M\ A\ L\ t$ – вычисление матриц F и M по формулам (9.3) и (9.5).

dbr $F\ G\ M\ A\ B\ L\ t$ – вычисление матриц F, G и M по формулам (9.3), (9.4) и (9.5).

2.10. Интегрирование дифференциальных уравнений

Данная группа операций выполняет интегрирование уравнения

$$\dot{x} = Ax + b + Lq,\ x(0) = x_0 \qquad (10.1)$$

на интервале времени $[0, t]$. Здесь x – n-мерный вектор состояний объекта; b – n-мерный вектор входных постоянных воздействий; $q(t)$ – стандартный r-мерный белый гауссовский шум; A, L – заданные матрицы соответствующих размерностей. В процессе решения запоминается k промежуточных значений, равномерно расположенных на интервале интегрирования. В результате образуется матрица X порядка $n \times (k + 1)$, первым столбцом которой является вектор x_0, а последним – вектор $x(t)$.

tao $X\ A\ x_0\ t\ k$ – интегрирование уравнения (10.1) при $b = 0$ и $L = 0$.

tab $X\ A\ B\ x_0\ t\ k$ – интегрирование уравнения (10.1) при $L = 0$.

tar $X\ A\ L\ x_0\ t\ k$ – интегрирование уравнения (10.1) при $b = 0$.

tbr $X\ A\ B\ L\ x_0\ t\ k$ – интегрирование уравнения (10.1).

2.11. Решение разностных уравнений

Эта группа операций выполняет решение уравнения

$$x(k+1) = Fx(k) + b + Mr(k),\ x(0) = x_0, \qquad (11.1)$$

k=0, 1,. .. Здесь x – n-мерный вектор состояний объекта; b – n-мерный вектор входных постоянных воздействий; $r(0)$, $r(1)$, ..., $r(k)$, ... – последовательность стандартных гауссовских величин; F и M – заданные матрицы соответствующих размерностей. В процессе решения образуется матрица X порядка $n \times (k + 1)$, где k – число шагов. Первым столбцом матрицы X является вектор x_0, последним – вектор $x(k)$.

zao $X\ F\ x_0\ k$ – решение уравнения (11.1) при $b = 0$ и $M = 0$.

zab $X\ F\ b\ x_0\ k$ – решение уравнения (11.1) при $M = 0$.

zar $X\ F\ M\ x_0\ k$ – решение уравнения (11.1) при $b = 0$.

zbr $X\ F\ b\ M\ x_0\ k$ – решение уравнения (11.1).

Одновременно введены операции решения уравнения

$$x(k+1) = Fx(k) + b + Mr(k) + f(k),\ x(0) = x_0, \qquad (11.2)$$

где, в отличие от (11.1), добавлен вектор $f(k)$, который на каждом шаге с вероятностью p равен f и с вероятностью $1 - p$ равен 0.

zfo $X\ F\ x_0\ p\ k$ – решение уравнения (11.2) при $b = M = 0$.

zfb $X\ F\ b\ x\ p\ k$ – решение уравнения (11.2) при $M = 0$.

zfr $X\ F\ M\ x_0\ p\ k$ – решение уравнения (11.2) при $b = 0$.

zfp $X\ F\ M\ b\ x_0\ p\ k$ – решение уравнения (11.2).

2.12. Операции с многочленами

lpn $X\ n\ w$ – введение биномиального многочлена $[n]$: X – вектор, составленный из коэффициентов бинома Ньютона, т.е. $X(i) = C_n^i w^{n-i}$; n – порядок многочлена; w – параметр.

lpb $X\ n\ w$ – введение многочлена Батерворда $[n]$ порядка n, т.е. X – вектор, составленный из коэффициентов этого многочлена. Многочлены Батерворда соответствуют кругам Батерворда, когда собственные числа располагаются в левой полуплоскости по дуге радиуса w.

lsb $X\ n\ w$ – X – спектральная $n \times 2$-матрица Батерворда, составленная из корней многочлена Батерворда. В первый столбец включены вещественные части корней, во второй – мнимые.

lfm $X\ A$ – построение многочлена $[n]$ для спектральной матрицы A, т.е. X – вектор, составленный из коэффициентов многочлена, корни которого задаются спектральной матрицей A.

lmg $X\ A$ – матрица Рауса, построенная из коэффициентов многочлена $A(w)$ $[n]$, т.е.

$$X = \begin{bmatrix} a_n & 1 & 0 & \cdots & 0 & 0 \\ a_{n-2} & a_{n-1} & a_n & \cdots & 0 & 0 \\ a_{n-4} & a_{n-3} & a_{n-2} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & a_2 & a_3 \\ 0 & 0 & 0 & \cdots & 0 & a_1 \end{bmatrix}.$$

lnp $X\ A\ (a\ l)$ – перенормировка многочлена: в вектор A добавляется элемент $a_{n+1} = a$. Если a не указывается, то $a = 1$. Если справа добавить символ /, то из вектора A исключается элемент a_n (если только $n>1$).

lsm X A B (a b /) – сумма многочленов $A(w)$ и $B(w)$ [s]: $X(w) = aA(w) + bB(w)$. Если справа добавить символ /, то многочлены A и B, а также и многочлен X считаются нормированными, в противном случае – ненормированными, включая и многочлен X. Если весовые коэффициенты a и b не указываются, то $a = b = 1$. Если складываются два нормированных многочлена одинаковой степени, то коэффициенты суммарного многочлена делятся пополам.

lpm X A B (/) – произведение многочленов $A(w)$ и $B(w)$ [s]: $X(w) = A(w)B(w)$. Если справа добавить символ /, то многочлены A и B и, следовательно, X считаются нормированными, в противном случае – ненормированными (включая и X).

log X A – определители Гурвица для многочлена $A(w)$ [n]: X – вектор, составленный из главных миноров матрицы Рауса, построенной на основе многочлена $A(w)$.

lkp X A – корни многочлена $A(w)$ [n]: X – спектральная $n \times 2$-матрица, составленная из вещественных и мнимых частей корней многочлена.

lkr X Y B C D – $X(w)$ [n] и $Y(w)$ [p] – многочлены, удовлетворяющие уравнению $B(w)X(w) - C(w)Y(w) = D(w)$, где многочлены $B(w)$ [n], $C(w)$ [p] и $D(w)$ [n] заданы. Эта операция используется при вычислении коэффициентов пропорционально-интегрального регулятора.

lmr X A – X – матрица Джури, используемая при канонических преобразованиях:

$$X = \begin{bmatrix} a_2 & a_3 & \cdots a_{n-1} & a_n & 1 \\ a_3 & a_4 & \cdots a_n & 1 & 0 \\ a_4 & a_5 & \cdots 1 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_n & 1 & \cdots 0 & 0 & 0 \\ 1 & 0 & \cdots 0 & 0 & 0 \end{bmatrix},$$

где $a_2, a_3, \ldots, a_n$ – коэффициенты многочлена $A(w)$ [n].

ltb X A a b n (/) – табуляция многочленов [p], коэффициенты которых содержатся в строках матрицы A. Эти коэффициенты располагаются по мере возрастания степени аргумента. Коэффициент при старшей степени указывать обязательно. [a, b] – интервал табулирования, n – число точек. В первую строку X заносятся значения аргумента. При наборе этой операции значения a и b можно задавать в логарифмическом масштабе. Для этого справа после набора всей операции нужно добавить символ /. Графики функций следует строить с помощью операций gkf, gpf, glf (фазовый портрет).

lmm X B A (/) – вычисление матричного многочлена $B(A)$ от матрицы A. Многочлен $B(w)$ задается вектором $B=[b_1, b_2, \ldots, b_n]$. Если символ / не указывается, то многочлен $B(w)$ предполагается нормированным и

$$X = B(A) = A^n + b_n A^{n-1} + b_{n-1} A^{n-2} + \ldots + b_2 A + b_1 I.$$

Если символ / указывается, то многочлен $B(w)$ предполагается ненормированным и

$$X = B(A) = b_n A^{n-1} + b_{n-1} A^{n-2} + \ldots + b_2 A + b_1 I.$$

2.13. Частотные методы

wgd X A (a b n /) – годограф Михайлова: первая строка X состоит из значений вещественной части многочлена $A(iw)$ [p], вторая – из мнимой.

waf X W (a b n /) – амплитудно-фазовая частотная характеристика (АФЧХ): первая строка X состоит из значений вещественной части передаточной функции $W(iw)$, вторая – мнимой.

wax X W (a b n /) – амплитудно-частотная характеристика (АЧХ): первая строка X состоит из значений параметра w, вторая – из значений модуля передаточной функции $W(iw)$.

wfx X W (a b n /) – фазово-частотная характеристика (ФЧХ): первая строка X состоит из значений параметра w, вторая – из значений аргумента передаточной функции $W(iw)$, деленных на $\pi = 3{,}14\ldots$

В этих операциях [a, b] – интервал изменения параметра w, n – число точек. Если числа a, b, n не указываются, то по умолчанию $a = 0$, $b = 10$^(2/p), $n = 100$, где p – наибольшая степень многочленов. При наборе этих операций значения a и b можно задавать и в логарифмическом масштабе. Для этого справа после набора всей операции нужно добавить символ /. Если при этом числа a, b, n не указываются, то по умолчанию $a = -4$, $b = 2/p$, $n = 100$. Графики функций следует строить с помощью операций gkf, gpf, glf (фазовый портрет). Передаточные функции $W(iw)$ задаются в виде матрицы порядка $2 \times (n + 1)$ (см. п.1.2.3).

wpr W W_1 W_2 – перемножение передаточных функций $W(p) = W_1(p)W_2(p)$.

wsm W W_1 W_2 – сложение передаточных функций $W(p) = W_1(p) + W_2(p)$.

wob W W_0 – вычисление передаточной функции для замкнутой системы $W(p) = W_0(p)/(1 + W_0(p))$, где $W_0(p)$ – передаточная функция разомкнутой системы.

wds A B C W – построение динамической системы вида

$$\dot{x} = Ax + Bu,\ z = Cx \qquad (13.1)$$

для передаточной функции $W(p)$, представленной в виде матрицы порядка $2 \times (n + 1)$. Эта передаточная функция должна быть физически реализуемой, т.е. $a_{n+1} \neq 0$. Если $a_{n+1} = 1$, то A будет матрицей Фробениуса (4.1), в нижней строке которой стоят элементы второй строки матрицы W без последнего элемента, $B = [0\ 0 \ldots 1]^T$, C – первая строка матрицы W без последнего элемента. Если коэффициент $a_{n+1} \neq 1$, то элементы нижней строки матрицы A и элементы строки C делятся на a_{n+1}.

wpf W A B C – $W(p) = C(pI - A)^{-1}B$ – передаточная функция для системы (13.1), где C должна быть строкой, B – столбцом.

2.14. Представление графического материала

Организация представления графического материала описана в первой части.

glg A_1(\номера) A_2(\номера)... (= t /) – график в левой части экрана.

gpg A_1(\номера) A_2(\номера)... (= t /) – график в правой части экрана.

gkg A_1(\номера) A_2(\номера)... (= t /) – график в центре экрана.

lf A_1(\номера) A_2(\номера)... (/) – фазовый портрет в левой части экрана.

gpf A_1(\номера) A_2(\номера)... (/) – фазовый портрет в правой части экрана.

gkf A_1(\номера) A_2(\номера)... (/) – фазовый портрет в центре экрана.

gls A_1 A_2 ... – спектр матрицы в левой части экрана.

gps A_1 A_2 ... – спектр матрицы в правой части экрана.

gks A_1 A_2 ... – спектр матрицы в центре экрана.

2.15. Программный режим

В системе имеется возможность из операций составлять программы и работать с этими программами. Каждой программе присваивается свое имя аналогично тому, как вводятся имена матриц, причем имена программ могут совпадать с какими-либо именами введенных матриц.

2.15.1. Операции ввода и изменения программ

pri *A* (комментарий) – ввод новой программы *A*.

pio *A* (комментарий) – оформление выполненных операций в программу *A*. Все выполненные в процессе работы операции запоминаются. Данная операция позволяет оформить из них новую программу *A*.

pvp *A* – изменение программы *A* – переход в редактор.

pvm $P\ A_1\ A_2\ ...\ A_k\ /\ B_1\ B_2\ ...\ B_k$ – в программе *P* имена матриц A_1, ..., A_k заменяются соответственно на имена B_1, ..., B_k. Если после имени указать *, например A_1*, то во всех именах, начинающихся с A_1, буквы A_1 заменятся на B_1.

Эта операция аналогична изменению имен матриц в редакторе с помощью Ctrl + W, но выполняется без перехода в редактор. Поэтому она может быть использована в программном режиме

2.15.2. Операции условного перехода

pie *x a C*\ *D*\ *E* – условный переход: если $x < a$, то выполняется *C*; если $x = a$, то – *D*; если $x>a$, то – *E*.

pif *x a b C*\ *D*\ *E* – условный переход: если $x < a$, то выполняется *C*; если $a \le x \le b$, то – *D*; если $x> b$, то – *E*.

Здесь *x*, *a* и *b* могут быть числами или какими-то выражениям; *C*, *D*, *E* – какие-либо операции. Операции *C*, *D* или *E* могут отсутствовать, но знак \ должен указываться два раза. Если *C*, *D* или *E* отсутствуют, то при выполнении соответствующего условия осуществляется переход к следующей в программе операции. Если *C*, *D*, *E* – операции, то для их ввода можно воспользоваться подсказкой с помощью ENTER.

pbf **n* – безусловный переход к операции с меткой **n*.

2.15.3. Операции выполнения программ

prn *A* (*n* /) – запуск программы *A*. Она повторяется *n* раз. Если *n* не указывается, то $n = 1$. Эту операцию можно использовать для организации цикла. Если в конце операции указать символ /, то при счете выводятся результаты всех промежуточных операций, что может быть полезным при отладке программ.

ppp *A B n b h i j* (/) – прогонка программы A для *n* вариантов значений параметра – *i j*-го элемента матрицы *B*. Если *i* и *j* не указываются, то $i = j = 1$. *b* – значение параметра $B(i, j)$ в первом варианте, *h* – шаг изменения этого параметра. Если в конце операции символ / не указывается, то при счете *k*-го варианта параметр $B(i, j) = b + +h(k - 1)$, т.е. значение параметра увеличивается на шаг *h*. В противном случае $B(i, j) = bh^{k-1}$, т.е. значение параметра умножается на *h*. Матрица *B* не должна удаляться в процессе выполнения программы, а элемент $B(i, j)$ изменяться. Полученные результаты можно формировать в виде одной матрицы, для чего программа должна содержать операции вида: fkc *X X Y* или fks *X X Y*, где *Y* – результат счета в одном варианте, *X* – запоминаемый массив. Например, при наличии в программе операций: msm *Q A B*: fkc *X X Q* и выполнении трех вариантов счета образуется матрица $X = [Q1, Q2, Q3]$. В программе может формироваться несколько подобных матриц. В программе после операций, вычисляющих нужные матрицы, можно также поставить знак ! (это можно сделать в редакторе). В результате эти матрицы будут запоминаться и к их именам будет добавляться номер варианта. Например, если в программе имеется операция msm *X A B* ! и будет выполняться 3 варианта, то в результате счета появятся матрицы *X*_1, *X*_2, *X*_3. После выполнения операции ppp варьируемый элемент $B(i, j)$ принимает первоначальное значение. В конце работы программы на экран выводится результат последней операции. Промежуточные результаты можно получить, включив в программу операции ctm или clm.

2.15.4. Специальные вычисления

ptb $Z\ A\ X\ Y\ n\ x_0\ h$ (/) – табуляция функции $y = f(x)$. Здесь *Z* – результат табуляции – матрица, состоящая из столбцов $f(x_1), f(x_2), ..., f(x_n)$; *A* – имя программы, вычисляющей функцию $y = f(x)$; *X* – имя аргумента; *Y* – имя результата в этой программе. Переменная *Y* должна быть вектором или скаляром, *n* – число точек, x_0 – начальное значение аргумента *X*, *h* – шаг изменения значения *X*. Если справа символ / не указывается, то на *k*-м шаге $x_k = x_0 + h(k-1)$, в противном случае $x_k = x_0 h^{k-1}$.

pig *Z A X Y a b* (*n*) – интегрирование функции $y = =f(x)$. Здесь *Z* – результат; *A* – имя программы, вычисляющей функцию $y = f(x)$; *X* – имя аргумента; *Y* – имя функции (*Y* должно быть вектором или скаляром); *a*, *b* – пределы интегрирования; *n* – число точек интегрирования. Интеграл вычисляется по формуле Симпсона. Если число *n* не указывается, то $n = 100$.

ptf *Z X f*(*x*) *a b n* – табуляция функции $y = f(x)$. Здесь *Z* – результат табуляции: первая строка –значения аргумента, вторая – значения функции, *X* – имя переменной; $f(x)$ – скалярное выражение; $[a, b]$ – интервал изменения аргумента; *n* – число точек. При вводе функции недопустимы пробелы.

pin *Z X f*(*x*) *a b* (*n*) – интегрирование функции $y= =f(x)$. Здесь *Z* – результат; *X* – имя переменной интегрирования; $f(x)$ – выражение; *a b* – пределы интегрирования; *n* – число точек. Интеграл вычисляется по формуле Симпсона. Если *n* не указывается, то $n = 100$.

Отличие операций ptb и pig от ptf и pin состоит в том, что в операциях ptb и pig функция вычисляется с помощью программы *A* и может быть векторной, а в операциях ptf и pin функция задается непосредственно скалярным выражением.

2.15.5. Введение специальных функций

pfn *Fa x f*(*x*) – ввод функции $Fa[x] = f(x)$. Здесь *x* – имя аргумента. Имя функции должно начинаться с *F*. При обращении к этим функциям аргумент ставится в квадратные скобки. Аргумент может быть любым. Например, при обращении допустимо выражение $F[F[t + 3]]$. При вводе выражения пробелы недопустимы.

psf – список введенных функций.

ppf *A* (комментарий) – оформление введенных функций в программу *A*. Данная операция позволяет оформить операции ввода функций в программу, которую можно запомнить и затем использовать в дальнейшей работе.

2.16. Работа с файлами

Подробно об организации файлов и работы с ними изложено выше в п.1.7.

rpc *A* – запись сформированного числового массива в файл.

ric *A* – считывание числового массива из файла.

rdc *A* – то же, но считываемый числовой массив записывается в дополнение к имеющемуся. Здесь нужно учитывать возможность появления матриц с одинаковыми именами.

rpp *A* – запись массива программ в файл.

rip *A* – считывание массива программ из файла.

rdp *A* – то же, но считываемый массив программ записывается в дополнение к имеющемуся. Здесь нужно учитывать возможность появления программ с одинаковыми именами.

ruf *A* – удаление файла.

rof *A* – список всех файлов.

rsf *A* – список файлов с расширением <d-c> и <d-p>.

rrik *A* – открытие каталога.

ruk *A* – удаление каталога.

В этих операциях *A* – имя каталога или имя файла. Если при записи расширение файла не указывается, то файлам, содержащим числовые массивы, по умолчанию присваивается расширение <d-c>, а файлам, содержащим массивы программ, – расширение <d-p>. При считывании таких файлов расширение не указывается. Если при считывании имя файла не указывается, то появляется окно со всеми файлами, но только с расширениями <d-c> или <d-p>. С другой стороны, при записи массивов можно указывать любое расширение, но тогда при их считывании расширение нужно указывать обязательно. При записи внизу панели высвечиваются имена ранее считанных файлов. Можно записать массивы в эти файлы, нажав клавишу F с соответствующим номером. При удалении файлов расширение указывается во всех случаях, кроме удаления с помощью F3 (или F4), INS и F8. При входе в другую директорию указывается ее имя (например, *a*:\ или *a*:\dir\).

rpf A_1 A_2 ... (*.aaa*) ... – запись матриц A_1, A_2, ... в файлы с последовательным доступом с именами A_1*.aaa*, A_2*.aaa*, ... Расширение указывается только в конце вводимого слова (точка обязательна). Если расширение не указывается, то по умолчанию присваивается расширение <doc>.

rif *F* (*.aaa*) *A n m* – считывание числового массива из файла с последовательным доступом *F.aaa* (если расширение не указывается, то по умолчанию к имени добавляется расширение <doc>). *A n m* – имя матрицы и ее размерность, которое присваивается считанному массиву. Если *A n m* не указываются, то матрице присваивается имя файла, $n = 1$, а m равно числу считанных чисел.

Последние две операции позволяют проводить обмен числовыми массивами с другими вычислительными системами.

Статья поступила в научную редакцию «Кибернетика и информатика» 20 декабря 1999 г.