УДК 519.622.2 DOI: 10.31114/2078-7707-2020-4-31-38

Проблемы разработки математического ядра для программ моделирования динамики технических систем

Д.М. Жук, Д.Ю. Кожевников, В.Б. Маничев

zhuk@bmstu.ru, kozhevn@bmstu.ru, manichev@bmstu.ru,

МГТУ им. Н.Э. Баумана, г. Москва

***Аннотация* — В статье рассматриваются основные утверждения и научные результаты многолетней работы авторов в области моделирования динамических систем (включая изделия микроэлектроники), положенные в основу разработки библиотеки математических программ MZK - библиотеки на языке Си для решения алгебраических и дифференциальных уравнений с максимально возможной компьютерной точностью. Планируется использовать библиотеку MZK в качестве математического ядра для программного комплекса математического моделирования разнородных динамических систем ПА10 (PA10 – Программа Анализа 10 версия, Program for Analysis version 10). Показано, что для решения систем очень высокой размерности необходимо учитывать погрешности округления чисел при выполнении элементарных арифметических операций.**

***Ключевые слова* — автоматизированное проектирование электронных схем, математическое моделирование, динамические системы, обыкновенные дифференциальные уравнения, дифференциально-нелинейные алгебраические уравнения, линейные алгебраические уравнения.**

I. Введение

Важнейшим элементом систем автоматизированного проектирования (САПР) технических объектов, включая изделия микроэлектроники, является математическое моделирование объектов, процессов и явлений во временной области. Достоверность и точность математического моделирования динамических систем зависит от математического ядра, которое представляет собой блок решения систем дифференциально-алгебраических уравнений (ДАУ), частным случаем которых являются системы обыкновенных дифференциальных уравнений (ОДУ) в нормальной форме Коши, разрешенные относительно производных. Для решения этих систем применяются неявные методы интегрирования, которые сводятся к решению систем нелинейных алгебраических уравнений (НАУ) и в конечном счете линейных алгебраических уравнений (ЛАУ).

После написания математического ядра разработчику необходимо провести его тестирование для выявления ошибок и определения, удовлетворяет ли математическое ядро поставленным требованиям. Тестирование программ анализа динамики технических систем проводится в двух направлениях: тестирование на конкретных известных практических задачах (электрические, механические, гидравлические схемы и т.д.) и тестирование математического ядра (системы дифференциально-алгебраических уравнений (ДАУ). Поскольку основные характеристики (точность и скорость решения) зависят от математического ядра, его тестирование должно подтверждать надежность и эффективность разрабатываемых САПР. В настоящее время первое направление развито достаточно широко, особенно для САПР микроэлектроники (ECAD/EDA систем) [2], а второе применяется крайне редко, несмотря на то что оно необходимо для дальнейшего развития существующих САПР и разработки новых, поэтому авторы выполнили ряд работ по второму направлению [3]-[5]. В статье рассматриваются основные утверждения и научные результаты многолетней работы авторов в области моделирования динамических систем, положенные в основу разработки библиотеки математических программ MZK - библиотеки на языке Си для решения алгебраических и дифференциальных уравнений с максимально возможной компьютерной точностью. Планируется использовать библиотеку MZK в качестве математического ядра для программного комплекса математического моделирования разнородных динамических систем ПА10 (PA10) [1]. Решатели систем ОДУ в ПА10 будут превосходить по точности и достоверности решатели систем ОДУ в программах моделирования электронных схем общего назначения типа SPICE (например, программы Gear и Trapezoidal в программном комплексе Multisim). Показано, что для решения систем очень высокой размерности необходимо учитывать погрешности округления чисел при выполнении элементарных арифметических операций, что не учитывается в вышеуказанных программах моделирования электронных схем общего назначения.

II. Математические модели динамических систем

Математическое моделирование, инженерный анализ, верификация и валидация проектных решений при проектировании во временной области изделий микроэлектроники, наноэлектроники и мехатроники основано на решении систем ДАУ. Для решения систем ДАУ, как правило, используются неявные методы интегрирования, которые требуют решения соответствующих систем НАУ на каждом шаге интегрирования.

Математические модели динамических процессов в реальных технических объектах и системах необходимо разрабатывать на основе фундаментальных физических или химических законов в форме систем ДАУ, не разрешенных относительно производных: решать эти системы следует без каких-либо преобразований и без обязательного получения правых частей для соответствующих производных в явном аналитическом виде.

При математическом моделировании динамических процессов в реальных технических объектах и системах состояние этих объектов и систем обычно рассматривают в пространстве дифференциальных переменных или в пространстве переменных состояния с получением выражений для производных данных переменных по времени в явном, аналитическом виде. Однако этого недостаточно для достоверного и точного математического моделирования реальных технических систем и объектов, так как любой дифференциальной переменной состояния в реальном мире обычно соответствует определенная алгебраическая переменная. Например, в электротехнике и электронике переменным состояния – изменяющимся во времени напряжениям на емкостных элементах и токам через индуктивные элементы – соответствуют алгебраические переменные – токи через емкостные элементы и напряжения на индуктивных элементах; в механике переменным состояния – изменяющимся во времени скоростям тел определенной массы соответствуют такие алгебраические переменные, как силы инерции, и т.д. и т.п. В связи со всем изложенным выше состояние реальных динамических объектов и систем следует рассматривать в пространстве дифференциально-алгебраических переменных.

Основным физическим свойством алгебраических переменных является возможность идеального скачка их значения за бесконечно малый интервал времени, в то время как для дифференциальных переменных состояния подобные скачки невозможны, По этой причине общепринятое приведение систем ДАУ к системам ОДУ в нормальной форме Коши с преобразованием алгебраических переменных в переменные состояния физически неверно, а полученная таким образом математическая модель динамических процессов в реальных технических объектах и системах будет физически недостоверной. Следует учесть, что многие эквивалентные преобразования, выполняемые численно, на самом деле таковыми не являются вследствие ограниченности разрядной сетки компьютера. Только аналитические (символьные) преобразования исходных уравнений математических моделей являются строго эквивалентными. Например, для системы ЛАУ эквивалентными преобразованиями будут только перестановки строк и столбцов матрицы коэффициентов, тогда как матричные операции умножения и сложения уже не будут эквивалентными. Это утверждение легко обосновать на примере умножения матрицы коэффициентов системы ЛАУ на обратную матрицу. Теоретически в результате подобного умножения должна получиться единичная матрица, однако на компьютере такой результат в случае плохо обусловленных матриц достигнут не будет. По этой причине необходимо использовать только те методы и алгоритмы, которые не требуют численных матрично-векторных преобразований исходных уравнений. С помощью нормирования исходных данных и матрично-векторных преобразований всегда можно преобразовать плохо обусловленную систему ЛАУ в хорошо обусловленную (методы предобусловливания), однако полученная после таких преобразований система ЛАУ вследствие ограниченности разрядной сетки компьютера не будет соответствовать исходной системе. Из сказанного вытекает вывод о том, что необходимо разработать методы и алгоритмы достоверного и точного решения систем ЛАУ без использования каких-либо численных преобразований. Рассмотрим в качестве примера вычисление на компьютере значения арифметического выражения (1/3 + 1 – 1) * 3, точное значение которого равно 1. Заметим, что подобные выражения встречаются, в частности, при решении систем ЛАУ методом Гаусса. Вычисление значения этого выражения в программах на языке C/C++ при представлении чисел в стандартном для математических пакетов формате двойной точности (double precision) дает результат 0,99999999999999978000000000000000. Причиной погрешности является тот факт, что округление при выполнении арифметических операций в ЭВМ производится до чисел, равных степеням числа 2, а не числа 10. Эта особенность компьютерных вычислений может в конечном итоге приводить к серьезным ошибкам, особенно для систем большой размерности.

Задача решения систем ДАУ в расширенном пространстве дифференциально-алгебраических переменных без преобразований исходных систем ДАУ основана на постановке задачи в виде:

$$G(X, PX, Y, t) = 0, \qquad (1)$$

где X – вектор дифференциальных переменных (дифференцируемых по времени функций) в координатном базисе переменных состояния размерностью m; Y – вектор алгебраических переменных в полном координатном базисе дифференциально-алгебраических переменных размерностью k; $PX = dX/dt$ - вектор производных переменных состояния по времени размерностью m; G – вектор-функция размерностью m+k, - левая часть системы ДАУ. Заданы согласованные начальные условия: $X_0 = X(0), Y_0 = Y(0)$, отрезок интегрирования $t = [0, TK]$ и TK– время окончания интегрирования.

Учитывая тот факт, что вследствие ограниченности разрядной сетки компьютеров некоторые выполняемые на них матрично-векторные преобразования не являются эквивалентными, классические формы представления систем ОДУ рекомендуется использовать только для теоретических исследований при условии, что все преобразования исходных дифференциальных уравнений выполняются аналитически или в символьном виде.

III. Проблемы достоверного и точного решения математических моделей динамических систем

Отладка решателей математических моделей динамических систем выполняется на задачах (1) невысокой размерности и поэтому фактически не учитывается влияние погрешностей округления чисел при выполнении элементарных арифметических операций. Однако решение задач нейтронной кинетики высокой размерности подтвердило закон перехода количества в качество, так как на точность конечного результата значительное влияние оказали погрешности округления чисел при выполнении элементарных арифметических операций и необходимость применения вычислений с повышенной разрядностью чисел.

Простейший пример необходимости учета влияния погрешностей округления и вычислений с повышенной разрядностью для получения точного численного решения - это вычисление числа π по методу Архимеда. Он рассматривал круг диаметра 1, имеющий, следовательно, длину окружности π. Внутрь круга Архимед вписывал квадрат; периметр квадрата, равный $2\sqrt{2}$, меньше длины окружности и давал оценку снизу для π. Далее Архимед рассматривал вписанный правильный восьмиугольник, 16-угольник и т.д. Каждый раз число сторон вписанного правильного многоугольника удваивалось и в результате получались все лучшие оценки для π по формуле:

$$p_{n+1} = 2^n \sqrt{2 * (1 - \sqrt{1 - (\frac{p_n}{2^n})^2}},$$

где p_n – периметр правильного вписанного многоугольника с 2^n сторонами, при этом для вписанного квадрата имеем $p_2 = 2\sqrt{2}$, Вычисляя π с точностью элементарных арифметических операций double precision (стандартная точность для программ математического моделирования) получим все время разные результаты, а максимальную точность получим при n=10, при больших n точность уменьшается именно из-за ошибок округления, поэтому для получения более точной оценки числа π надо использовать вычисления с повышенной разрядностью при выполнении элементарных арифметических операций.

Особенностью всех разностных формул неявного интегрирования ОДУ является деление на шаг интегрирования h. В работе [6] показано, что если программировать эти формулы в классическом виде, взятом из учебников, то при уменьшении шага будем получать все время разные результаты, как и в вышеприведенном примере. При этом шаг не должен быть меньше машинной точности вычислений, равной 1E-8 для одинарной и 1E-16 для удвоенной точности вычислений. При меньших значениях шага h программы интегрирования, реализующие неявные методы интегрирования, могут выдавать качественно неверное решение из-за погрешностей округления чисел при выполнении операции деления. В работе [6] предложен метод решения этой проблемы для неявных методов интегрирования и приведена формула неявного метода Эйлера в приращениях, а в дальнейшем была разработана программа manzhuk, реализующая этот метод и неявный метод трапеций в приращениях. Исходная формула неявного метода трапеций — это система НАУ, которая относительно X_n имеет вид:

$$X_n = X_{n-1} + \left(\frac{h}{2}\right) * (PX_n + PX_{n-1}),$$

где n – номер шага интегрирования, h – шаг интегрирования. Соответствующая система НАУ решается только методом Ньютона. В приращениях формула имеет вид:

$$PX_n = (X_n - X_{n-1}) * 2/h - PX_{n-1}.$$

Обозначим $\sum DX$ – сумму всех приращений ньютоновских итераций в процессе решения соответствующей системы НАУ. В результате формула модифицированного неявного метода трапеций в приращениях имеет вид:

$$PX_n = \sum DX * 2/h - PX_{n-1},$$

а система НАУ имеет вид:

$$F(X_n) = PX_n = \sum DX * \frac{2}{h} + PX_{n-1} = 0. \qquad (2)$$

Для тестирования того, как в программах, реализующих неявные методы интегрирования, решена эта проблема, надо выполнять согласованное изменение параметров систем ОДУ таким образом, чтобы переменные и шаги интегрирования изменялись в максимально возможных пределах разрядной сетки процессора при сохранении качественно корректных результатов интегрирования. Принципы согласованного изменения параметров системы ОДУ изложены в работе [6] и выполнялись на примере расчета линейной электрической схемы с известным аналитическим решением (RLC: high Q filter – рис.1 в [6]). В табл. 1 приведены результаты тестирования по данному методу программы manzhuk и наиболее известных программ, в которых реализованы неявные методы интегрирования: Radau-MathCAD, ode15s-MATLAB и Rosenbrock-Maple. В этой таблице приведены полученные диапазоны значений масштабных коэффициентов, предложенных в [6], для которых результаты расчета этой схемы еще совпадают с эталонным, аналитическим решением (порядок масштабных коэффициентов по времени Kt, по току Ki и по напряжению Ku при тестировании не задавался более +250 и менее -250). За пределами приведенных значений программы MathCAD, MATLAB выдают качественно неверный результат без каких-либо сообщений (Maple выдает сообщение об ошибке).

Таблица 1

Сравнение программ решателей систем ОДУ для high Q filter

Масштабные коэффициенты Ki, Ku, Kt				
Для всех	Radau-MathCAD	ode15s-MATLAB	Rosenbrock-Maple	manzhuk
Ki=1,Ku=1	1E-2<Kt<1E250	1E-80<Kt<1E250	1E-103<Kt<1E250	1E-250<Kt<1E250
Kt=1,Ku=1	1E-250<Ki<1E250	1E-250<Ki<1E2	1E-250<Ki<1E250	1E-250<Ki<1E250
Kt=1,Ki=1	1E-2<Ku<1E250	1E-1<Ku<1E7	1E-250<Ku<1E250	1E-250<Ku<1E250

Одна из важнейших задач при реализации метода Ньютона для решения системы НАУ (2) - это проблема вычисления Якобиана для вектор-функции $F(X_n)$. Многочисленные эксперименты на разнообразных тестовых задачах показали, что надо максимально возможно использовать аналитическое вычисление элементов Якобиана, а при численном вычислении применять методы четного порядка точности, вычисляя приращение по формуле $2\sqrt{epsm} * abs\ (x_i)$, где $epsm$ - компьютерная точность вычислений операции сложения, x_i – текущее значение соответствующей переменной [10].

IV. Новые алгоритмы для решения систем ЛАУ с гарантированной компьютерной точностью

На каждой итерации метода Ньютона решается соответствующая система ЛАУ, достоверность и точность решения которой определяет точность конечного результата решения системы ДАУ, особенно это проявилось при решении задач нуклидной кинетики высокой размерности [7]. Математическая модель нуклидной кинетики базируется на системе ОДУ первого порядка, представляемых уравнениями вида

$$\frac{dX}{dt} = A * X, X_0 \neq 0, \tag{3}$$

где A – матрица эффективных скоростей реакций накопления и увода элементов вектора X, размерность которого определяется количеством рассматриваемых элементов (нуклидов). Для решения задач нуклидной кинетики программы семейства ORIGEN на протяжении более 30-и лет остаются наиболее востребованными. В программе ORIGEN2, формально не имеющей ограничений на число элементов и структуру переходов, используются разные подходы для вычисления элементов вектора X. Для вычисления короткоживущих элементов используется метод Гаусса-Зейделя, для остальных решение

$$X(t) = X_0 * e^{At}$$

находится с помощью вычисления матричной экспоненты в виде ряда Тейлора. При этом увеличение параметров точности для программы ORIGEN2 дает постоянно разные результаты, что очевидно является результатом влияния погрешностей округления для элементарных арифметических операций, как и в вышеприведенном примере вычисления числа π. Поэтому выбор правильного решения системы (3) возлагается на опытного пользователя. При этом формируемая матрица коэффициентов имеет размерность 3000x3000 и оказывается несимметричной с числом обусловленности $\mu(A)$ до 10^{27} , что приводит к принципиальным математическим сложностям получения решений. Поскольку для конкретных значений μ(A) и размерности задачи численная оценка применимости используемого метода весьма затруднительна, получение гарантированно точного решения принципиально важно в отсутствие контрольной тестовой задачи с точным аналитическим решением. Следует отметить, что известные контрольные тестовые задачи имеют невысокую размерность и влияние погрешностей округления на решение таких задач незначительно. Таким образом, процесс получения численного решения вне зависимости от способа должен обеспечивать достоверность и компьютерную точность решения. Для решения поставленной задачи использовалась библиотека MZK, как развитие библиотеки SADEL, в которой реализованы AL устойчивые неявные методы с переменным шагом интегрирования и алгоритмы вычислений с гарантированной точностью [3]. В библиотеке MZK также реализованы алгоритмы вычислений, учитывающие влияние погрешностей округления на конечный результат расчетов. Требование компьютерной точности подразумевает получение заданного количества верных десятичных знаков мантиссы для каждого компонента нормализованных чисел вектора решения. Универсальным способом контроля достоверности n верных десятичных значащих цифр является применение следующего алгоритма:

1. Интегрирование с исходными параметрами достижения точности (например, с выбранной величиной математической относительной погрешности).

2. Повторное интегрирование с более жёсткими параметрами (например, с меньшей математической относительной погрешностью).

3. Сравнение полученных результатов п. 1 и 2 на совпадение первых n знаков во всех компонентах вектора решения и выполнение п. 1, 2 до совпадения требуемых n десятичных значащих цифр.

4. Для контроля отсутствия влияния на формирование n значащих цифр результата погрешностей округления, обусловленных размерностью и (или) жесткостью системы, расчёт с параметрами интегрирования п.3 повторяется с повышенной разрядностью и последующим сравнением с результатом п.3. При расхождении в пределах n десятичных знаков расчёты п.1 и 2 выполняются с повышенной разрядностью. Выполнение п.4 принципиально невозможно в стандартных вычислительных пакетах, в которых не поддерживается решение систем ЛАУ с повышенной разрядностью чисел. Для плохо обусловленных систем ЛАУ точное решение можно получить с помощью символьных вычислений, например, в MATLAB с помощью функции vpa [8]. Недостатком такого подхода является резкое увеличение времени счета при увеличении размерности задачи. Наши исследования показали, что итерационное уточнение с обычной для MATLAB точностью решает эту проблему, но правая часть систем ЛАУ должна быть вычислена с повышенной разрядностью. Опишем алгоритм итерационного уточнения решения для систем ЛАУ вида $Ax = b$.

Пусть Xp – приближенное решение, R – соответствующая невязка. В работе [9] показано, что точность решения можно существенно повысить, итерационно улучшая его следующим образом:

• вычисляем с повышенной разрядностью невязку: $R = b - AXp$;

• вычисляем приращение dX для СЛАУ $AdX = R$;

• $X = X + dX$

Итерационный процесс продолжается до достижения требуемой точности — относительного значения $abs(\frac{dX}{X})$, либо до указанного априори числа итераций. Вычисление dX можно значительно ускорить при использовании LU разложения матрицы A . Реализованный нами в MATLAB этот алгоритм обеспечивает достижение требуемой точности с помощью итерационного процесса уточнения. Число итераций улучшения достаточно ограничить 7.

V. Результаты расчетов тестовых систем

Для тестовой задачи с матрицей 1779x1779 и $\mu(A) = 10^{27}$) решение было получено с помощью пакета MZK, в котором реализованы AL устойчивые неявные методы M1, M2 и M4 [3]. Среди неявных методов интегрирования минимальное влияние погрешностей округления имеют одностадийные неявные методы Рунге-Кутты – методы Эйлера и трапеций, коэффициенты которых абсолютно точно представляются в двоичной арифметике с плавающей точкой, в отличие от методов высокого порядка точности. Для получения гарантированно точного решения задач нуклидной кинетики в пакете MZK реализован одностадийный (в реализации неявных методов Рунге-Кутты) неявный метод Эйлера, как с удвоенной, так и учетверенной точностью вычислений, этим методом было получено гарантированно точное решение для заданной относительной погрешности eps = 10^{-3}. Точность решения была проверена с помощью вычислений в quadruple и double precision. В табл. 2 и 3 представлены значения отдельных компонент вектора численного решения ОДУ с матрицей 1779x1779, вычисленные с обычной и повышенной разрядностью. Совпадающие в решениях значащие цифры (табл. 2) можно признать верными. Совпадающие во всех трёх методах цифры (табл. 3), можно считать гарантированно верными (минимальное значение в последней колонке).

Решение этой задачи неявным методом переменного порядка точности ode15s в MATLAB при соответствующем подборе параметров точности RelTol и AbsTol дало в итоге полное совпадение порядков и первых двух значащих цифр мантиссы для всех переменных решения с решениями, полученными с помощью программ библиотеки MZK, что подтвердило правильность расчетов с помощью разработанных нами алгоритмов и программ.

VI. Выводы

1. При решении систем ДАУ высокой и сверхвысокой размерности следует применять неявные методы интегрирования, применяя на всех этапах выполнения алгоритмов вычисления с повышенной разрядностью чисел, чтобы исключить влияние погрешностей округления на конечный результат вычислений.

2. При решении систем НАУ следует использовать метод Ньютона с максимально возможным аналитическим вычислением элементов Якобиана, а при численном вычислении применять методы четного порядка точности, вычисляя приращение по формуле $2\sqrt{epsm} * abs(x_i)$, где $epsm$ – машинная точность вычислений операции сложения, x_i – текущее значение соответствующей переменной.

3. При решении систем ЛАУ следует обязательно учитывать разреженность соответствующих матриц по принципу – хранить только ненулевые элементы и вычисления проводить только с ненулевыми элементами, установив для них соответствующий минимальный барьер, ниже которого все числа считаются нулевыми.

4. Обязательно реализовать неявный метод Эйлера для получения всегда заведомо точного и достоверного решения соответствующей системы ДАУ с апериодическими решениями и неявный метод трапеций для получения всегда заведомо точного и достоверного решения соответствующей системы ДАУ с быстро осциллирующими решениями.

5. Во всех алгоритмах следует использовать оценку только относительной математической погрешности, ограничив абсолютную математическую погрешность минимально возможным числом для используемой погрешности вычислений элементарных арифметических операций.

Таблица 2

Определение верных значащих цифр мантиссы нормализованных чисел результатов при вычислении с повышенной разрядностью (eps = 10−3)

Компонент	Начальное значение, отн. ед.	Конечное значение, отн. ед.		Число верных значащих цифр
		Разрядность		
		Double (64 бит)	Quadruple (128 бит)	
56	0	$9.6208352 \cdot 10^{-12}$	$9.6208301 \cdot 10^{-12}$	6
82	8.5291	8.528585083555638	8.528585083556097	12
88	0	$4.28207 \cdot 10^{-15}$	$4.28210 \cdot 10^{-15}$	4
102	0.012503	0.012504453607224	0.012504453607233	12
306	0	$5.369878 \cdot 10^{-55}$	$5.370170 \cdot 10^{-55}$	2

Таблица 3

Определение верных значащих цифр при вычислении разными методами пакета MZK (quadruple precision)

Компонент	Метод			Число верных значащих цифр
	M1 $\varepsilon = 1 \cdot 10^{-3}$ $h_0 = 1 \cdot 10^{-12}$	M2 $\varepsilon = 1 \cdot 10^{-3}$ $h_0 = 1 \cdot 10^{-10}$	M2 $\varepsilon = 1 \cdot 10^{-5}$ $h_0 = 1 \cdot 10^{-10}$	M1 и M2 ($\varepsilon = 1 \cdot 10^{-3}$) / M2 ($\varepsilon = 1 \cdot 10^{-3}$ и $\varepsilon = 1 \cdot 10^{-5}$)
56	$9.621150 \cdot 10^{-12}$	$9.621896 \cdot 10^{-12}$	$9.621895 \cdot 10^{-12}$	4/6
78	4.856970853893407	4.856970853765135	4.856970853765135	10/16
82	8.528585083553850	8.528585083548615	8.528585083548615	11/16
88	$4.280472 \cdot 10^{-15}$	$4.276675 \cdot 10^{-15}$	$4.276673 \cdot 10^{-15}$	2/6
100	3.148138326	3.148138*3141*40	3.1481383*1414*32	8/12
101	0.15507764376	0.155077643776481	0.155077643776483	9/14
102	$1.2504453606 \cdot 10^{-2}$	$1.250445360553e \cdot 10^{-2}$	$1.250445360553 \cdot 10^{-2}$	10/16
306	$5.351 \cdot 10^{-55}$	$5.30947 \cdot 10^{-55}$	$5.30932 \cdot 10^{-55}$	2/4

Литература

[1] Маничев В.Б., Жук Д.М., Андронов А.В. Платформа математического моделирования во временной области разнородных технических систем и объектов FMS PA10. Всероссийская научно-техническая конференция "Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС)". Сборник трудов. 2010. № 1. С. 160-165.

[2] Vityaz O., Porra V. Testing of Time Domain Simulators for Nonlinear Electronic Circuits. Helsinki University of Technology, Faculty of Electrical Engineering, Electronic Circuit Design Laboratory. Report 4. Finland, July 1988. 65 p.

[3] Жук Д.М., Маничев В.Б., Сахаров М.К. Сравнение современных решателей жестких систем обыкновенных дифференциальных уравнений с решателями СИ библиотеки SADEL. Наука и образование: электронное научно-техническое издание. № 08, август 2012 DOI: 10.7463/0812.0445558 URL: http://technomag.edu.ru/doc/445558.html (дата обращения: 30.03.2020).

[4] Б Борисенко Н.Д., Жук Д.М., Маничев В.Б. Метод математического тестироавния реализация языка MODELICA, Технологии инженерных и информационных систем. 2015. № 3. С. 43-49.

[5] Manichev V.B., Zhuk D.M., Vitukov Method of mathematical testing of programs for transientanalysis in EDA packages F.A. Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2015. № 1. С. 21-22.

[6] Маничев В.Б. Новые алгоритмы для программ анализа динамики технических систем // Вестник МГТУ, сер. Приборостроение. -1996. - Вып. 1. - С. 48-56

[7] Маничев В.Б., Митенкова Е.Ф.,, Фельдман Э.О., Кожевников Д.Ю., Соловьева Е.В. Достоверность и точность решения задач нуклидной кинетики, (будет опубликовано).

[8] URL: https://www.mathworks.com/products/matlab.html (access date 04.01.2018)

[9] Уоткинс Д. Основы матричных вычислений. М.: Бином, 2006. 664 с.

[10] Higham, Nicholas J. Accuracy and stability of numerical algorithms. 2nd ed. 2002, 680 p.

Problems of Development of Mathematical Core for Technical Systems Dynamics Simulation

D. Zhuk, D Kozhevnikov, V.Manichev

zhuk@bmstu.ru, kozhevn@bmstu.ru, manichev@bmstu.ru,

BMSTU, Moscow

Abstract — **The article discusses the main statements and scientific results of many years of work of the authors in the field of dynamic systems simulation, which are the basis for the development of the library of mathematical programs MZK-a library in C for solving algebraic and differential equations with the maximum possible computer accuracy. It is planned to use the MZK library as a mathematical core for the mathematical modeling of heterogeneous dynamic systems PA10 software package (Program for Analysis version 10). It is shown that to solve systems of very high dimensions, it is necessary to take into account the errors of rounding numbers when performing elementary arithmetic operations. The most important element of computer-aided design (CAD) systems for technical objects, including microelectronics products, is mathematical modeling of objects, processes and phenomena in the time domain. The reliability and accuracy of mathematical modeling of dynamic systems depends on the mathematical core, which is a block for solving systems of differential-algebraic equations (DAU), a special case of which are systems of ordinary differential equations (odes) in normal Cauchy form, resolved with respect to derivatives. To solve these systems, implicit integration methods are used, which are reduced to solving systems of nonlinear algebraic equations (NAU) and ultimately linear algebraic equations (LAU).**

After writing the math core, the developer must test it to identify errors and determine whether the math core meets the requirements. Testing of programs for analyzing the dynamics of technical systems is carried out in two directions: testing on specific known practical problems (electrical, mechanical, hydraulic circuits, etc.) and testing of the mathematical core (systems of differential-algebraic equations (DAU). Since the main characteristics (accuracy and speed of the solution) depend on the mathematical core, its testing should confirm the reliability and efficiency of the CAD developed. Currently, the first direction is developed quite widely, especially for microelectronics CAD (ECAD/EDA systems) [2], and the second is rarely used, despite the fact that it is necessary for the further development of existing CAD systems and the development of new ones, so the authors have performed a number of works on the second direction [3]-[5]. The article deals with the main statements and scientific results of the authors ' long-term work in the field of dynamic systems modeling, which are the basis for the development of the library of mathematical programs MZK-a library in C for solving algebraic and differential equations with the maximum possible computer accuracy. It is planned to use the MZK library as a mathematical core for the mathematical modeling of heterogeneous dynamic systems PA10 software package [1]. The ODE system solvers in PA10 will exceed the accuracy and reliability of the ODE system solvers in General-purpose SPICE-type electronic circuit simulation programs (for example, the Gear and Trapezoid programs in the Multisim software package). It is shown that to solve systems of very high dimensions, it is necessary to take into account the errors of rounding numbers when performing elementary arithmetic operations, which are not taken into account in the above-mentioned General-purpose electronic circuit modeling programs.

When solving DAE systems of high and ultra-high dimensions, implicit integration methods should be used, applying the calculation algorithms with increased bit depth at all stages in order to exclude the influence of rounding errors on the final result of calculations.

When solving NAE systems, you should use the Newton method with the maximum possible analytical calculation of the Jacobian elements, and for numerical calculation, use even-order accuracy methods, calculating the increment using the formula $2\sqrt{eps} * abs(x_i)$, where eps is the machine accuracy of calculating the addition operation, x_i is the current value of the corresponding variable.

When LAE systems is solved, it is necessary to take into account the sparsity of the corresponding matrices according to the principle – store only non-zero elements and perform calculations only with non-zero elements, setting an appropriate minimum barrier for them below which all numbers are considered zero.

It is necessary to implement the implicit Euler method to obtain an always known accurate and reliable solution of the corresponding DAE systems with aperiodic solutions and the implicit trapezoidal method to obtain the always known accurate and reliable solution of the corresponding DAE systems with fast oscillating solutions.

Keywords — **computer-aided design of electronic circuits, mathematical modeling, dynamical systems, ordinary differential equations, differential-nonlinear algebraic equations, linear algebraic equations**

References

[1] Manichev V. B., Zhuk D. M., Andronov A.V. PLATFORM for MATHEMATICAL MODELING in the TIME

DOMAIN of HETEROGENEOUS TECHNICAL SYSTEMS AND OBJECTS FMS PA10. All-Russian scientific and technical conference "Problems of development of advanced micro-and nanoelectronic systems (MES)". Collection of works. 2010. # 1. Pp. 160-165.

[2] Vityaz O., Porra V. Testing of Time Domain Simulators for non-linear Electronic Circuits. Helsinki University of Technology, Faculty of Electrical Engineering, Electronic Circuit Design Laboratory. Report 4. Finland, July 1988. 65 p.

[3] Zhuk D. M., Manichev V. B., Sakharov M. K. COMPARISON of MODERN SOLVERS of RIGID systems of ORDINARY DIFFERENTIAL EQUATIONS WITH SI SOLVERS of the SADEL LIBRARY. Science and education: electronic scientific and technical publication. No. 08, August 2012 DOI: 10.7463/0812.0445558 URL: http://technomag.edu.ru/doc/445558.html (accessed: 30.03.2020).

[4] b Borisenko N. D., Zhuk D. M., Manichev V. B. METHOD of MATHEMATICAL TESTING of MODELICA LANGUAGE IMPLEMENTATIONS, engineering and information systems Technologies. 2015. # 3. Pp. 43-49.

[5] Manichev V. B., Zhuk D. M., Vitukov METHOD OF MATHEMATICAL TESTING OF PROGRAMS FOR TRANSIENT ANALYSIS IN EDA PACKAGES F. A. Problems of development of advanced micro-and nanoelectronic systems (MES). 2015. # 1. Pp. 21-22.

[6] Manichev V. B. New algorithms for programs for analyzing the dynamics of technical systems. Vestnik MSTU, ser. Instrument making. -1996. - Vol. 1. - Pp. 48-56

[7] Manichev V. B., Mitenkova E. F., Feldman E. O., Kozhevnikov D. Yu., Solovieva E. V. Reliability and accuracy of solving nuclide kinetics problems, (to be published).

[8] URL: https://www.mathworks.com/products/matlab.html (access date 04.01.2018)

[9] Watkins D. Fundamentals of matrix computing. Moscow: Binom, 2006. 664 PP.

[10] Higham, Nicholas J. Accuracy and stability of numerical algorithms. 2nd ed. 2002, 680 p.