

УДК 629.7.016

А.Б.Никитенко (4 курс, каф. ИСУ), В.В.Кучинский, д.ф.-м.н., проф. НИПГС

МАГНИТОПЛАЗМОХИМИЧЕСКИЙ ДВИГАТЕЛЬ НА ОСНОВЕ ПОЛОЖЕНИЯ КОНЦЕПЦИИ «АЯКС»

Наиболее ответственной системой ГЛА является двигатель. На данном этапе работа направлена на математическое моделирование процессов в камере сгорания. Для описания потока в двигателе используется трехжидкостная модели горения. Одно из уравнений модели, описывающее движение электронной компоненты плазмы:

$$ne me \left(\left(\frac{\partial}{\partial t} Ue \right) + Ue \left(\frac{\partial}{\partial x} Ue \right) \right) + C(Ue - Ua) + \left(\frac{\partial}{\partial x} Pe \right) - qe ne E = 0 ,$$

где ne – концентрация, me – масса, Ue – скорость движения, qe – заряд, Pe – плотность электронов, Ua – скорость движения атомов, C – отражает подвижность электронов.

На первом шаге была построена модель в её простейшем варианте, проведен расчет и получены первые результаты. Недостатки такой модели в том, что она чрезвычайно медленна и требовательна к памяти вычислительной машины, также не известны границы параметров её применения. Главное достижение улучшения модели - это создание класса разреженных матриц и solver (части программы, реализующей решение СЛАУ), работающего с ним. Solver - сердце расчёта и очень интересен в реализации. За счёт этого класса получена гораздо меньшая требовательность к памяти и времени расчёта. Скорость расчёта средних моделей повысилась на три порядка, а объёмы памяти в рассчитываемом пространстве параметров теперь зависят практически линейно от увеличения размерности задачи, а не квадратично, как ранее. Видится возможность дальнейшего увеличения скорости, при этом теоретически придётся жертвовать требовательностью к памяти, примерно, на 50%.

Одним из интересных моментов в ходе проведения работы является автоматическая генерация кода, используя математический пакет MAPLE 6. Автоматическая генерация необходима, т.к. нужно исключить человеческий фактор из написания кода вручную, система базовых дифференциальных уравнений в первоначальном виде и так уже довольно сложна, после перехода к разностной схеме структура ещё усложняется, а в результате дифференцирования для оптимизации (нахождения решения) вид уравнений таков, что занимает не один печатный лист.

Дальнейшим шагом было написание модуля визуализации результатов. Были рассмотрены варианты написания вручную, написание с использованием библиотеки 3D графиков, и, наконец, использование готового пакета прикладных программ по математике MathCad11. Был выбран последний вариант, т.к. он требует меньше времени для реализации, хотя и менее гибкий в применении. MathCad11 обладает хорошими модулями графиков, которые обеспечивают достаточно информативное представление результатов.

После решения проблемы визуализации появился вопрос об области применения данной модели. Т.к. сами процессы, описанные построенной моделью очень сложные, системы, которые получаются в результате расчётов, очень жёсткие. Большие размеры систем также приводят к вычислительным трудностям. В связи с этим, не при всех значениях параметров модель адекватно рассчитывается. Эмпирически были определены границы пространства параметров, при которых расчёт возможен. Это пространство оказалось невыпуклым, до конца не понятно, с чем это связано. После анализа результатов оказалось,

что параметры, при которых модель наиболее интересна, находятся вне этого пространства. Модель реализовывалась на разных разрядных сетках, с увеличением разрядной сетки замечено расширение границ применимости модели, поэтому появилась необходимость написать свой тип данных, отличный от стандартных типов с плавающей точкой, предлагающихся в среде разработки Borland Builder C++ 6.0. В первом приближении новый тип должен иметь ёмкость 40 знаков против 18 стандартных. Его недостатки - это время выполнения, операции и функции выполняются на порядки дольше, в зависимости от их сложности. Для уменьшения этого времени необходима реализация на языке программирования более низкого уровня (ассемблер).

Параллельно с написанием модуля увеличения разрядной сетки ведётся проверка аналитических выражений, логики работы модели и правильности реализации кода. Необходимость этого диктуется сложностью и обширностью модели.