Министерство образования и науки Российской Федерации

КАЗАНСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ имени А.Н.ТУПОЛЕВА - КАИ

Кафедра автоматики и управления

Пояснительная записка к курсовой работе

по дисциплине "ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ"

на тему

Моделирования движения сверхзвукового самолета методом прогноза и коррекции

Выполнил студент группы 3139

(подпись) (Фамилия И.О.)

Руководитель

(должность, подпись) (Фамилия И.О.)

Оценка

ф

Оценка

(Фамилия И.О.)

(подпись)

Казань, 2015

Оглавление

[Введение 3](#_Toc418080008)

[1. Постановка задачи 4](#_Toc418080009)

[2. Метод решения 5](#_Toc418080010)

[3. Блок-схема алгоритма 6](#_Toc418080011)

[4. Текст программы 7](#_Toc418080012)

[5. Описание программы 7](#_Toc418080013)

[5.1. Назначение программы 7](#_Toc418080014)

[5.2. Исходные параметры программы 7](#_Toc418080015)

[5.3. Выходные параметры 7](#_Toc418080016)

[5.4. Структура программы 7](#_Toc418080017)

[5.5. Инструкция по использованию программы 8](#_Toc418080018)

[6. Контрольный пример 8](#_Toc418080019)

[7. Исходные данные и результаты счета 10](#_Toc418080020)

[8. Сравнение результатов счета с использованием пакета MathCad 10](#_Toc418080021)

[Заключение и выводы 14](#_Toc418080022)

[9. Список использованной литературы 15](#_Toc418080023)

[Приложение 1 16](#_Toc418080024)

## Введение

Во многих областях науки и техники, а также отраслях наукоемкой промышленности, таких как: авиационная, космическая, химическая, энергетическая - являются весьма распространенные задачи прогноза протекания процессов, с дальнейшей их коррекцией.

Решение такого рода задач связано с необходимостью использования численных методов, таких как: метод прогноза и коррекции, метод Адамса-Башфорта, метод Эйлера, метод Рунге-Кута, и др. При этом, стоит задача решения системы линейных дифференциальных уравнений первого порядка одним из методов интегрирования, на произвольном промежутке времени.

Разработка программных средств реализующих расчет точного прогноза протекания процессов, является важнейшей вспомогательной научно-технической задачей.

В данной курсовой работе для расчета движение сверхзвукового самолета применяется метод прогноза и коррекции [1], который реализуется на ПК в виде программы на языке Паскаль.

## Постановка задачи

Дано: траектория полета действующих сил, изображение которой приведена на рис.1, система дифференциальных уравнений 1-го порядка(1),(2), начальные условия, шаг интегрирования, начальная и максимальная высота.

 (1)

 (2)

где

 - ускорение свободного падения

*W* - вес самолета

*V* - скорость самолета

 - угол, который траектория составляет с горизонтом

 - сопротивления воздуха

 - подъемная сила

 - тяга самолета

 - плотность воздуха

Требуется составить математическую модель, разработать программу для численного решения уравнения модели методом прогноза и коррекции, составить и отладить программу для отображения траектории самолета на экране монитора, провести сравнение решения задачи с использование стандартных средств пакета MathCad.

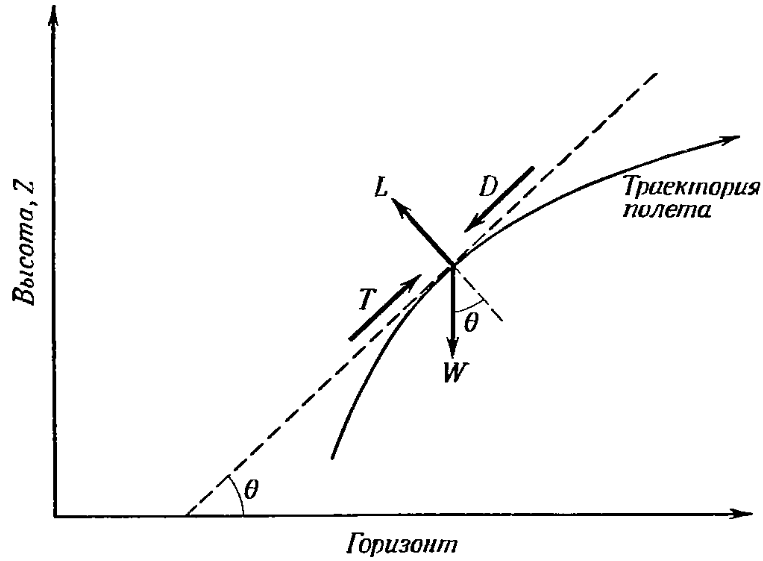


Рис.1. Силы, действующие на идеализированный самолет [1]

## Метод решения

Используя исправленный метод Эйлера, находит значения ,. [1, с.429]

,(3)

, (4)

уравнения (3),(4) будем решать одновременно, так и в (3), и в (4) входят обе неизвестных

После того как с помощью исправленного метода Эйлера вычислены точки , можно для дальнейших вычислений использовать метод прогноза и коррекции. Для m=1,2, ...

Значения , предсказываются с помощью формул (5) [1, с.430]

. (5)

Коррекция значений , производится следующим образом: [1, с.430]

, (6)

, (7)

коррекция производится до тех пор, пока , где - заданная точность. [1, с.415]

Далее мы находим ошибку ограничения с помощью соотношений(8). [1, с.431]

 (8)

Оценки ошибки ограничения печатаются, а также используются для учета окончательной поправки в значениях , согласно формуле (9),(10). [1, с.424]

, (9)

. (10)

## Блок-схема алгоритма

Да

Нет



Нет

Да

V2,R2,Z2,

график

Z2=Z2+h

Произвести коррекцию значений с помощью формул(6),(7)

Предсказать значения V2,R2 с помощью форулы (5)

Z2<Hmax

Z2=Z1+h

Вычислить V1,R1 с помощью формулы Эйлера(3),(4)

Z1=Z0+h

G,w,Cd,Cl,SV0,R0,Z0,h

## Текст программы

По представленному в п.3 алгоритму была разработана программа на языке C++. Текст программы с необходимыми комментариями приведен в Приложении 1.

## Описание программы

### . Назначение программы

Программа предназначена для расчета и построение траектории полета сверхзвукового самолета. Параметры программы приведены ниже.

### . Исходные параметры программы

 - переменная вещественного типа - ускорение свободного падения;

- переменная целого типа - вес самолета;

- переменная вещественного типа - коэффициент сопротивления;

- переменная вещественного типа - коэффициент подъемной силы;

- переменная целого типа - площадь крыльев;

- переменная целого типа - начальная скорость;

- переменная целого типа - начальный угол;

- переменная целого типа - начальная высота;

- переменная целого типа - максимальная высота;

- переменная целого типа - шаг интегрирования.

### . Выходные параметры

- переменная целого типа - скорость в определенный момент времени;

- переменная целого типа - угол отклонения от горизонта в определенный момент времени;

- переменная целого типа - высота над горизонтом;

График траектории движения сверхзвукового самолета.

### 5.4. Структура программы

Вычисление плотности воздуха

void ro(float z0, float ro)

Входные параметры:

z0 - высота

Выходные параметры:

ro - плотность воздуха

Вычисление силы тяги

void t(float z0, float v0, float t);

Входные параметры:

z0 - высота

v0 - скорость

Выходные параметры:

t - сила тяги

### 5.5. Инструкция по использованию программы

После запуска программы на выполнение и выдачи приглашений вводятся исходные параметры самолета (с клавиатуры)

*  - ускорение свободного падения
* - вес самолета
*  - коэффициент сопротивления
*  - коэффициент подъемной силы
*  - площадь крыльев
*  - начальная скорость
*  - начальный угол
* - начальная высота
* - максимальная высота
* - шаг интегрирования
* - точность
* *n* - количества итераций

После ввода всех параметров самолета, программа выдает результаты расчета в виде таблиц с заголовками и график траектории.

### Контрольный пример

Отладка программы проводилась на примере из [1], с типовыми коэффициентами

 - ускорение свободного падения

 - вес самолета

 - коэффициент сопротивления

 - коэффициент подъемной силы

 - площадь крыльев

и начальными условиями







и условиями интегрирования

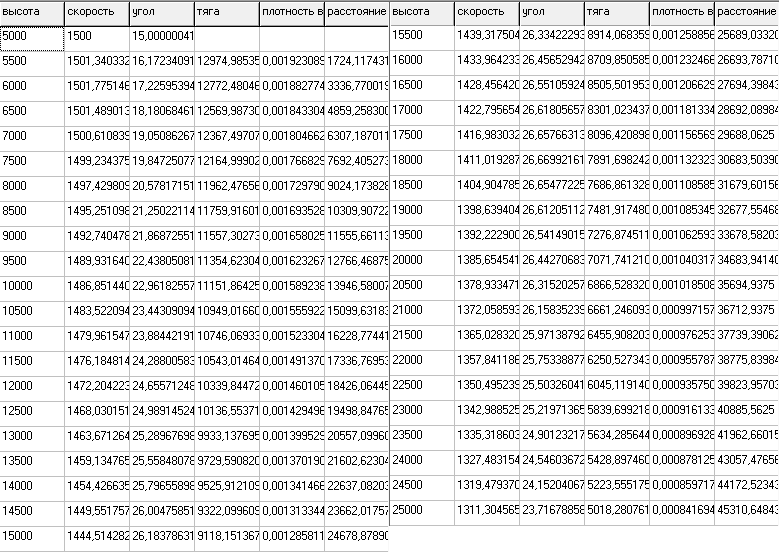




Для указанных исходных данных с помощью разработанной программы были получены следующие результаты расчета:

Таблица 1

Результаты моделирования сверхзвукового самолета



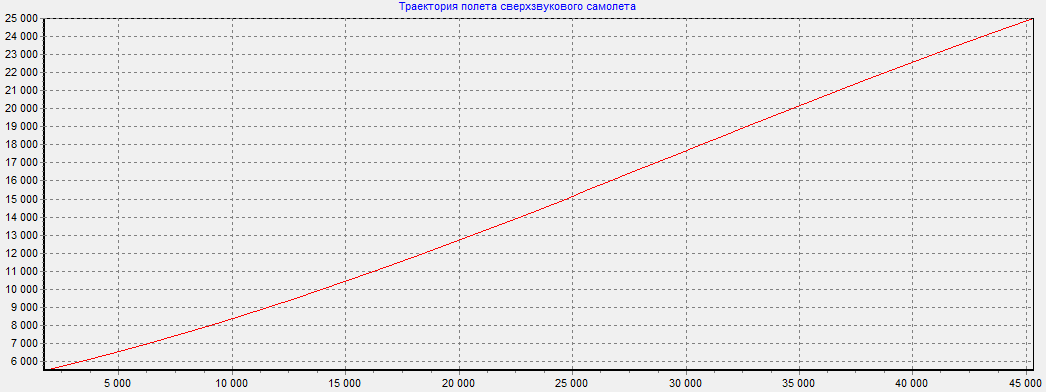


Рис. 2. Траектория полета сверхзвукового самолета

Эти результаты совпадают с результатами, приведенными в [1] для рассматриваемого примера движение сверхзвукового самолета. Поэтому можно считать, что разработанная программа работоспособна.

### Исходные данные и результаты счета

Пусть задано движение сверхзвукового самолета, приведенный на рис. 1. Для формирования входных параметров программы составляется график полета сверхзвукового самолета.

Интерфейс программы с исходными данными и с результатами счета представлено на рис. 3:

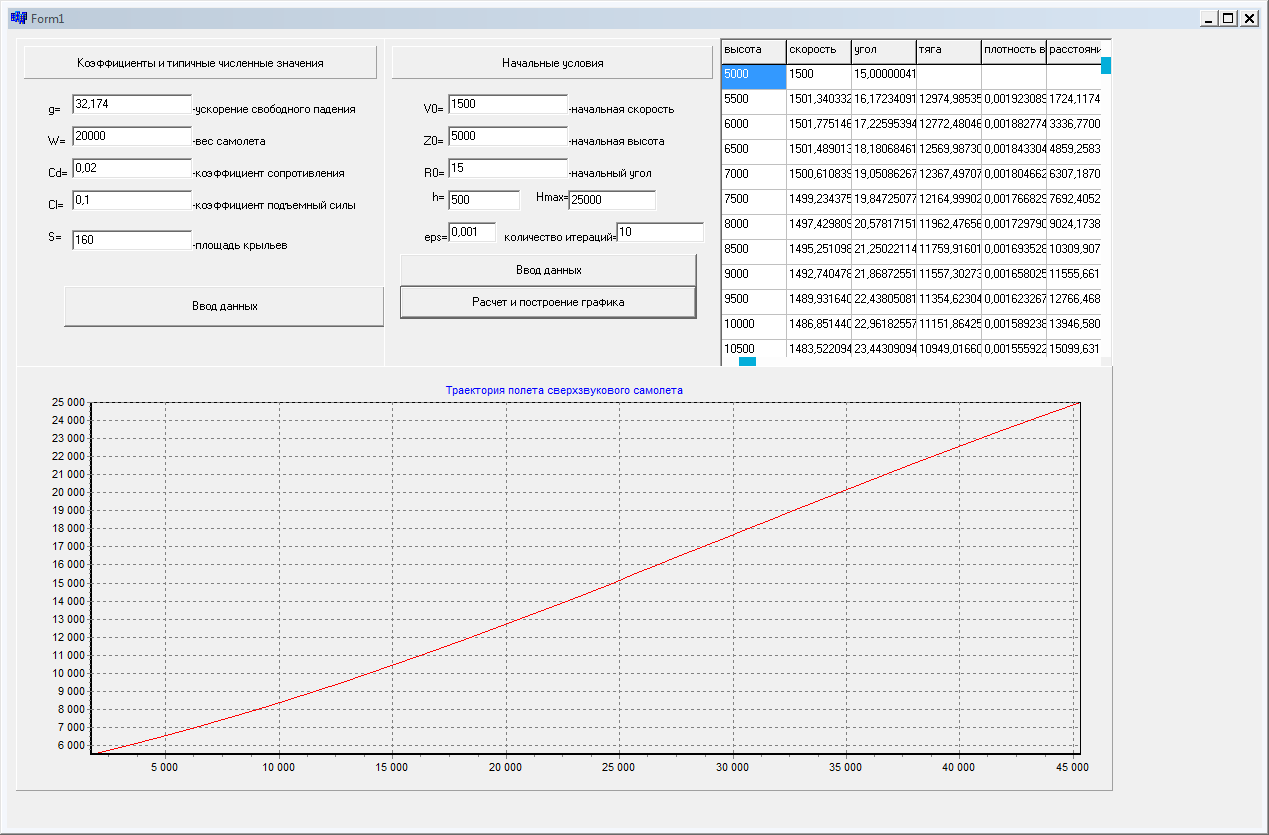


Рис. 3. Интерфейс программы с исходными данными и с результатами счета

### Сравнение результатов счета с использованием пакета MathCad

Для нахождения решения движение сверхзвукового самолета были использованы стандартные средства пакета MathLab. Для этого была составлена программа на входном языке пакета MathLab, с помощью которой было найдено решение. Эти значения совпадают со значением, полученным с помощью разработанной программы для контрольного примера. Текст программы для пакета MathLab приведен ниже:

1. Задание исходных данных

-Ускорение свободного падения











- коэффициент сопротивления

-Вес самолета

- площадь крыльев

- коэффициент подъемной силы

1. Начальные условия



1. Дифференцирование системы однородных уравнений





1. Построение таблицы значений



- расстояние, пройденное самолетом за один шаг



- перевод радиан в градусы









- расстояние, пройденное самолетом от начала координат





1. Построение траектории движения сверхзвукового самолета



Таким образом, еще раз убеждаемся, что разработанные программы правильно рассчитывают траекторию движения самолета.

### Заключение и выводы

В данной курсовой работе были разработаны алгоритм и программа на языке C++ для моделирования траектории движения сверхзвукового самолета. Проведена отладка программы на контрольном примере. Полученные с помощью разработанной программы результаты для исходных данных контрольного примера совпали с результатами, приведенными литературе. Это говорит о правильности работы программы.

Проведен расчет движения сверхзвукового самолета при других исходных данных. Проведено сравнение полученных с помощью разработанной программы для контрольного примера результатов с результатами расчетов, проведенных в пакете MathCAD 2000. Из результатов сравнений видно, что полученные значения.

### Список использованной литературы

1. Мак-Кракен Д., Дорн У. Численные методы и программирование на фортране. – М.:Мир. - 1977. - 584 с.

### Приложение 1

Текст программы на языке С++ для расчета движение сверхзвукового самолета методом прогноза и коррекции

**Unit1.h**

//---------------------------------------------------------------------------

#ifndef Unit1H

#define Unit1H

//---------------------------------------------------------------------------

#include <Classes.hpp>

#include <Controls.hpp>

#include <StdCtrls.hpp>

#include <Forms.hpp>

#include <ExtCtrls.hpp>

#include <Chart.hpp>

#include <Grids.hpp>

#include <TeEngine.hpp>

#include <TeeProcs.hpp>

#include <Series.hpp>

#include <math.h>

//---------------------------------------------------------------------------

class TForm1 : public TForm

{

\_\_published: // IDE-managed Components

TPanel \*Panel1;

TPanel \*Panel2;

TEdit \*Edit1;

TEdit \*Edit2;

TEdit \*Edit3;

TEdit \*Edit4;

TEdit \*Edit5;

TLabel \*Label1;

TLabel \*Label2;

TLabel \*Label3;

TLabel \*Label4;

TLabel \*Label5;

TLabel \*Label6;

TLabel \*Label7;

TLabel \*Label8;

TLabel \*Label9;

TLabel \*Label10;

TButton \*Button1;

TPanel \*Panel3;

TPanel \*Panel4;

TEdit \*Edit6;

TEdit \*Edit7;

TEdit \*Edit8;

TLabel \*Label11;

TLabel \*Label12;

TLabel \*Label13;

TLabel \*Label14;

TLabel \*Label15;

TLabel \*Label16;

TButton \*Button2;

TButton \*Button3;

TStringGrid \*StringGrid1;

TChart \*Chart1;

TLineSeries \*Series1;

TEdit \*Edit9;

TLabel \*Label17;

TLabel \*Label18;

TEdit \*Edit10;

TEdit \*Edit11;

TLabel \*Label19;

TEdit \*Edit12;

TLabel \*Label20;

void \_\_fastcall Button1Click(TObject \*Sender);

void \_\_fastcall Button2Click(TObject \*Sender);

void \_\_fastcall Button3Click(TObject \*Sender);

private: // User declarations

public: // User declarations

\_\_fastcall TForm1(TComponent\* Owner);

};

//---------------------------------------------------------------------------

extern PACKAGE TForm1 \*Form1;

//---------------------------------------------------------------------------

#endif

**Unit1.cpp**

//---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "Unit1.h"

#include <math.h>

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

TForm1 \*Form1;

float g,w,cd,cl,s,v0,z0,r0,shug,hmax,eps,V1,R1,V2,R2,V2new,R2new;

int n,ro;

void ro(float z0, float ro)

{

float ro0=2.3769\*0.001;

ro=ro0\*exp(((-1)\*z0)/(2.36\*10000));

}

void t(float z0, float v0, float t);

{

t=(1+0.5\*(1/exp((v0\*0.001-1.5)\*(v0\*0.001-1.5))))\*(10000-0.27\*z0);

}

//---------------------------------------------------------------------------

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

g = StrToFloat(Edit1->Text);

w = StrToFloat(Edit2->Text);

cd = StrToFloat(Edit3->Text);

cl = StrToFloat(Edit4->Text);

s = StrToFloat(Edit5->Text);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

v0 = StrToFloat(Edit6->Text);

z0 = StrToFloat(Edit7->Text);

r0 = StrToFloat(Edit8->Text);

r0 = r0\*M\_PI/180;

shug = StrToFloat(Edit9->Text);

hmax = StrToFloat(Edit10->Text);

eps = StrToFloat(Edit11->Text);

n = StrToInt(Edit12->Text);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

StringGrid1->Cells[0][0]="высота";

StringGrid1->Cells[0][1]=FloatToStr(z0);

StringGrid1->Cells[1][0]="скорость";

StringGrid1->Cells[1][1]=FloatToStr(v0);

StringGrid1->Cells[2][0]="угол";

StringGrid1->Cells[2][1]=FloatToStr(r0/M\_PI\*180);

StringGrid1->Cells[3][0]="тяга";

StringGrid1->Cells[4][0]="плотность воздуха";

StringGrid1->Cells[5][0]="расстояние пройденное по x";

//--------------------исправленный метод Эйлера-------------------------------

float F,F1,G,G1,t,ro;

float x=0;

for (int i=2;z0<hmax;i++)

{

//нахождение плотности воздуха

float ro0=2.3769\*0.001;

ro=ro0\*exp(((-1)\*z0)/(2.36\*10000));

//нахождение силы тяги

t=(1+0.5\*(1/exp((v0\*0.001-1.5)\*(v0\*0.001-1.5))))\*(10000-0.27\*z0);

//нахождение F(V0,R0)

float D;

D=0.5\*cd\*ro\*s\*v0\*v0;

F=g\*(t-(w\*(sin(r0)))-D)/(w\*v0\*sin(r0));

//нахождение G(V0,R0)

float L;

L=0.5\*cl\*ro\*s\*v0\*v0;

G=(g\*(L-w\*cos(r0)))/(w\*v0\*v0\*sin(r0));

float Vtemp, Rtemp;

Vtemp=v0+shug\*F;

Rtemp=r0+shug\*G;

//нахождение силы тяги при новом значении скорости

t=(1+0.5\*(1/exp((Vtemp\*0.001-1.5)\*(Vtemp\*0.001-1.5))))\*(10000-0.27\*z0);

//нахождение F(V0+hF(V0,R0),R0+hG(V0,R0))

D=0.5\*cd\*ro\*s\*Vtemp\*Vtemp;

F1=g\*(t-(w\*(sin(Rtemp)))-D)/(w\*Vtemp\*sin(Rtemp));

//нахождение G(V0+hF(V0,R0),R0+hG(V0,R0))

L=0.5\*cl\*ro\*s\*Vtemp\*Vtemp;

G1=(g\*(L-w\*cos(Rtemp)))/(w\*Vtemp\*Vtemp\*sin(Rtemp));

//расчет скорости и угла с помощью метода Эйлера

V1=v0+((shug/2)\*(F1+F));

R1=r0+((shug/2)\*(G1+G));

//вывод полученных значений на экран

x=x+(shug/(sin(R1)/cos(R1)));

StringGrid1->Cells[0][i]=FloatToStr(z0+shug);

StringGrid1->Cells[1][i]=FloatToStr(V1);

StringGrid1->Cells[2][i]=FloatToStr(R1/M\_PI\*180);

StringGrid1->Cells[3][i]=FloatToStr(t);

StringGrid1->Cells[4][i]=FloatToStr(ro);

StringGrid1->Cells[5][i]=FloatToStr(x);

Series1->AddXY(x,z0+shug);

StringGrid1->RowCount = i+1;

v0=V1;r0=R1;z0=z0+shug;

}}

//---------------------------------------------------------------------------