Performance Simulation of a Ramjet Using Visual C++ Program

.George Omollo Owino, Changduk Kong***
**, **Department of Aerospace Engineering Chosun University*
375 Susuk dong, dong-gu Kwangju, Rep .of Korea
geomo123@yahoo.co.kr, cdgong@mail.chosun.ac.kr

Keywords: Ramjet, Performance Simulation, Simulation code, Visual C++,

Abstract

This paper presents on research findings of how Visual C++ program can be used to generate codes capable of performing ramjet engine simulation

To understand the diversity and applicability of this tool an arbitrary ramjet model will be considered for which generated output values will be compared with those from a commercial program GASTURB 9 iterated under the same input parameters.

Several governing thermodynamic equations will first be discussed in order that we understand the fundamental idea behind values printed out on the GUI.

C++ compiler was chosen as a tool of use due to its availability, ease of use, ability to compute functions faster and uniquely possible to make a stand alone GUI executable in DOS mode.

The program is developed in such a way that given the ambient flight conditions, burner exit temperature and several geometry areas the program generates its own input values used in the succeeding stations.

A close resemblance of output values that define performance and thermodynamic state of the engine was realized between GASTURB 9 and using this code made from C++ compiler.

1. Introduction

It is a necessity in the design of a ramjet engine to know pressures, temperature, velocities and flow areas at each point along the gas path as they are used to estimate stage performance of the given engine.

Various analytical methods of varying degree of accuracy are in common usage for these calculations. It is of general practice to assume that the flow we will consider here is one-dimensional across the passage to avoid complex flow patterns in actual flow regimes.

We will consider the average specific heat method of calculation to try and limit errors most prevalent in other simpler methods like arbitrary and constant specific methods.

It would be more easy and accurate to evaluate defined dynamic model with given known values of K (pressure loss coefficient) C_d (nozzle discharge coefficient) and other parameters only determined after research. However though we shall assume ideal conditions where such values are needed.

2.1 Model geometry

A convergent-divergent ramjet model with station numbering as used in this paper is represented in figure 1 below

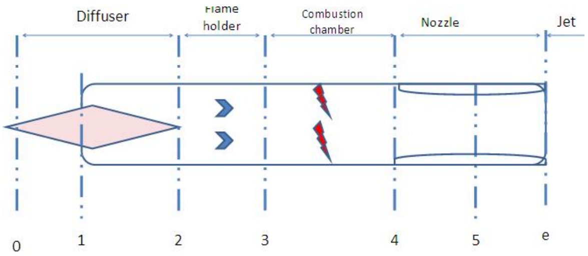


Fig. 1, Ramjet model station numbering

Respective model areas of interest necessary for calculating air mass flow and nozzle area ratio are given in the following table 1 below.

Table 1, (Area of relevance)

Area	1 (Inlet)	5 (Nozzle throat)	e (Nozzle exit)
M^2	0.0257	0.033	0.0615

2.2 Operational condition

Flight and environmental conditions at 50000ft was selected to be the operational altitude which gives us the environmental ISA conditions values of pressure, temperature, and density P , T , ρ respectively.

Table 2, (Ambient conditions at 50000ft)

Parameter	Temperature	Pressure	Mach no
	216.65 (K)	12112 (N/M^3)	3

2.3 Theory

We calculate total pressure, temperature and inlet air mass flow rate for Mach number 3 taken to be our flight mach number. These values are used to calculate different states of stage 1.

$$\frac{T_t}{T} = 1 + \frac{\gamma - 1}{2} M^2 \quad \text{and} \quad \frac{P_t}{P} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \qquad (1)$$

$$\dot{M} = M \rho A \qquad (2)$$

Similarly stage 2 diffuser downstream conditions calculated for static, total temperature and pressures. Diffuser pressure ratio and burner entry Mach number may also be determined at this stage.

$$P_t = P\left(\frac{T_t}{T}\right)^{\frac{\gamma}{\gamma-1}} \quad , \quad T_{t2} = \frac{T_t}{(1+0.2M_2^2)} \qquad (3)$$

$$M_2 = \sqrt{\frac{1+\frac{\gamma-1}{2}M_1^2}{\gamma M_1^2 - \frac{\gamma-1}{2}}} \qquad (4)$$

Calculations for stage 3 commences with the assumption that momentum is conserved, although not practical, pressure and friction losses are assumed so small to be ignored

Flame holder drag K was taken as 1. It should however be determined experimentally

$$\frac{P_{t3}}{P_{t2}} = 1 - \frac{K}{2}\frac{\gamma M_2^2}{(1+\frac{\gamma-1}{2}M_2^2)^{\frac{\gamma}{\gamma-1}}} \qquad (5)$$

This formula opens a way to calculate stage 3 total temperature. Further assumptions made at stage 4 (combustor) are that the combustor is of constant area passage and supplied with liquid fuel.

Given combustor total exit temperature, Mach number M_4 is calculated, that also servers as reheat entry Mach number, although gamma value for air is ($\gamma = 1.4$) we will consider it to be $\gamma = 1.3$ due to temperature effect from the combustor inlet onwards

Nozzle calculations for stage 5 start with the assumption that the nozzle is choked $M_5 = 1$

$$T_5 = \frac{T_{t4}}{1+\frac{\gamma-1}{2}M_5^2} \quad , \quad V_5 = M_5\sqrt{\gamma R T_5} \qquad (6)$$

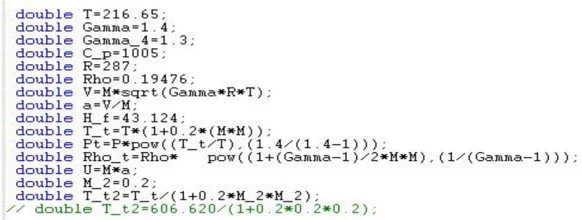
Using the area-mach relation exit Mach number is calculated

$$\frac{A_e}{A_t} = \left(\frac{\gamma+1}{\gamma}\right)^{\frac{\gamma+1}{2(\gamma-1)}} \frac{M_e}{(1+\frac{\gamma-1}{2}M_e^2)^{\frac{\gamma+1}{2(\gamma-1)}}} \qquad (7)$$

$$T_e = \frac{T_{te}}{1+\frac{\gamma-1}{2}M_e} \qquad (8)$$

Once all stage 5 is done the final stage 6 calculations start with pressure at exit taken to be equal to ambient pressure $P_e = P_{ambient}$ Different performance parameters are then determined.

All the above calculations and more form the fundamental backbone idea behind the programs working. It is at this stage that the constants and governing equations are edited into visual c++ compiler



```
double T=216.65;
double Gamma=1.4;
double Gamma_4=1.3;
double C_p=1005;
double R=287;
double Rho=0.19476;
double V=M*sqrt(Gamma*R*T);
double a=V/M;
double H_f=43.124;
double T_t=T*(1+0.2*(M*M));
double Pt=P*pow((T_t/T),(1.4/(1.4-1)));
double Rho_t=Rho*    pow((1+(Gamma-1)/2*M*M),(1/(Gamma-1)));
double U=M*a;
double M_2=0.2;
double T_t2=T_t/(1+0.2*M_2*M_2);
// double T_t2=606.620/(1+0.2*0.2*0.2);
```

Fig. 2, Visual C++ compiler inputting constants and formulae

It is easily noticeable from figure 2 the ease in which formulae is edited in this compiler. Traditional form of the formulas is maintained and does not need special editing knowledge or style to input data.

The important thing to keep in mind is the progressive order from the ambient to exit stage. it should be maintained to ensure flow as each preceding stage forms input values to the successive stage

Printing output command should be run after each formula edit to cross check its output closeness to the expected value.

Once all is finished a GUI is made to allow edit ability of input values.

3.0 Input values

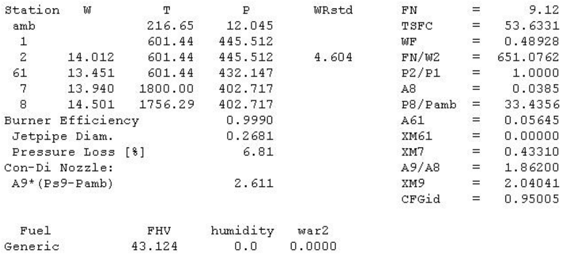
ITEM	Altitude	Mach no	Temperature	Pressure
GASTURB 9	50000ft	3	216.65	12.045
C++ GUI	50000ft	3	216.65	12.045

Table 3, input condition

Similar inputs were maintained for both cases to ensure output value comparison derived using the same preceding conditions.

3.1 GASTURB 9

This is a GUI type commercial program capable of performing steady state ramjet simulation; it involves entering the above table 3 inputs to produce thermodynamic calculation results of each stage



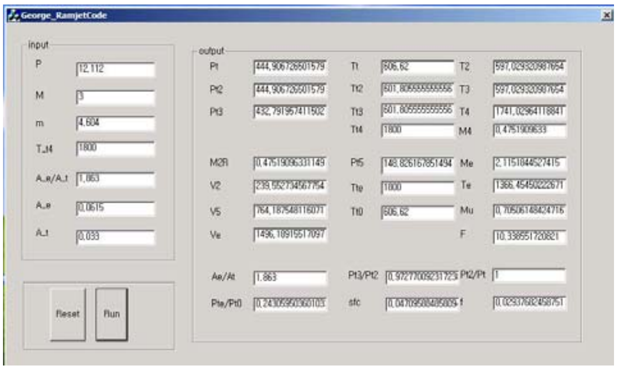
```
Station   W          T        P          WRstd       FN        =      9.12
 amb              216.65    12.045                   TSFC      =   53.6331
  1               601.44   445.512                   WF        =   0.48928
  2     14.012    601.44   445.512       4.604       FN/W2     =  651.0762
 61     13.451    601.44   432.147                   P2/P1     =    1.0000
  7     13.940   1800.00   402.717                   A8        =    0.0385
  8     14.501   1756.29   402.717                   P8/Pamb   =   33.4356
Burner Efficiency           0.9990                   A61       =   0.05645
 Jetpipe Diam.              0.2681                   XM61      =   0.00000
 Pressure Loss [%]            6.81                   XM7       =   0.43310
Con-Di Nozzle:                                       A9/A8     =   1.86200
 A9*(Ps9-Pamb)               2.611                   XM9       =   2.04041
                                                     CFGid     =   0.95005

  Fuel            FHV      humidity   war2
Generic          43.124       0.0    0.0000
```

Fig. 3, GASTURB 9 output window

3.2 Visual C++ GUI

This GUI is designed with the left hand side having editable input window with the out put on the right side



George_RamjetCode

input

P 12.112

M 3

m 4.604

T_t4 1800

A_e/A_t 1.863

A_e 0.0615

A_t 0.033

output

Pt 444.906726601579

Pt2 444.906726601579

Pt3 432.791957411502

M2R 0.47519096331149

V2 239.552734567754

V5 764.187548116071

Ve 1496.10915517097

Tt 606.62

Tt2 601.805555555556

Tt3 601.805555555556

Tt4 1800

Pt5 148.826167851494

Tte 1800

Tt0 606.62

T2 597.029320987654

T3 597.029320987654

T4 1741.02964118841

M4 0.4751909633

Me 2.1151844527415

Te 1366.45450222671

Mu 0.70506148424716

F 10.338551720821

Ae/At 1.863

Pte/Pt0 0.243059503960103

Pt3/Pt2 0.972770092317725

sfc 0.0470950840850805

Pt2/Pt 1

f 0.0293760245875 1

Fig. 4, Visual C++ Output GUI

The run button allows the program to start computing the given inputs through governing equations to generate the output values printed on the right hand side.

3.3 Output

Table 4 below shows simulation results that were generated by both programs using environmental conditions at 50000ft and Mach number 3 as input

Reference should be made to figure 1 for station numbering, although similar, different numbering method are used in Gasturb 9 from station 4

Item	Temp Gasturb	Temp Visual c	(T) Pressure Gasturb	(T)Pressure Visual c
Station 1	216.65	216.65	12.045	12.112
2	601.44	601.805	445.512	444.906
3, or 61	601.44	601.805	445.512	444.906
5 or 8	601.44	601.805	432.147	432.791
e or 9	1756.29	1741.02	-	148.826

Table 4, Thermodynamic station outputs

Item	Thrust (kN)	Inlet Press Ratio	Reheat Mach no	Nozzle Exit Mach	Nozzle Area Ratio
Gasturb	9.12	1	0.433	2.029	1.862
C++	10.33	1	0.475	2.115	1.862

Table 5, Performance output data

The difference in thrust may be attributed to the fact that Gasturb 9 considered pressure loss of 6.81% whereas we assumed ideal conditions and losses were ignored.

Mach no	1.5	2	2.5	3
Units	KN	KN	KN	KN
(Gasturb 9)	-	2.45	5	9.12
(Visual C)	8.39	9.76	10.41	10.33

Table 6, Performance with varying Mach no

Since Gasturb 9 has no input provision for intake area which was considered in Visual C++ we may assume that they used the theory that the frontal area equals exit area. Hence explains great difference in thrust at low Mach numbers

3.4 Performance

One amongst the most importance performance defining parameters is the fuel flow rate.

Calculated herein as

$$f = \frac{\left(\frac{T_{t4}}{T_{t3}} - 1\right)}{\frac{h_f}{C_p T_{t3}} - \frac{T_{t4}}{T_{t3}}} = 0.029166 \qquad (9)$$

$$M_f = 3600 f M_a \qquad (10)$$

$$sfc = \frac{M_f}{F} = 0.04679667340 \qquad (11)$$

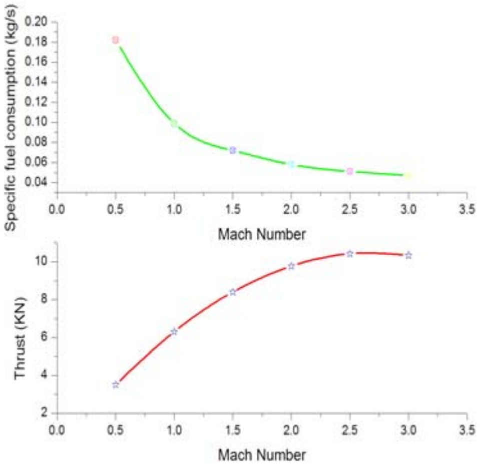


Fig. 5, Thrust and specific fuel consumption against Mach number

This graphs were plotted from performance results output of the simulation code program developed using visual C++ both graphs show expected trend

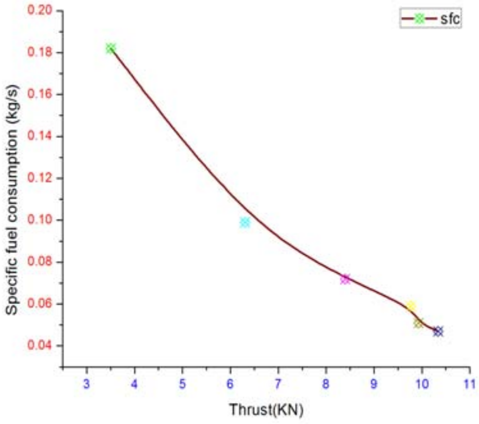


Fig. 6, Specific fuel consumption against thrust

The graph indicate that specific fuel consumption reduces with increasing thrust, this is so because specific fuel consumption is inversely related to thrust as defined in equation (11).

4.0 Conclusion

From the analysis results we experienced close or similar values of temperature and pressure for the stations considered; it would be adequate hence to conclude that Visual C++ program is accurate as may be verified by the commercial program used.

Net thrust also reflects closeness the difference of which may be attributed to frictional pressure losses considered in Gasturb 9.

This code generated by C++ compiler may be suitable to simulate defined dynamic model as it allows editing of geometric data like intake area, combustor temperatures to match desired values

Visual compiler being readily available would be a suitable tool for making simulation codes at intellectual level with additional advantage of being used for commercial purposes due to its numerous interface capabilities with other programs.

Nomenclature

(T) Pressure	Total pressure
(T) Temperature	Total temperature
T_t4	Burner exit total temperature
A_e	Exit area
A_t	Throat area
V	Velocity
Ma	Air mass flow rate

5.0 References

1) High Speed Aerodynamics and Jet Propulsion Text by O.E Lancaster 1959 pp268
2) Ramjet Gas dynamics Lecture note Cranfield University
3) C++ Programming for Engineers Text Book Fluid Dynamics Tables pp 301
4) Nasa .Glenn research website
5) http://www.nasa.gov/centers/glenn/home/index.html