

SOFTWARE FOR NATURAL GAS PIPELINE DESIGN AND SIMULATION (GASPISIM)

A. A. ADAMU

(Received 17 January 2017; Revision Accepted 17 March 2017)

ABSTRACT

There is a sudden increase in demand and utilization of natural gas in Nigeria; this may be attributed to federal government policy on establishment of independent power plants and the rising cost of kerosene and fire wood. This increase in demand for natural gas call for design and construction of pipelines for natural gas transportation from production zones to consuming areas. Natural gas transmission through pipelines is accepted as the best method for high capacities gas transmission. Due to the high capital investment and operating cost required for natural gas pipeline transmission systems, optimum designs become inevitable. Cost of compression form a significant part of the operating cost of natural gas pipeline system and can be reduced by optimum design. This paper focuses on the development of software for optimum design and simulation of natural gas pipeline. General Gas equation coupled with correlations for friction factor, Reynolds number, viscosity of gas mixture, compressibility factor etc were used to developed an algorithm from which the software was developed using VISUAL BASIC 6.0. The effect of different operating variables and pipeline dimensions could be investigated using this software. The software was tested for functionality using a typical design problem from literature. Comparison of the software result and that of literature shows that the software can be use for preliminary design and simulation purposes.

KEYWORDS: Pipeline, design, simulation, Natural gas.

INTRODUCTION

Natural gas pipeline design is an important aspect of natural gas transmission and distribution. The economy of natural gas pipeline operation largely depends among other factors on the pipeline system design. The design of natural gas pipeline system was mostly based on empirical gas flow equations, such as Weymouth, Panhandle and modified Panhandle (Paulo & Pinho, 2007). These empirical equations assumed a relationship between friction factor and other flow variables; this assumption is invalid because of the strong dependency of friction factor on Reynolds number and pipe roughness. An error in determination of the friction factor could lead over-sizing or under-sizing of compression facilities which will affect negatively the economy of the pipeline system. The current trend centered on the use of general gas equation which takes account of the

computation of the actual value of the friction factor in a design, instead of the inefficient empirical equations. This led to the use of computer programs since iterative methods are required in providing solutions to the model equations.

Many factors are usually considered in the design of gas pipelines. These include the composition and volumetric flow rate of the gas, the length of the pipeline, the type of terrain to be crossed, and maximum elevation of the route, as well as the pressure difference between the field and the main transmission line (Guo & Ghalambor, 2005). Complex engineering studies are needed to decide on diameter, yield strength, and compression power requirements to give the optimum results for any particular pipeline transmission system. In gas pipeline design, several designs options are normally considered in order to decide on the optimum flow rate, compression power requirement, pressure and

A. A. Adamu, Petroleum and Natural Gas processing department, Petroleum Training Institute, PMB 20, Effurun, Delta State, Nigeria.

pipe dimensions all of which affect the economy and safe operation of the pipeline system. Though, computer programs that enable high-pressure gas transmission networks to be dynamically simulated on a digital computer have been developed and are used by gas pipeline operating companies and service companies (Guo & Ghaleb, 2005), they are usually expensive and void of Nigerian content. Hence this paper focuses on the development of software for optimum design and simulation of natural gas pipeline.

of gas flow equations for calculating gas pipeline capacity and pressure requirements used in sizing compression facilities. There are several equations in the petroleum industry for calculating the flow of gases in pipelines. In the early development of the natural gas transmission industry, pressures were low and the equations used for design purposes were simple and adequate. However, as pressure increased to meet higher capacity demands, more complicated equations were developed to meet the new requirements. The general Gas equation which is based on Bernoulli's equation (Gas pipeline Hydraulics, 2011) is presented in the final form in U.S. Customary system of units (USCS) as follows;

2. DESIGN EQUATIONS

2.1 General Natural Gas Pipeline Equation;

Designing a long-distance pipeline for transportation of natural gas requires knowledge

$$q_s = 77.54 \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - P_2^2}{L T Z \lambda f}} \dots\dots\dots (1)$$

And the corresponding equation in SI units is given as;

$$q_s = 1.1494 \times 10^{-3} \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - P_2^2}{L T Z \lambda f}} \dots\dots\dots (2)$$

Where;

- q_s = gas flow rate, standard ft³/day (SCFD), [standard m³/day (m³/d)]
- L = pipe length, mi [km], d = inside diameter of pipe, in. [mm]
- P₁ = Absolute upstream pressure, Psia [kPa], P₂ = Absolute downstream pressure, Psia [kPa]
- P_s = Absolute base pressure, Psia [kPa], T_s = Absolute base temperature, °R [K]
- T = Absolute average flowing temperature of gas, °R [K],
- λ = gas specific gravity (Air = 1.00)
- Z = gas compressibility factor at the average gas temperature and pressure, dimensionless
- f = friction factor, dimensionless

2.2 Effect of Pipe Elevations

The effect of elevation difference between the upstream and downstream locations of the pipe (H₂ - H₁) can be included in the general gas flow equation which can then be written as follows;

$$q_s = 77.54 \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - e^s P_2^2}{L_e T Z \lambda f}} \dots\dots\dots (3)$$

Where;

S = Elevation adjustment parameter, dimensionless,

L_e = Equivalent length, mi

In SI unit the equation can be written as;

$$q_s = 1.1494 \times 10^{-3} \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - e^S P_2^2}{L_e T Z \lambda f}} \dots\dots\dots (4)$$

L_e and the term e^S depend upon the elevation difference ($H_2 - H_1$).

The term L_e and e^S are related as follows:

$$L_e = \frac{L(e^S - 1)}{S} \dots\dots\dots (5)$$

$$S = 0.0375 \lambda \left(\frac{H_2 - H_1}{TZ} \right), \quad H_2 \neq H_1 \dots\dots\dots (6)$$

Where;

T = Average temperature, R,

H_2 = Elevation at the downstream, ft

H_1 = Elevation at the upstream, ft

In SI units the corresponding equations are;

$$L_e = \frac{L(e^S - 1)}{S} \dots\dots\dots (7)$$

$$S = 0.0375 \lambda \left(\frac{H_2 - H_1}{TZ} \right) \dots\dots\dots (8)$$

Where;

T = Average temperature, K,

H = Elevation, m

The General Flow equation can be used for calculating the flow rate in a gas pipeline, given the upstream and downstream pressures. Alternatively, it can be used to calculate the pressure drop for a given flow rate.

Numerous other less accurate but simplified gas flow equations that assumed fixed relationship between Reynolds number and friction factor exist and are commonly used. Among them are; Weymouth, Panhandle and modified Panhandle Equations (Paulo & Pinho, 2007).

2.3 Weymouth equation (for horizontal pipes) is given by;

$$q_s = 433.8 \left(\frac{T_s}{P_s} \right) d^{8/3} \sqrt{\frac{P_1^2 - P_2^2}{L T Z \lambda}} \dots\dots\dots (9a)$$

q_s = Gas flow rate ft^3/day (CFD) at temperature T_s and pressure P_s
 T_s = Base temperature ($^{\circ}\text{R}$), P_s = Base pressure (psia), d = Pipe internal diameter (in)
 P_1 = Upstream pressure (psia), P_2 = Downstream pressure (psia), L = pipe length (miles)
 T = Average pipeline temperature ($^{\circ}\text{R}$), λ = Gas specific gravity
 Z = Average pipeline gas compressibility (measured at average pipeline pressure and temperature)
 Similarly the equations for Panhandle and modified Panhandle are represented by Equations 9b and 9c respectively;

$$q_s = 435.87 \left(\frac{T_s}{P_s} \right)^{1.0788} d^{2.6182} \left[\frac{P_1^2 - P_2^2}{LTZ\lambda} \right]^{0.5394} \left[\frac{1}{\lambda} \right]^{0.4604} \dots\dots\dots (9b)$$

$$q_s = 737 \left(\frac{T_s}{P_s} \right)^{1.02} d^{2.53} \left[\frac{P_1^2 - P_2^2}{LTZ\lambda} \right]^{0.510} \left[\frac{1}{\lambda} \right]^{0.49011} \dots\dots\dots (9c)$$

2.4 Reynold’s Number; The flow through a pipeline may be classified as laminar, critical (transition), or turbulent flow depending upon the value of the dimensionless parameter called the Reynolds number. The Reynolds number depends upon the gas properties, pipe diameter and flow velocity and is defined as follows;

$$N_{Re} = \frac{\rho u d}{\mu} \dots\dots\dots (10)$$

Where;
 N_{Re} = Reynolds number, dimensionless, u = average gas velocity, ft/s
 d = pipe inside diameter, ft, ρ = gas density, lb_m/ft^3 , μ = gas viscosity, $\text{lb}_m/\text{ft}\cdot\text{s}$
 In terms of the more commonly used units in the gas pipeline industry, the following formula for Reynolds number is more appropriate,

$$N_{Re} = \frac{2.001 \times 10^4 q_s \lambda}{\mu d} \dots\dots\dots (11)$$

Where;
 μ = gas viscosity, cP, λ = gas specific gravity
 q_s = gas flow rate, million standard ft^3/day (MMSCFD), d = pipe inside diameter, in.
 Another equation commonly used to calculate Reynolds’ number (Guo & Ghalambor, 2005) is as follows;

$$N_{Re} = 0.0004778 \left(\frac{P_s}{T_s} \right) \frac{q_s \lambda}{\mu d} \dots\dots\dots (12)$$

The corresponding version of the equation in SI units is as follows;

$$N_{Re} = 0.5134 \left(\frac{P_s}{T_s} \right) \frac{q_s \lambda}{\mu d} \dots\dots\dots (13)$$

Where;

P_s = base pressure, kPa , T_s = base temperature, K, q_s = gas flow rate, standard m^3/day
 d = pipe inside diameter, mm, μ = gas viscosity, Poise

The flow in a gas pipeline is considered to be laminar flow when the Reynolds number is below 2000. Turbulent flow is said to exist when the Reynolds number is greater than 4000. When the Reynolds numbers is between 2000 and 4000, the flow is called critical flow, or undefined flow.

In practice, most gas pipelines operate at flow rates that produce high Reynolds numbers and therefore in the turbulent flow regime. Actually, the turbulent flow regime is further divided into three regions known as smooth pipe flow, fully rough pipe flow and transition flow.

2.5 The friction factor

In the simplified version of the general natural gas equation such as Weymouth, Panhandle, White, Modified Panhandle etc, the relationship between the transmission factor $\frac{1}{\sqrt{f}}$ and other flow variables such as Reynolds' number and pipe roughness is defined based on certain assumptions. For example in Modified Panhandle Equation, the transmission factor for turbulent flow in rough pipes is related to Reynolds' number according to the following equation;

$$\frac{1}{\sqrt{f}} = 8.245 N_{Re}^{0.01961} \dots\dots\dots (14)$$

But the friction factor f (Darcy-Weisbach friction factor) is strongly dependent on Reynolds' number as well as pipe roughness and can be calculated from Colebrook-White equation;

The Colebrook-White equation is the most widely used for calculating friction factor in transition and turbulent flows in hydraulically smooth and rough pipes (Genić, 2011)

$$\frac{1}{\sqrt{f}} = 1.74 - 2 \log_{10} \left(2\ell_d + \frac{18.7}{N_{Re} \sqrt{f}} \right) \dots\dots\dots (15a)$$

This equation was modified by Jain in 1976 to be explicit in the friction factor (Guo & Ghalambor, 2005).

$$\frac{1}{\sqrt{f}} = 1.14 - 2 \log_{10} \left(2\ell_d + \frac{21.25}{N_{Re}^{0.9}} \right) \dots\dots\dots (15b)$$

Where;

$$\ell_d = \text{relative roughness} = \frac{\text{absolute pipe roughness}}{\text{diameter}} = \frac{\epsilon}{d} \dots\dots\dots (16)$$

For the whole range of Reynolds number and relative pipe roughness, a more accurate equation is that developed by Serghides (1984).

$$f = 0.25 \left[A - \frac{(B - A)^2}{C - 2B + A} \right]^{-2} \dots\dots\dots (17)$$

Where;

$$A = -2 \log_{10} \left(\frac{\varepsilon}{3.7d} + \frac{12}{N_{Re}} \right) \dots\dots\dots (18)$$

$$B = -2 \log_{10} \left(\frac{\varepsilon}{3.7d} + \frac{2.51A}{N_{Re}} \right) \dots\dots\dots (19)$$

$$C = -2 \log_{10} \left(\frac{\varepsilon}{3.7d} + \frac{2.51B}{N_{Re}} \right) \dots\dots\dots (20)$$

For laminar flow;

$$f = \frac{64}{N_{Re}} \dots\dots\dots (21)$$

And for smooth pipes in turbulent flow,

$$f = 0.0056 + \frac{0.5}{N_{Re}^{0.32}} \dots\dots\dots (22)$$

The major problem in using the Gas flow equations containing the friction factor term is that the friction factor is a function of Reynolds' number. If the gas flow rate is not known the friction factor can only be calculated using iteration method.

2.6 Gas Viscosity; Viscosity of a fluid relates to the resistance to flow of the fluid. Viscosity may be referred to as absolute or dynamic viscosity measured in Ns/m^2 , Pa.s, or cP and kinematic viscosity measured in m^2/s or centistokes (cSt). Other units of viscosity are $lb_m/ft-s$, $lb_f.s/ft^2$ for dynamic viscosity and ft^2/s for kinematic viscosity. Though there are many correlations for calculating viscosity of natural gas, applying Lucas method for viscosity of pure gases and

mixing rule (Reid and others, 1988) will be more appropriate when using computer program provided the critical properties of the gas components are known. Critical properties of natural gas components are available in literature (Perry & Green, 1998; Reid and others, 1988).

2.7 Gas Compressibility factor Z; Compressibility factor is a factor, which gives the ratio of the actual volume of gas at a given temperature and pressure to the volume of gas when calculated by the ideal gas law equation. Several graphical and analytical methods are available to calculate Z. Among these, the Standing-Katz, AGA and CNGA methods are quite popular (Pipeline hydraulic, 2011). The CNGA equation is as follows;

$$Z = \frac{1}{\left[1 + \left(\frac{344400 P (10^{1.785 \lambda})}{T^{3.285}} \right) \right]} \dots\dots\dots (23)$$

Where;

P = Gas average pressure, Psig.

T = Gas temperature, °R

λ = Gas specific gravity (air = 1.00)

The CNGA equation for compressibility factor is valid when the average gas pressure P is greater than 100 psig. For pressures less than 100 psig, compressibility factor is taken as 1.00. It must be noted that the pressure used in the CNGA equation is the gauge pressure, not the absolute pressure.

There are quite a number of correlations using critical properties to calculate compressibility factor, among them is Redlich-Kwong equation (Smith & Van Ness, 1987) given by;

$$Z = \frac{1}{1-h} - \frac{4.9340}{T_r^{1.5}} \left(\frac{h}{1+h} \right) \dots\dots\dots (24)$$

Where;

$$h = \frac{0.08664P_r}{ZT_r} \dots\dots\dots (25)$$

$$P_r = \frac{P}{P_c} \text{ and } T_r = \frac{T}{T_c} \dots\dots\dots (26)$$

T = temperature, K ,

P = absolute pressure, N/m²

2.8 Specific gravity of gas; The specific gravity of a fluid is defined as a ratio of the density of the fluid to that of a standard fluid such as water or air at some standard temperature. For liquids, water is the standard of comparison, while for gases air is used as the basis.

$$\lambda = \frac{\rho_g}{\rho_a} \dots\dots\dots (27)$$

Where;

λ = Specific gravity of gas, dimensionless, ρ_g = density of gas, g/cm³

ρ_a = density of air (at the same temperature), g/cm³

Being a ratio of similar properties, the specific gravity is dimensionless.

Using molecular weights, we can define the gas gravity as the ratio of the molecular weight of the gas to that of air. The molecular weight of air is usually considered to be 29.0 and therefore, the specific gravity of gas can be stated as follows:

$$\lambda = \frac{M_{wg}}{29.0} \dots\dots\dots (28)$$

Where;

M_{wg} = molecular weight of gas, g/gmol.

The specific gravity of a gas like its density varies with temperature and pressure.

2.9 Compression power requirements

The natural gas flow rate calculated from gas flow equations can be used to calculate compression power requirement for a given pipeline system. The power requirement for an adiabatic compressor (Pipeline hydraulic, 2011) in FPS units is given as;

$$\text{Power(Hp)} = 0.0857 \left(\frac{\gamma}{\gamma - 1} \right) \frac{T_{in} q_s Z}{\eta} \left[\left(\frac{P_{2c}}{P_{1c}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right] \dots\dots\dots (29)$$

T_{in} = Suction temperature of the gas in °R, q_s = gas flow rate, MMSCFD

P_{1c} = suction pressure of gas, Psia, P_{2c} = discharge pressure of gas, Psia

Z = Average compressibility of gas based on suction and discharge conditions.

η = compressor adiabatic (isentropic) efficiency, γ = Heat capacity ratio (C_p/C_v)

In SI units, the compressor Power required is as follows;

$$\text{Power(kW)} = 4.0639 \left(\frac{\gamma}{\gamma - 1} \right) \frac{T_{in} q_s Z}{\eta} \left[\left(\frac{P_{2c}}{P_{1c}} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right] \dots\dots\dots (30)$$

Where;

q_s = gas flow rate, million m^3 /day, T = temperature of gas, K, P = pressure of gas, kPa

3. DEVELOPMENT OF THE SOFTWARE (GASPSIM)

Algorithms and Computer Programs based on Visual BASIC 6.0 (Holzner, 2004; Jonnes, 2001) were developed. One of the programs is presented in the appendix. Two algorithms for design equations involving iterations are presented in simplified forms as follows;

(a) Algorithm for Natural Gas Flow Rate using General Gas Equation;

1. Assume a flow rate, q_{as}

2. Calculate Reynold's number, $N_{Re} = 0.0004778 \left(\frac{P_s}{T_s} \right) \frac{q_s \lambda}{\mu d}$

3. Calculate friction factor, $\frac{1}{\sqrt{f}} = 1.74 - 2 \log_{10} \left(2 \ell_d + \frac{18.7}{N_{Re} \sqrt{f}} \right)$

4. Use General Gas Equation to calculate gas flow rate, $q_s = 77.54 \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - P_2^2}{LTZ\lambda f}}$

5. If calculated flow rate is equal to the assumed flow rate ($q_{as} = q_s$) go to step 8, else

6. Increase the assumed gas flow rate by a small value, $q_{as} = q_{as} + \delta$

7. Go to step 2

8. Record the calculated gas flow rate

9. Stop
- (b) Algorithm for calculating downstream pressure of Gas;
 1. Assume a downstream pressure, P_{2as}
 2. Calculate the average pressure P_{av} , $P_{av} = \frac{2}{3} \left(\frac{P_1^3 - P_2^3}{P_1^2 - P_2^2} \right)$
 3. Calculate viscosity (Using Luca's method)
 4. Calculate Reynold's number, $N_{Re} = 0.0004778 \left(\frac{P_s}{T_s} \right) \frac{q_s \lambda}{\mu d}$
 5. Calculate friction factor, $\frac{1}{\sqrt{f}} = 1.74 - 2 \log_{10} \left(2 \ell_d + \frac{18.7}{N_{Re} \sqrt{f}} \right)$
 6. Calculate P_2 from the equation $q_s = 77.54 \left(\frac{T_s}{P_s} \right) d^{2.5} \sqrt{\frac{P_1^2 - P_2^2}{L T Z \lambda f}}$
 7. If calculated downstream pressure is equal to the assumed pressure, ($P_{2as} = P_2$) go to step 10, else
 8. Increase the assumed gas downstream pressure by a small value, $P_{2as} = P_{2as} + \delta$
 9. Go to step 2
 10. Record the calculated gas flow rate
 11. stop

4. RESULTS

4.1 Structure of the software; The Visual Basic programs written for the design and simulation of natural gas pipeline were linked to form the software GASPSIM. It is a user friendly software consisting of 7 windows (forms) displayed in

Graphic User Interface GUI. The main menu which is the start-up window enables the user to select a model, or to use unit conversion calculator. This software requires detailed characterization of the natural gas. Figure 1 through Figure 6 shows some of the software's windows.

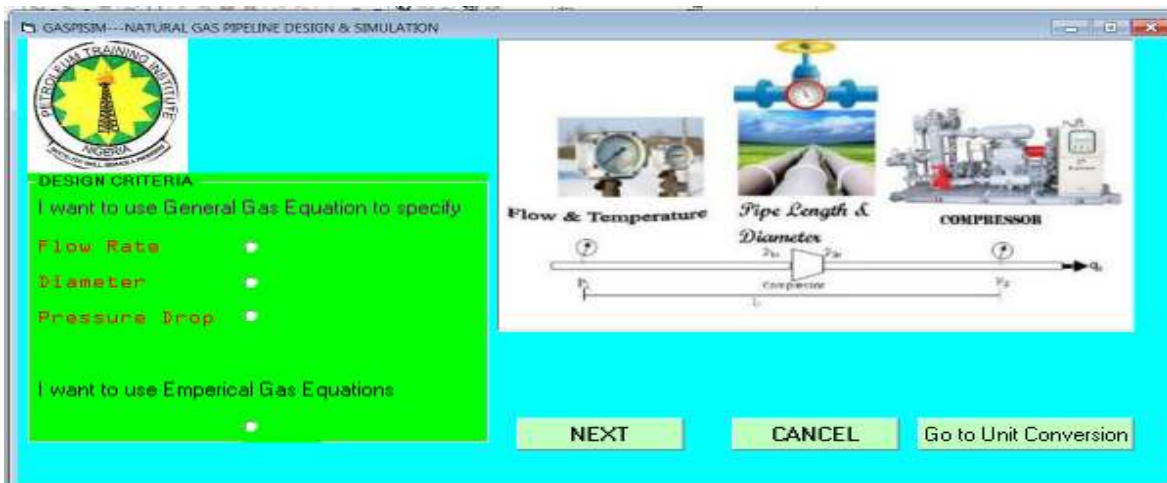


Figure 1: The main menu of the software (Start-up window)

4.2 Entering input data

The user can enter the data in the text boxes. The software has an error control capability in

which error messages are displayed, for example for; “no entry”, or “Incomplete entry”, the error message is as shown in Figure 2.

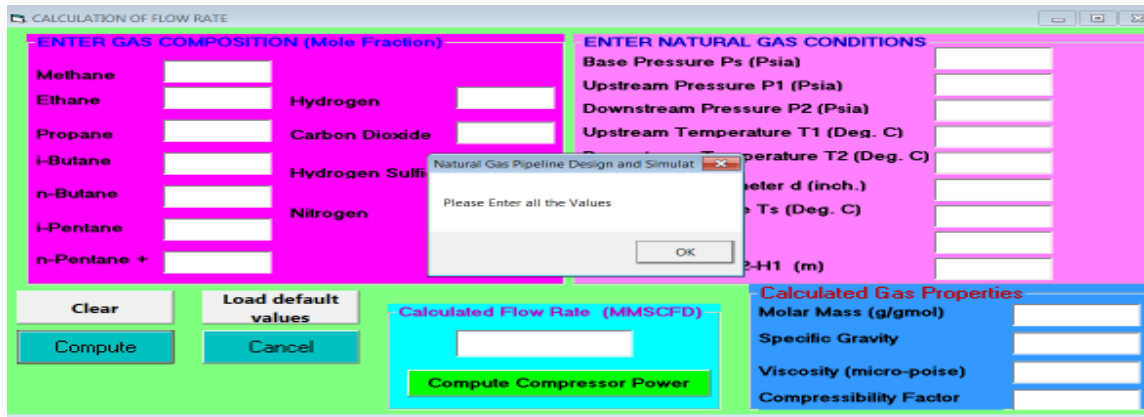


Figure 2: Window showing error message (no data entry)

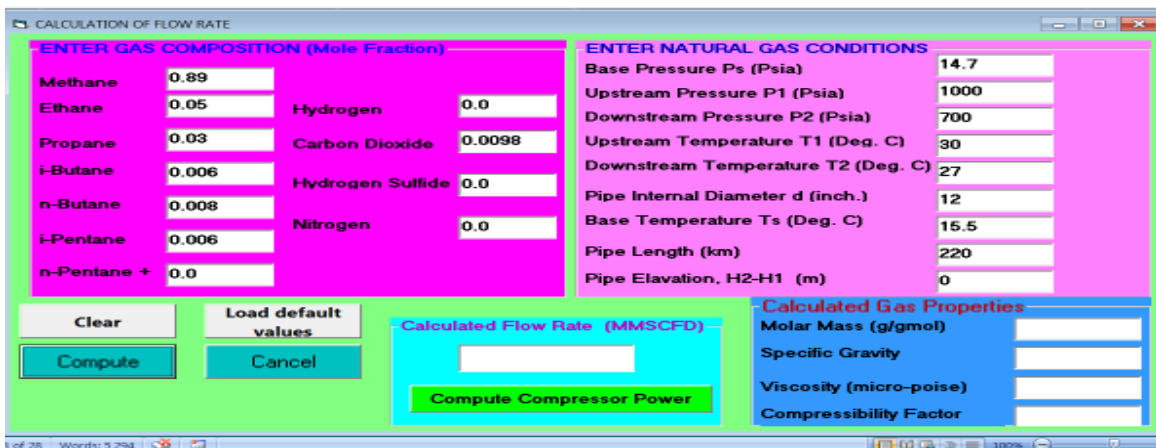


Figure 3: Window showing input data.

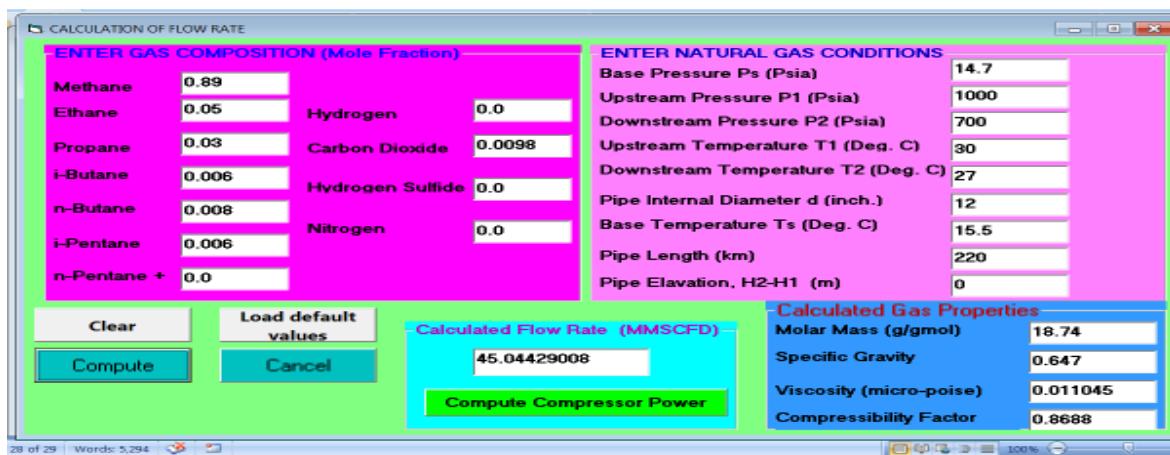


Figure 4: Window showing calculated flow rate and other gas properties.

Figure 5: Graphic user interface for compression power requirement.

Figure 6: Graphic user interface for unit conversion.

4.3 Testing the software; The last steps in developing software is debugging and testing. The statement that “If a program runs and prints out results then it is correct” is a misconception. There is a danger that the results appear to be “reasonable” but in fact wrong (Adamu, 2009). To

avoid the occurrence of such problem, the program must be subjected to a series of tests for which the correct answer is known before hand. The software, GASPISIM was tested using solved design problems from literature (Guo & Ghalambor, 2005).

Test Problem: Using general gas equation, calculate the flow rate of gas in a pipeline system having the following characteristics;

Component		Mole %
Methane	C ₁	89
Ethane	C ₂	5
Propane	C ₃	3
i-Butane	i-C ₄	0.6
n-butane	n-C ₄	0.8
i-Pentane	i-C ₅	0.6
n-Pentane	n-C ₅	0
Hydrogen	H ₂	0
Carbon dioxide	CO ₂	0.98
Hydrogen Sulfide	H ₂ S	0
Nitrogen	N ₂	0

D = 12.09 in
 L = 200 miles
 T = 80 °F = 540 °R
 T_s = 60 °F = 520 °R
 P_s = 14.7 Psia
 P₁ = 600 Psia
 P₂ = 200 Psia

Table 1: Comparison of results between Literature and software values.

Parameter	Units	SOLUTION		% diff. (Software-Literature)*100/Software
		Literature	Software	
Molar Mass	g/mol	20.71	21.33	2.90
Gas Specific Gravity		0.71	0.74	4.05
Gas Viscosity	cP	0.0099	0.0109	9.17
Compressibility Factor, Z		0.9188	0.9298	1.18
Gas Flow Rate	MMSCFD	28.511088	27.386490	-4.10

5. DISCUSSIONS

Comparison of the results obtained shows that both the software and literature values are close. The slight difference may be due to manual iteration and round up errors in the literature solution. It is expected that the software will give more accurate results since the round up errors are minimal.

CONCLUSION

GASPISIM is a useful tool in preliminary design and simulation of gas pipeline. GASPISIM is user friendly with ease of changing design parameters to accommodate changes in design conditions. Comparison of the software result and that of literature shows that the software can be used for design and simulation purposes.

REFERENCES

- Adamu, A. A., 2009. "Visual Basic Simulator VBCMAS for completely mixed activated sludge CMAS process, Part-1: BOD and Temperature Model", *Global Journal of Engineering Research*, 8, (1&2): 11.
- "Gas pipeline hydraulics"; Available online at: <http://www.pdengineer.com/>[Accessed 23rd Febuary, 2011]
- Genić, S., 2011. "A Review of Explicit Approximations of Colebrook's Equation", *FME Transactions* 39, 67-71
- 67, Innovation Center of Faculty of Mechanical Engineering, Kraljice Marije 16, 11120 Belgrade 35, Serbia.
- Guo, B and Ghalambor, A., 2005. *Natural gas engineering handbook*, Gulf Publishing Company 2 Greenway Plaza, Suite 1020 Houston, TX 77046.
- Coelho, P. M and Pinho, C., 2007. "Considerations About Equations for Steady State Flow in Natural Gas Pipelines", *Journal of the Brazilian Society of Mechanical Science & Engineering*, Vol. XXIX, No. 3, p 273.
- Perry, R. H and Green, D. W., 1998. *Perry's Chemical Engineer's Handbook*, 7th ed. McGraw-Hill International Edition Chemical Engineering Series, Australia.
- Reid, R. C., Prausnitz, J. M and Poling, B. E., 1988. *The Properties of Gases and Liquids*, 4th ed. McGraw-Hill International Edition Chemical Engineering Series, Singapore.
- Serghides, T. K., 1984. "Estimate friction factor accurately", *Chemical Engineering* 91(5) pp 63-4.
- Smith, J. M and Van Ness, H. C., 1987. *Introduction to Chemical Engineering thermodynamics*, 4th ed. McGraw-Hill Book Company, New York.

APPENDIX

```

Private Sub Command1_Click()
'GAS PIPELINE DESIGN USING VISUAL BASIC 6.0
'Program for calculation of Natural Gas Flow Rate using general gas equation
'Program written by: Engr. A. A. Adamu
'Petroleum & Natural Gas Processing Department, P.T.I. Effurun.
'Date: March 2012
If Text1.Text = "" Or Text2.Text = "" Or Text3.Text = "" Or Text4.Text = "" Or Text5.Text = "" Then
MsgBox ("Please Enter all the Values")
Exit Sub
ElseIf Text6.Text = "" Or Text7.Text = "" Or Text8.Text = "" Or Text9.Text = "" Or Text10.Text = "" Then
MsgBox ("Please Enter all the Values")
Exit Sub
ElseIf Text11.Text = "" Or Text12.Text = "" Or Text13.Text = "" Or Text14.Text = "" Or Text15.Text = "" Then
MsgBox ("Please Enter all the Values")
Exit Sub
ElseIf Text16.Text = "" Or Text17.Text = "" Or Text18.Text = "" Or Text19.Text = "" Or Text25.Text = "" Then
MsgBox ("Please Enter all the Values")
Exit Sub
ElseIf Text1.Text < 0 Or Text2.Text < 0 Or Text3.Text < 0 Or Text4.Text < 0 Or Text5.Text < 0 Then
MsgBox ("Please Enter only positive Values")
Exit Sub
ElseIf Text6.Text < 0 Or Text7.Text < 0 Or Text8.Text < 0 Or Text9.Text < 0 Or Text10.Text < 0 Then
MsgBox ("Please Enter only positive Values")
Exit Sub
ElseIf Text11.Text < 0 Or Text12.Text < 0 Or Text13.Text < 0 Or Text14.Text < 0 Or Text15.Text < 0 Then
MsgBox ("Please Enter only positive Values")
Exit Sub
ElseIf Text16.Text < 0 Or Text17.Text < 0 Or Text18.Text < 0 Or Text19.Text < 0 Or Text25.Text < 0 Then
MsgBox ("Please Enter only positive Values")
Exit Sub
End If
t1e = CDbI(Text15.Text): t2e = CDbI(Text16.Text)
tse = CDbI(Text18.Text)
lle = CDbI(Text19.Text): l = lle / 1.609
Dim tav, zav, spgr As Double
'Dim ts, ps, d, p1, p2, l, tav, zav, spgr, t1, t2 As Double
Dim pav As Double
'Dim y1, y2, y3, y4, y5, y6, y7, y8, y9, y10, y11 As Double
Dim x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11 As Double
'T in deg R, P in psia, L in miles,d in inches.
'Components: 1-methane, 2-ethane, 3-propane, 4-isobutane, 5-n-butane
'6-isopentane, 7-normalpentane, 8-Hydrogen, 9-carbondioxide,10-Hydrogen sulfide,'11-Nitrogen
ps = CDbI(Text12.Text):p1=CDbI(Text13.Text): p2 = CDbI(Text14.Text)
d = CDbI(Text17.Text):Y1 = CDbI(Text1.Text):Y2 = CDbI(Text2.Text)
y3 = CDbI(Text3.Text):y4 = CDbI(Text4.Text):y5 = CDbI(Text5.Text)
y6 = CDbI(Text6.Text):y7 = CDbI(Text7.Text):y8 = CDbI(Text8.Text)
y9 = CDbI(Text9.Text):y10= CDbI(Text10.Text):y11 = CDbI(Text11.Text)
Dim tav1 As Double
'T in Rankine
t1 = ((t1e + 273.15) * 9) / 5: t2 = ((t2e + 273.15) * 9) / 5
ts = ((tse + 273.15) * 9) / 5: tav1 = 0: tav1 = (t1 + t2) / 2
'Convert T average in Rankine to Kelvin
tav = (5 / 9) * (tav1 - 492) + 273.15
Dim pc1,pc2,pc3,pc4,pc5,pc6,pc7,pc8,pc9,pc10,pc11 As Double
Dim tc1,tc2,tc3,tc4,tc5,tc6,tc7,tc8,tc9,tc10,tc11 As Double
pc1=46#:pc2=48.8:pc3=42.5:pc4=36.5: pc5= 38#:pc6=33.9:
pc7=33.7:pc8=13#:pc9=73.8: pc10 = 89.4: pc11 = 33.9:tc1 = 190.4:

```

```

tc2 = 305.4: tc3 = 369.8: tc4 = 408.2: tc5 = 425.2: tc6 = 460.4:
tc7 = 469.7: tc8 = 32.2: tc9 = 304.1: tc10 = 373.2: tc11 = 126.2:
Dim pav1, kon1, kon2 As Double
pav1 = 0: kon1 = 0: kon2 = 0
kon1 = p1 * p2 / (1 * p1 + 1 * p2): kon2 = 1 * p1 + 1 * p2
pav1 = (2 * (kon2 - kon1)) / 3
pav = pav1 * (101325 / (14.7 * 100000#))
Dim pcm, tcm, z, h, pr, fx As Double
pcm = 0
pcm = Y1 * pc1 + Y2 * pc2 + y3 * pc3 + y4 * pc4 + y5 * pc5 + y6 * pc6 + y7 * pc7 + y8 * pc8 + y9 * pc9 + y10 * pc10
+ y11 * pc11
pr = pav / pcm: tcm = 0
tcm = Y1 * tc1 + Y2 * tc2 + y3 * tc3 + y4 * tc4 + y5 * tc5 + y6 * tc6 + y7 * tc7 + y8 * tc8 + y9 * tc9 + y10 * tc10 + y11
* tc11
tr = tav / tcm
Dim mm, m1, m2, m3, m4, m5, m6, m7, m8, m9, m10, m11 As Double
mm = 0: m1 = 16: m2 = 30: m3 = 44: m4 = 58: m5 = 58: m6 = 72: m7 = 72: m8 = 2: m9 = 44: m10 = 34: m11 = 28
mm = Y1 * m1 + Y2 * m2 + y3 * m3 + y4 * m4 + y5 * m5 + y6 * m6 + y7 * m7 + y8 * m8 + y9 * m9 + y10 * m10 + y11
* m11
spgr = mm / 28.97
'Print "The molar mass of the Gas mixture in g/gmol is = "; mm
Text21.Text = Format(mm, "##.00")
'Print "The Specific Gravity of the Gas mixture (air = 1) is = "; Format(mm / 28.97, "0.#####")
Text22.Text = Format(spgr, "0.000")
Dim zeta, zeta1, zp As Double
zp = 1 / 6: zeta1 = 0
zeta1 = tcm / ((mm ^ 3) * (pcm ^ 4))
zeta = 0: zeta = 0.176 * (zeta1 ^ zp)
Dim dpm, dp1, dp2, dp3, dp4, dp5, dp6, dp7, dp8, dp9, dp10, dp11 As Double
dpm = 0: dp1 = 0: dp2 = 0: dp3 = 0: dp4 = 0.1: dp5 = 0: dp6 = 0.1: dp7 = 0: dp8 = 0: dp9 = 0: dp10 = 0.9: dp11 = 0
dpm = Y1 * dp1 + Y2 * dp2 + y3 * dp3 + y4 * dp4 + y5 * dp5 + y6 * dp6 + y7 * dp7 + y8 * dp8 + y9 * dp9 + y10 *
dp10 + y11 * dp11
Dim dpr As Double
dpr = 0: dpr = (52.46 * dpm ^ 2 * pcm) / (tcm ^ 2)
Dim zcm, z1, z2, z3, z4, z5, z6, z7, z8, z9, z10, z11 As Double
zcm = 0: z1 = 0.288: z2 = 0.285: z3 = 0.281: z4 = 0.283: z5 = 0.274: z6=0.271:z7=0.263:z8 = 0.306:z9 = 0.274:z10
= 0.284: z11 = 0.29
zcm = Y1 * z1 + Y2 * z2 + y3 * z3 + y4 * z4 + y5 * z5 + y6 * z6 + y7 * z7 + y8 * z8 + y9 * z9 + y10 * z10 + y11 * z11
Dim fpo, fzo As Double
fpo = 0: fzo = 0
If dpr > 0 And dpr < 0.022 Then
fpo = 1
ElseIf dpr >= 0.022 And dpr < 0.075 Then
fpo = 1 + 30.55 * ((0.292 - zcm) ^ 1.72)
ElseIf dpr > 0.075 Then
fpo = 1 + 30.55 * ((0.292 - zcm) ^ 1.72) * Abs(0.96 + 0.1 * (tr - 0.7))
End If
fzo = 1
Dim z1f, z2f As Double
z1f = 0: z2f = 0
Dim kz1, kz2, kz3 As Double
kz1 = 0: kz2 = 0: kz3 = 0
kz1 = 0.807 * (tr ^ 0.618): kz2 = 0.357 * Exp((-0.449 * tr))
kz3 = 0.34 * Exp((-4.058 * tr)): z1f = (kz1 - kz2 + kz3 + 0.018) * fpo * fzo
Dim wm, w1, w2, w3, w4, w5, w6, w7, w8, w9, w10, w11 As Double
wm = 0: w1 = 0.011: w2 = 0.099: w3 = 0.153: w4 = 0.183: w5 = 0.199: w6=0.227:w7=0.251: w8 = 0.081: w9 =
0.239: w10 = -0.218: w11 = 0.039
wm = Y1 * w1 + Y2 * w2 + y3 * w3 + y4 * w4 + y5 * w5 + y6 * w6 + y7 * w7 + y8 * w8 + y9 * w9 + y10 * w10 + y11 *
w11
Dim fo, f1, pvpr1, pvpr, pvp As Double
fo = 0: f1 = 0: pvpr = 0: pvpr1 = 0: pvp = 0

```

```

fo = 5.92714-(6.09648/tr)-(1.28862* Log(tr)) + (0.169347 * (tr ^ 6))
f1 = 15.2518 -(15.6875 /tr)-(13.4721 * Log(tr))+0.43577 * (tr ^ 6))
pvpr1 = fo + wm * f1: pvpr = Exp(pvpr1): pvp = pvpr * pcm
Dim pvptopcm As Double
pvptopcm = pvp / pcm
If tr <= 1# And pr < pvptopcm Then
Dim alpha, beta As Double
alpha = 0#: beta = 0#
alpha = 3.262 + 14.98 * (pr ^ 5.508): beta = 1.39 + 5.746 * pr
z2f=0.6+ 0.76 *(pr ^ alpha) +(6.99 *(pr ^ beta) - 0.6) * (1 - tr)
Elseif tr > 1 And (pr > 0 And pr <= 100) Then
Dim a1, a2, gama As Double
a1 = 0.001245: a2 = 5.1728: gama = -0.3286
Dim b1, b2 As Double
b1 = 1.6553: b2 = 1.2723:
Dim c1, c2, delta As Double
c1 = 0.4489: c2 = 3.0578: delta = -37.7332
Dim d1, d2, epsilon, jj As Double
jj = 0#: d1 = 1.7368: d2 = 2.231: epsilon = -7.6351
Dim f1c, f2c, zai, tt As Double
tt = 0#: f1c = 0.9425: f2c = -0.1853: zai = 0.4489
Dim a, b, cf, df, f, uu As Double
uu = 0#: a = 0#: b = 0#: cf = 0#: df = 0#: f = 0#
a = (a1 / tr) * Exp(a2 * (tr ^ gama))
b = a * (b1 * tr - b2)
cf = (c1 / tr) * Exp(c2 * (tr ^ delta))
df = (d1 / tr) * Exp(d2 * (tr ^ epsilon))
f = f1c * Exp(f2c * (tr ^ zai))
Dim con1, con2, con3, vv As Double
vv = 0: con1 = 0#: con2 = 0#: con3 = 0#
con1 = b * (pr ^ f): con2 = (1 + cf * (pr ^ df)) ^ -1
con3 = a * (pr ^ 1.3088): z2f = z1f * (1 + (con3 / (con1 + con2)))
End If
Dim ybig, fp, fq, viscosity As Double
ybig = 0: fp = 0: fq = 0: ybig = z2f / z1f
fp = (1 + (fpo - 1) * (ybig ^ (-3))) / fpo
Dim fq1, fq2 As Double
fq1 = 0: fq2 = 0
fq1 = (ybig ^ -1) - 0.007 * (Log(ybig) ^ 4)
fq2 = 1 + (fqo - 1): fq = fq1 * fq2: viscosity = 0
'viscosity in micro poise
viscosity = (z2f * fp * fq) / zeta
h = 0: z = 0.1
Dim h1, fx1 As Double
h1 = (0.08664 * pr) / (z * tr)
fx1 = z - (1 / (1 - h1)) + ((4.934 * h2) / ((tr ^ 1.5) * (1 + h3)))
If fx1 < 0 Then
fx = -1.5
Do While fx < 0
h = (0.08664 * pr) / (z * tr)
fx = z - (1 / (1 - h)) + ((4.934 * h) / ((tr ^ 1.5) * (1 + h)))
z = z + 0.0001
Loop
Else
z = 0.1: fx = 1.5
Do While fx > 0
h = (0.08664 * pr) / (z * tr)
fx = z - (1 / (1 - h)) + ((4.934 * h) / ((tr ^ 1.5) * (1 + h)))
z = z + 0.0001
Loop
End If

```

```

Text24.Text = Format (z, "0.0000")
Dim aw1, qwey As Double
qwey = 0
'Gas flow rate in standard cfd
aw1 = (p1 ^ 2 - p2 ^ 2) / (l * tav1 * z * spgr)
qwey = 433.8 * (d ^ (8 / 3)) * (ts / ps) * Sqr(aw1)
Dim qas, qcal, qs As Double
'Gas flow rate in standard CFD
qas = qwey: qcal = 0: qs = 0
Dim qdf As Double
qdf = 10
Do While qdf > 0
Dim nre, fr As Double
nre = 0: fr = 0
Dim viscosity1, viscosity2 As Double
'viscosity in centipoise
viscosity1 = 0: viscosity2 = 0
viscosity2 = 0.0001 * viscosity
'viscosity in lbf/ft.s
viscosity1 = viscosity2 * 0.000672
nre = ((0.0004778 * qas * spgr) / (viscosity1 * d)) * (ps / ts)
If nre <= 2100 Then
fr = 64 / nre
Else
Dim epsilond, epsilonfr As Double
epsilond = 0: epsilonfr = 0.0006
epsilond = epsilonfr / d
Dim fr1 As Double
fr1 = 0
fr1 = 1.14 - (2 * Log(epsilond + (21.25 / (nre ^ 0.9)))) / Log(10)
fr = (1 / fr1) ^ 2
End If
Dim elev, elevation, sel, lel As Double
Dim eft As Double
elev = CDbl(Text25.Text)
If elev <> 0 Then
Elevation = elev * 3.28
sel = 0.0375 * spgr * (elevation / (tav1 * z))
lel = l * (Exp(sel) - 1) / sel
eft = Exp(sel)
Else
lel = l: eft = 1
End If
Dim c1f, c2f, c3f As Double
c1f = (p1 ^ 2 - (eft * p2 ^ 2)) * (d ^ 5)
c2f = lel * tav1 * z * spgr * fr
c3f = c1f / c2f
'qs in SCFD
qcal = 77.54 * (ts / ps) * Sqr(c3f)
qdf = qcal - qas: qas = qcal
Loop
'qs in MMSCFD
Text20.Text = Format ((qcal / 1000000), "#.00000000")
Text23.Text = Format (viscosity2, "0.000000")
End Sub

```