The Gaussian dispersion model serves as an excellent approximation to point source emission scenarios, such as coal plants with smokestacks. The simulation model in this exercises attempts to address the issue of sulfur removal in order to satisfy permitted ground level concentrations.
Introduction

The Gaussian dispersion (plume) model is one of five mathematical models used to describe the nature of the dispersion of air pollutants in the atmosphere using several parameters\(^1\). It is so named because it assumes a normal probability distribution of the pollutant in the plumes. The individual parameters used to define the model are:

a) Pollutant emission rate  
b) Height of the smokestack  
c) The speed at which the gases exit the stack  
d) The diameter of the smokestack  
e) The temperature of the gas within the stack  
f) The temperature of the air surrounding the stack  
g) The speed of the wind surrounding the stack  
h) The atmospheric pressure surrounding the stack  
i) The specific atmospheric stability class.

The Pasquill atmospheric stability class ranges from A to F and defines the amount of turbulence within the atmosphere. Turbulence is due to changes in wind speed that may be the result of thermal effects (change of temperature) and surface characteristics. As a result, the stability classes are defined differently for different times of the day, as well as for different cloud cover at night.

In addition to the normal distribution describing the diffusion of the plume, there are several other key assumptions that form part of the Gaussian dispersion model:

1) The air pollutant is released into the atmosphere at a certain height \(H\), comprising of both the smokestack height \(h\) and the plume rise \(\Delta H\).  
2) The degree of dilution of the plume is inversely proportional to the wind speed.  
3) Any pollutant that reaches ground level is totally reflected back into the atmosphere.

Defining the Model

The model equation was first proposed by D. Bruce Turner in *Workbook of Atmospheric Dispersion Estimates* in 1967. The **ground level** concentration equation is shown below:

\(^{1}\) The others are the Box, Lagrangian, Eulerian, and Dense gas models.
\( \chi(x, y, 0, H) = \left[ \frac{Q}{\pi s_y s_z u} \right] e^{-0.5 \left( \frac{y}{s_y} \right)^2} e^{-0.5 \left( \frac{u}{s_z} \right)^2} \) \hspace{1cm} \text{Equation 1}

- \( Q \) = emission rate of pollutant in g/s
- \( s_y, s_z \) = plume standard deviations, m
- \( u \) = wind speed in m/s
- \( H = h + \Delta H \), \( h \) = physical stack height, \( \Delta H \) = plume rise

\( s_y, s_z \) are dependent on turbulence and atmospheric stability. A graphical representation is shown in the appendix, but they can also be defined using:

\[
\begin{align*}
  s_y &= a x^{0.894} \quad \text{Equation 2} \\
  s_z &= c x^d + f \quad \text{Equation 3}
\end{align*}
\]

The variables \( a, b, c, \) and \( d \) are defined in a separate table, and combined help define the standard deviations for any distance \( x \) from the stack.

The plume rise is derived from yet another equation:

\[
\Delta H = \frac{v_s d}{u} \left[ 1.5 + 2.68 \times 10^{-2} (P) \left( \frac{T_s - T_a}{T_s} \right) d \right] \quad \text{Equation 4}
\]

- \( v_s \) = stack velocity, m/s
- \( d \) = stack diameter, m
- \( u \) = wind speed in m/s
- \( P \) = pressure, kPa
- \( T_s \) = stack temperature, K
- \( T_a \) = ambient temperature, K
Procedure

For this exercise, Matlab software was used to model centerline and ground level concentration of sulfur dioxide emitted from a coal-fired power station using hypothetical values for smokestack height, wind speed, emitted gas and ambient temperature, as well as other variables outlined previously. The emission rate of sulfur can be calculated using the amount of power supplied, the efficiency of the plant, as well as the sulfur concentration per weight.

\[
\text{Power supplied x efficiency} = 500 \text{MW} = 500 \times 10^6 \frac{J_{\text{electric}}}{s} \left( \frac{J_{\text{thermal}}}{0.33 J_{\text{electric}}} \right) = 1515 \times 10^6 \frac{J_{\text{thermal}}}{s}
\]

\[
\text{Coal consumed given energy used} = 1515 \times 10^6 \frac{J_{\text{thermal}}}{s} \left( \frac{1 \text{g-coal}}{28,000 J_{\text{thermal}}} \right) = 54,113 \frac{\text{g-coal}}{s}
\]

\[
\text{Sulfur content in coal consumed} = 54,113 \frac{\text{g-coal}}{s} \left( \frac{3 \text{g-sulfur}}{100 \text{g-coal}} \right) = 1623.4 \frac{\text{g-sulfur}}{s}
\]

\[
\text{Sulfur dioxide emission using available sulfur} = 1623.4 \frac{\text{g-sulfur}}{s} \left( \frac{64 \text{g-SO}_2}{32 \text{g-sulfur}} \right) = 3247 \frac{\text{g-SO}_2}{s}
\]
Results

Wind speed and atmospheric stability conditions were varied in order to determine conditions in which the maximum permissible sulfur dioxide concentration at ground level would be violated. For the purposes of this experiment, that upper threshold was set at 365 μg/m³. A graphical user interface (GUI) was made using Matlab in order to facilitate the incremental changes to individual variable values. With a stack height of 20m, wind speed of 5m/s, a stack diameter of 6m and using the ‘A’ atmospheric stability class, the permissible concentration is clearly violated. Reducing the wind speed only allowed for concentrations to stay within the allowable limit for a certain distance, and this was only more effective with less turbulent atmospheric stability classes.
**Table 1.** Results from varying individual parameters (shaded in gray).

<table>
<thead>
<tr>
<th>Stack height (m)</th>
<th>Wind speed (m/s)</th>
<th>Stack diameter (m)</th>
<th>Outdoor temperature (°C)</th>
<th>Stability Class</th>
<th>Emission rate (g/s)</th>
<th>Violates limit? (and notes)</th>
<th>Plume rise (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>5</td>
<td>6</td>
<td>15.3</td>
<td>A</td>
<td>3247</td>
<td>Yes. With the E stability class, however, it is not violated up to nearly 10km.</td>
<td>167</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>6</td>
<td>15.3</td>
<td>A</td>
<td>3247</td>
<td>Yes. With the D (and beyond) stability class, however, no.</td>
<td>833</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>8.2</td>
<td>15.3</td>
<td>B</td>
<td>3247</td>
<td>No.</td>
<td>1466</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>8.2</td>
<td>15.3</td>
<td>B</td>
<td>3247</td>
<td>Yes. Higher wind speeds lead to violation.</td>
<td>293</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>8.2</td>
<td>18.5</td>
<td>B</td>
<td>3247</td>
<td>Yes. Higher temperature leads to violation.</td>
<td>1441</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.4</td>
<td>1</td>
<td>8.2</td>
<td>18.5</td>
<td>B</td>
<td>3247</td>
<td>Yes. Lower stack heights lead to violation.</td>
<td>1441</td>
</tr>
</tbody>
</table>

**Table 2.** Changing the smokestack height to satisfy standards. As compared to 5th row in Table 1.

<table>
<thead>
<tr>
<th>Stack height (m)</th>
<th>Wind speed (m/s)</th>
<th>Stack diameter (m)</th>
<th>Outdoor temperature (°C)</th>
<th>Stability Class</th>
<th>Emission rate (g/s)</th>
<th>Violates limit? (and notes)</th>
<th>Plume rise (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>53.1</td>
<td>1</td>
<td>8.2</td>
<td>18.5</td>
<td>B</td>
<td>3247</td>
<td>No.</td>
<td>1441</td>
</tr>
</tbody>
</table>
Table 3. Changing emission rate to satisfy standards. As compared to 4th row in Table 1.

<table>
<thead>
<tr>
<th>Stack height (m)</th>
<th>Wind speed (m/s)</th>
<th>Stack diameter (m)</th>
<th>Outdoor temperature (°C)</th>
<th>Stability Class</th>
<th>Emission rate (g/s)</th>
<th>Violates limit? (and notes)</th>
<th>Plume rise (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>5</td>
<td>8.2</td>
<td>15.3</td>
<td>B</td>
<td>945</td>
<td>No. Lowering emission rate allows standard to no longer be violated.</td>
<td>293</td>
</tr>
</tbody>
</table>

**Discussion**

The amount of sulfur dioxide emission reduction necessary to satisfy standards using a stack height of 20m, windspeed of 5m/s, a stack diameter of 8.2m and an outdoor temperature of 15.3°C with stability condition B is

\[
\frac{945}{3247} = 0.291 \text{ or } 1 - 0.291 \text{ or approximately } 70.9\% \text{ reduction in sulfur content.}
\]

For the purposes of comparing initial parameters (with a stack diameter of 6m instead), the emission rate needs to be reduced to 364g/s or a **89.3% reduction in sulfur content.**

In order to address the issue of reducing the amount of sulfur to the emission rate shown in the results so as to remain within acceptable concentration limits using the dispersion model, several methods can be adopted. The most typical and widely used involves **flue gas desulfurization (FGD),** a method that converts harmful sulfur dioxide into harmless byproducts using chemical reactions. Using limestone or lime, the following chemical equation represents the reaction:

Limestone: \( \text{SO}_2 + \text{CaCO}_3 \rightarrow \text{CaSO}_3 + \text{CO}_2 \)

Lime: \( \text{SO}_2 + \text{Ca(OH)}_2 \rightarrow \text{CaSO}_3 + \text{H}_2\text{O} \)

Typical efficiencies using limestone are approximately 90%, which would satisfy the pollutant reduction amount necessary as outlined above. An unfortunate side-effect of using lime, despite being more efficient than limestone, is that up to 200kg of lime is necessary per ton of sulfur dioxide emitted in a ton of coal, resulting in significant amounts of waste as well as higher cost. The above methods are non-regenerative (or throwaway) – that is, the lime or the limestone used must be discarded after processing. A regenerative process includes using **magnesium oxide** slurry. Additionally, these systems are **wet** systems in that a liquid is used, but given the general messiness involved, the heat transfer
necessary for proper reaction, and issues of corrosion and scaling, dry systems are a favorable alternative. Wet systems are significantly more efficient and are adopted in nearly 85% of existing FGD systems.

As sulfate (gypsum) has considerable uses more practical than sulfite, an additional oxidation step is commonly adopted:

\[ \text{CaSO}_3 + \frac{1}{2} \text{O}_2 \rightarrow \text{CaSO}_4 \]

**Figure 2.** A commercially available dry flue gas desulfurization system as presented by Babcock & Wilcox. Image courtesy of Babcock & Wilcox.
References


‘Environmental Chemistry’. Stanley E. Manahan


Appendix

Figure 3. Gaussian dispersion model superimposed on a real photograph.
**Figure 4.** $s_y$, horizontal dispersion coefficient, for varying distances downwind and different atmospheric stability conditions.

**Figure 5.** $s_z$, vertical dispersion coefficient, for varying distances downwind and different atmospheric stability conditions.
Figure 6. Different plume types depending on different atmospheric stability conditions.

Matlab Code (Excerpt)

The following code forms part of the main component of the MatLab GUI developed for this exercise. Sections have been removed for clarity. The entire code is available for download at http://wiki.rioleo.org

```matlab
% --- Does all the plotting. This is a function that is part of a larger code.
function doall(a, b, c, d, e, beta, f, h)
load params.dat % read in file
Qstack = h; % Get pollutant emission rate from slider in GUI
Hstack = a; % Get height of stack from slider in GUI
Vstack = params(3);
Dstack = c; % Get stack diameter from GUI
Tstack = params(5);
Tamb = f; % Get ambient temperature from GUI
Uwind = b; % Get wind speed from GUI
```
Patmos = params(8);
Satmos = params(9);
Tstack = Tstack+273; % Convert temperatures to Kelvin
Tamb = Tamb + 273;
y = 0;
deltaH = (Vstack*Dstack/Uwind)*(1.5 + (2.68*10^-2*Patmos*((Tstack - Tamb) / Tstack)*Dstack));
Xplot = []; % array time will store all distance values
Cplot1 = []; % array Cplot will hold centerline concentrations
Gplot1 = []; % array of ground level concentrations
xmin = 0.1; % set smallest distance x to evaluate (in km)
xmax = 10; % set largest distance x to evaluate (in km)
xinc = 0.01; % set increment in x
if get(e, 'Value')==1
    a = 213;
    x = xmin;
    while x <= xmax
        if x <= 1
            c = 440.8;
            d = 1.941;
            f = 9.27;
        else
            c = 459.7;
            d = 2.094;
            f = -9.6;
        end
        Sy = a*(x)^0.894;
        Xplot = [Xplot, x];
        Sz = f + c*(x)^d;
        Center_Conc = 1e6*Qstack/(2*pi*Sy*Sz*Uwind);
        Cplot1 = [Cplot1,Center_Conc];
        x = x + xinc;
        chi = 1e6*Qstack / (pi*Sy*Sz*Uwind)* exp(-0.5*(y/Sy)^2) * exp(-0.5*((Hstack + deltaH)/Sz)^2);
        Gplot1 = [Gplot1,chi];
    end
elseif get(e,'Value')==2
    a = 156;
    x = xmin;
    while x <= xmax
        if x <= 1
            c = 100.6;
            d = 1.149;
            f = 3.3;
        else
            c = 108.2;
            d = 1.098;
            f = 2;
        end
        Sy = a*(x)^0.894;
        Xplot = [Xplot, x];
        Sz = f + c*(x)^d;
        Center_Conc = 1e6*Qstack/(2*pi*Sy*Sz*Uwind);
        Cplot1 = [Cplot1,Center_Conc];
        x = x + xinc;
chi = 1e6*Qstack / (pi*Sy*Sz*Uwind)* exp(-0.5*(y/Sy)^2) * exp(-0.5*((Hstack + deltaH)/Sz)^2);
  Gplot1 = [Gplot1,chi];
  end
elseif get(e,'Value')==3
  ...
elseif get(e,'Value')==4
  ...
elseif get(e,'Value')==5
  ...
elseif get(e,'Value')==6
  a = 34;
  x = xmin;
  while x <= xmax
    if x <= 1
      c = 14.35;
      d = 0.74;
      f = -0.35;
    else
      c = 62.6;
      d = 0.18;
      f = -48.6;
    end
    Sy = a*(x)^0.894;
    Xplot = [Xplot, x];
    Sz = f + c*(x)^d;
    Center_Conc = 1e6*Qstack/(2*pi*Sy*Sz*Uwind);
    Cplot1 = [Cplot1,Center_Conc];
    x = x + xinc;
    chi = 1e6*Qstack / (pi*Sy*Sz*Uwind)* exp(-0.5*(y/Sy)^2) * exp(-0.5*((Hstack + deltaH)/Sz)^2);
    Gplot1 = [Gplot1,chi];
  end
end
plot(Xplot,Gplot1,'.')
xlabel('Distance in km');
ylabel('Concentration in \mug / m^3');
hline(beta, 'r', 'Limit');