ANNEXURE - A

A.1 Algorithm for Runge Kutta 4\textsuperscript{th} order Method:

To approximate the solution of the initial value problem

\[ y' = f(t, y) \]

Let \( a \leq t \leq b \) \( y(a) = \alpha \) \( y(a) = \beta \)

N+1 equally spaced numbers in the interval \([a,b]\)

Input \(\rightarrow\) End points \(a, b; \) Integer \(N; \) Initial conditions of \(\alpha, \beta\)

Output \(\rightarrow\) Approximation \(w, v\) to \(y\) & \(y'\) at the \((N+1)\) values of \(t\)

Step 1: Set \(h = \frac{(b-a)}{N}; t = a; w = \alpha; \) \(v = \beta\)

Step 2: For \(i=1,2,3, \ldots \) \(N\) Do steps 3 to 5

Step 3: Set \(k_1 = h \times f(t, w, v)\)

\[ L_1 = h \times v \]

\[ K_2 = h \times f\left(t + \frac{h}{2}, w + \frac{L_1}{2}, v + \frac{K_1}{2}\right) \]

\[ L_2 = h \times \left(v + \frac{K_1}{2}\right) \]

\[ K_3 = h \times f\left(t + \frac{h}{2}, w + \frac{L_2}{2}, v + \frac{K_2}{2}\right) \]

\[ L_3 = h \times \left(v + \frac{K_2}{2}\right) \]

\[ K_4 = h \times f\left(t + h, w + L_3, v + K_3\right) \]

\[ L_4 = h \times \left(v + K_3\right) \]
Step 4: Set \( w = w + \frac{(L_1 + 2L_2 + 2L_3 + L_4)}{6} \) and compute \( W_i \)

Set \( v = v + \frac{(K_1 + 2K_2 + 2K_3 + K_4)}{6} \) and compute \( V_i \)

Step 5: Output \( (t, w, v) \)

Step 6: Stop

A.2 MONTE–CARLO TECHNIQUE:

Monte Carlo methods are a widely used class of computational algorithms for simulating the behavior or various physical and mathematical systems, and for other computations. They are distinguished from other simulation methods (such as molecular dynamics) by being stochastic, that is nondeterministic in some manner usually by using random numbers (in practice, pseudo-random numbers) as opposed to deterministic algorithms. Because of the repetition of algorithms and the large number of calculations involved, Monte Carlo is a method suited to calculation using a computer, using many computer simulation techniques.

Monte Carlo algorithm is often a numerical Monte Carlo method used to find solutions to mathematical problems (which may have many variables) that cannot easily be solved, for example, by integral calculus, or other numerical methods. For many types of problems, its efficiency relative to other numerical methods increases as the dimension of the problem increases. Or it may be a method for solving other mathematical problems that rely on (pseudo-)random numbers.
Monte-Carlo method is a numerical method of solving mathematical problems by means sampling. This method allows simulating any process whose development is affected by random factors. Many mathematical problems, which are not affected by any random influences, can be connected with an artificially constructed probabilistic model (even more than one) making possible the solution of these problems. Random motion of a particle in GIS bus can be calculated by applying Monte-Carlo method.

Two distinct features of the Monte-Carlo method:

- The first feature is the simple structure of the computation algorithm. As a rule, a program is prepared to perform only one random trial. This trial is repeated \( n \) times, each trial is independent of all Monte-Carlo method is sometimes called the method of statistical trails.

- The Monte-Carlo method is especially efficient in solving those problems, which do not require a high degree of accuracy (i.e. 5 to 10%)

The Simulation of the motion equation given in chapter III and Chapter IV yields particle movement in the radial direction only. However, the configuration at the tip of the particle is generally not sufficiently smooth enough to enable the movement unidirectional. This decides the movement of particle in axial direction. The randomness of movement can be adequately simulated by Monte-Carlo method. In order
to determine the randomness, it is assumed that the particle emanates from its original site at any angle less than $\phi$, where $\phi/2$ is half of the solid angle subtended with the vertical axis. At every step of movement, a new rectangular random number is generated between 0 and 1 and modified to $\phi$. The angle thus assigned, fixes the position of particle at the end of every time step, and in turn determines the axial and radial positions. The position in the next step is computed on the basis of equation of motion with new random angles as described above. The results of the simulations are given in chapter V.

**ALGORITHM FOR MONTE CARLO SIMULATION:**

Step 1 : Create a parametric model, $y = f(X_1, X_2, \ldots, X_q)$

Step 2 : Generate a set of random inputs, $X_{i1}, X_{i2}, \ldots, X_{iq}$.

Step 3 : Evaluate the model and store the results as $y_i$.

Step 4 : Repeat steps 2 and 3 for $i=1$ to $n$

Step 5 : Analyze the results
B.1 Moeal Calculations for Particle movement in Single phase isolated conductor Gas Insulated Busduct:

Consider Aluminum Particle of Length 12 mm and 0.2 mm as radiation of the surface of enclosure in a gas insulated bus duct of 152mm enclosure diameter and 55mm inner conductor diameter.

Let applied voltage \( V = 100kV \) (rms).

Pressure of the Gas \( P = 0.4Mpa \)

Maximum Voltage \( V_m = 100 \times 1.414 \times 1000 = 0.1414 \times 10^6 \) volts.

Density of Aluminum particle \( \rho = 2700kg/m^3 \)

Mass of the particle \( m = \pi r^2 l \rho \)

\[ m = 3.14 \times (0.2 \times 10^{-3}) \times (8 \times 10^{-3}) \times (2700) \]

\[ = 4.06944 \times 10^{-6} \text{ kg} \]

Lift off Field \( E_{LO} = 0.84 \sqrt{\frac{\rho g r}{\varepsilon_0}} \)

\[ = 0.84 \sqrt{\frac{2700 \times 9.81 \times 0.2 \times 10^3}{8.854 \times 10^{-12}}} \]

\[ = 6.4974 \times 10^5 \text{V/m} \]

Initial Time to lift off \( t_i = \left\{ \sin^{-1} \left[ E_{LO} \cdot r_0 \ln \left( \frac{r_0}{r_i} \right) / V \right] \right\} \)

\[ = 0.06622 \text{ sec} \]

Angular velocity \( \omega = 2\pi f^* = 2 \times 3.14 \times 50 = 314 \text{ rad / sec} \)

Gas density \( \rho_g = 7.118 + 6.332P + 0.2032P^2 \)

\[ = 9.683312 \]
Motion equation by considering all forces can be expressed as

$$m \ddot{y}(t) = \left\{ \begin{align*}
(2\pi \varepsilon_0 r|E_{LO}|)^* & \left[ \frac{V \sin \omega t}{r_0 - y(t) \ln \left( \frac{r_0}{r_i} \right)} \right] - mg \\
- y(t) \pi r & \left[ 6 \mu \kappa y(t) + 2.656 \left( \mu \rho_\kappa \ell y(t) \right)^{0.5} \right]
\end{align*} \right\}$$

This equation is second order non-linear differential equation and solved by using iterative methods. Here Runge-Kutta Method solving the above motion equation.

By substituting the values of $E_{LO}$, $\omega$, $R$, $\varepsilon_0$, $m$, $g$, $r_0$, $r_i$, $l$, $\mu$ in the motion equation becomes

$$\ddot{y}(t) = \left[ \frac{19.42 \sin \omega t}{76 \times 10^{-3} - y(t)} - 9.81 - 0.0215 \dot{y}(t) - 1.086 \left( \frac{\dot{y}(t)}{y(t)} \right)^{0.5} \right]$$

Applying the following initial conditions for the solving the motion equation $m \dot{y}(t = 0^+) = -Rm \dot{y}(t = 0)$ and $y(t=0^+) = 0$

Where

$y(t)$ = displacement of the particle.

$\dot{y}(t)$ = velocity of the particle.

$\ddot{y}(t)$ = acceleration of the particle.

The starting time is found to be 0.06622 seconds and the step is taken as 0.001

As per Runge–Kutta method
\[ K_1 = \Delta t \cdot f(t_0, y(t), y'(t)) \]

\[ = 0.0001 \cdot (20.773 - 9.81 - 0.0) \]

\[ = 0.4037 \cdot 10^{-3} \]

\[ L_1 = \Delta t \cdot y(t) \]

\[ = 0.0001 \times 0 \]

\[ = 0 \]

\[ K_2 = \Delta t \cdot \left( f\left( t_0 + \frac{\Delta t}{2}, y(t) + \frac{\Delta t}{2}, y'(t) + \frac{K_1}{2} \right) \right) \]

\[ = 0.0001 \cdot \left( 13.86 - 9.81 - 4.341925 \cdot 10^{-6} - 2.0778 \cdot 10^{-6} \right) \]

\[ = 0.4047 \cdot 10^{-3} \]

\[ L_2 = \Delta t \cdot \left( y'(t) + \frac{K_1}{2} \right) \]

\[ = 0.0001 \cdot (0 + 0.020195 \cdot 10^{-3}) \]

\[ = 0.020195 \cdot 10^{-6} \]

\[ K_3 = \Delta t \cdot \left( f\left( t_0 + \frac{\Delta t}{2}, y(t) + \frac{L_2}{2}, y(t) + \frac{K_2}{2} \right) \right) \]

\[ = 0.4057 \cdot 10^{-3} \]

\[ L_3 = \Delta t \cdot \left( y(t) + \frac{K_2}{2} \right) \]

\[ = 0.020235 \cdot 10^{-6} \]

\[ K_4 = \Delta t \cdot f\left( t_0 + \frac{\Delta t}{2}, y(t) + \frac{L_3}{2}, y(t) + \frac{K_3}{2} \right) \]

\[ = 0.4067 \cdot 10^{-3} \]
\[ L_4 = \Delta t^* \left( y(t) + K_3 \right) \]

\[ = 0.020285 \times 10^{-6} \]

Now

\[ Y_1(t) = y(t) + \frac{1}{6} \left( L_1 + 2L_2 + 2L_3 + L_4 \right) \]

\[ = 0.028825 \times 10^{-6} \text{ m} \]

\[ Y_1(t) = y(t) + \frac{1}{6} \left( K_1 + 2K_2 + 2K_3 + K_4 \right) \]

\[ = 0.4052 \times 10^{-3} \text{ m/sec} \]

**B.2 Model Calculations for Particle movement in single phase coated conductor Gas Insulated Busduct:**

Consider Aluminum Particle of Length 8 mm and 0.2 mm as radius at the surface of enclosure in a gas insulated bus duct of 152 mm enclosure diameter and 55 mm inner conductor diameter coated with dielectric material of thickness 75um.

For a parallel plate capacitor \( R = \frac{\rho c t}{s} \)

For a horizontal wire particle, the contact area \( s = \beta \pi rl \)

Then \( R = \frac{\rho c t}{\beta \pi rl} \)

\[ = \frac{1 \times 10^8 \times 75 \times 10^{-6}}{0.001 \times 3.14 \times 0.2 \times 10^{-3} \times 12 \times 10^{-3}} \]

\[ = 0.9960 \times 10^{13} \]

The capacitance \( C_c \) between the enclosure surface and the particle is
$$C_C = \varepsilon_0 \varepsilon_r \frac{s}{t}$$

$$= \frac{8854 \times 10^{-12} \times 7.23 \times 0.001 \times 3.14 \times 2 \times 10^{-3} \times 12 \times 10^{-3}}{75 \times 10^{-6}}$$

$$= 5.148336 \times 10^{-14} \text{ Farad}$$

The Capacitance $C_g$, between the horizontal wire particle of length 1 particle and enclosure is

$$C_g = 2\pi \varepsilon_0 \frac{rl}{r_0 \ln \frac{r_0}{r_i}}$$

$$= \frac{2 \times 3.14 \times 8.854 \times 10^{-12} \times 2 \times 10^{-3} \times 12 \times 10^{-3}}{76 \times 10^{-3} \ln \left( \frac{76 \times 10^{-3}}{27.5 \times 10^{-3}} \right)}$$

$$= 4.63155 \times 10^{-16} \text{ Farad}$$

The charging current through the conductance is given by

$$I_C = \frac{V}{\left[ R^2 \left( 1 + \frac{C_C}{C_g} \right)^2 + \frac{1}{\omega^2 C_g^2} \right]^{0.5}}$$

$$= \frac{141.4 \times 1000}{\left[ \left( 0.9960 \times 10^{13} \right)^2 \left( 1 + \frac{5.148336 \times 10^{-14}}{4.63155 \times 10^{16}} \right)^2 + \frac{1}{\left( 314 \right)^2 \left( 4.63155 \times 10^{16} \right)} \right]^{0.5}}$$

$$= 1.26578 \times 10^{-10} \text{ Amps}$$

$$\phi(t) = \tan^{-1} \left[ -\frac{1}{\omega R \left( C_g + C_C \right)} \right]$$
\[
B(\phi) = \left[ \cos \phi - \cos \left( \frac{2\pi + 2\phi}{3} \right) \right] \sin \frac{2\pi - \phi}{3}
\]

\[
= (0.9999878 - 0.99944752)0.03817
\]

\[
= 0.02062 \times 10^{-3}
\]

\[
A = \frac{K}{w \left( R^2 \left( 1 + \frac{C_C}{C_g} \right)^2 + \frac{1}{\omega^2 C_g^2} \right)^{0.5}} = 0.80135 \times 10^{-30}
\]

The Lift Field is expressed as

\[
E_{LO} = \left[ \frac{mg}{AB(\phi)r_0 \ln \left( \frac{r_0}{r_f} \right)} \right]^{0.5}
\]

\[
= \frac{2.71296 \times 10^{-6} \times 9.81}{1.5252 \times 10^{-18} \times 0.02062 \times 10^{-3} \times 76 \times 10^{-3} \times \ln \left( \frac{76 \times 10^{-3}}{27.5 \times 10^{-3}} \right)}
\]

\[
= 1.1600576 \times 10^6 \text{ kV/m}
\]
C.1 Flow Chart for Simulation of Particle Trajectories in a Gas Insulated Busduct

**ANNEXURE - C**

C.1 Flow Chart for Simulation of Particle Trajectories in a Gas Insulated Busduct

1. **READ**
   - Enclosure and particle dimensions: $r_0, r_1, h_0, k, r, l$
   - Particle density (pd), Power frequency voltage, SF6 gas pressure
   - Monte-Carlo angle, Runge-Kutta increment, Final time ($t_f$)

2. **INITIALIZE**
   - $y_0=0, z_0=0, \text{ind}=1, \text{ind}_1=1$

3. Compute the parameters
   - Lift-off field ($E_{l0}$)
   - Mass of the particle
   - $r, \theta_1, \theta_2, \text{and} \rho_2$ (for 3ph systems)

4. Calculate lift-off time ($t_l$) from the parameters obtained above

5. **A**
   - If $t_0 \neq t_f$
     - **Stop**
   - **Yes**
     - Compute $r, n, \text{and} \text{reynold's number}$

6. **B**
   - If $r < 5.0$
     - $K_d=e^{(0.1142+0.543\ln(re)+0.0516(\ln(re))^2)}$
   - **Yes**
     - $K_d=1$

**Start**
Print displacement $y_0$ in mm and time to in seconds

For Computing the displacement of a Vertical Particle, replace function(2) by function(3)

Yes

$Is \quad Ind=1$

$E_{l0}=E_{l0}$

Compute $s_1=dy(z_0)$ using function(1)

Compute $p_1=d2y(t_0, y_0, z_0, E_{l0}, p_0, m, K_d)$ using function(2))

$t_1=t_0+h/2$
$y_1=y_0+h.s_1/2$
$z_1=z_0+h.p_1/2$

Compute $s_2=dy(z_1)$ using function(1)

Compute $p_2=d2y(t_1, y_1, z_1, E_{l0}, p_0, m, K_d)$ using function(2))

$t_2=t_1+h$
$y_2=y_0+h.s_2/2$
$z_2=z_0+h.p_2/2$
Compute $s_2 = dy(z_1)$ using function(1)

Compute $p_2 = d^2y(t_1, y_1, z_1, E, p_0, m, K_d)$ using function(2)

$t_1 = t_0 + h$
$y_1 = y_0 + h \cdot s_2 / 2$
$z_1 = z_0 + h \cdot p_2 / 2$

Compute $s_4 = dy(z_1)$ using function(1)

Compute $p_4 = d^2y(t_1, y_1, z_1, E, p_2, m, K_d)$ using function(2)

$t_4 = t_0 + h$
$y_4 = y_0 + h \cdot s_4 / 2$
$z_4 = z_0 + h \cdot p_4 / 2$

$s = s_1 + 2 \cdot s_2 + 2 \cdot s_3 + s_4$
$p = p_1 + 2 \cdot p_2 + 2 \cdot p_3 + p_4$

$y_1 = y_0 + h \cdot s / 6$
$z_1 = z_0 + h \cdot p / 6$
$t_0 = t_1$

18

$y_0 < 0.00000002$

Yes

$\text{ind} = 2$

No

C
\[ y_{low} = 0.00009 \]

\[ L_s \]

\[ pd = 8900 \]

- **Yes**: \( L_s \)

- **No**: \( y_{low} = 0.0009 \)

\[ L_s \]

\[ (y_0 < \text{mod}(y_1) \text{ and } y_0 < y_{low}) \]

- **Yes**: \( \text{ind}_1 = 2 \)
  \[ z_1 = 0.8 \times z_1 \]
  \[ t_{\text{start}} = t_0 \]

- **No**: \( z_0 = z_1 \)

Using Monte-Carlo technique, compute:

\[ y_0 = \text{mod}(y_1) \]

\[ \theta = \text{angle ran} \]

\[ \theta = \theta_{\text{pt}}/180 \]

\[ L_s \]

\[ \theta < \text{angle/2} \]

- **Yes**:
  \[ y = y_0 \cos(\theta) \]
  \[ x = y_0 \sin(\theta) \]

- **No**: \( y = y_0 \cos(t) \)
  \[ x = -y_0 \sin(t) \]

Print Radial and Axial displacements obtained from Monte-Carlo technique.
C.2 Functions Used in the ‘C’ Language Software for simulation of Particle Trajectories in a Gas Insulated Bus duct

Double fv(et0, y0, m, yd0, rowg, kd)
    Double et0, t0, y0, m, yd0, rowg, kd;
    {
        Double k7, yk, k3, k4, k5, k6, k8, k9, k10;
        yk = fabs(yd0);
        k6 = (log(2.0*1/r) - 1.0);
        k3 = (pi*eps0*et0*v*sin(314*t0));
        k4 = (k6*(r0-y0)*(log(r0/ri)));
        k5 = m*gr+(yd0*pi*6.0*my*kd);
        k7 = (2.656*pi*r);
        k8 = (sprt(my*rowg*1));
        k9 = pow(yk, 1.5);
        k10 = ((k3/k4)-k5-(k7*k8*k9))/m;
        return(k10);
    }

double fh(et0, t0, y0, m, yd0, m, yd0, rowg, kd)
    double et0, t0, y0, m, yd0, rowg, kd;
    {
        Double k7, yk, k3, k4, k5, k6, k8, k9, k10;
        yk = fabs(yd0);
        k3 = (2*pi*eps0*r*1*et0*v*sin(314*t0));
        k4 = ((r0-y0)*log(r0/ri));
        k5 = m*gr+(yd0*pi*r*6.0*my*kd);
        k7 = (2.656*pi*r);
        k8 = (sprt(my*rowg*1));
        k9 = pow(yk, 1.5);
        k10 = ((k3/k4)-k5-(k7*k8*k9))/m;
        return(k10);
    }
Particle movement using Monte Carlo Technique

\[ \text{Theata} = \text{angle} \times \text{ran}; \]
\[ \text{Th}=\text{theate} \times \pi/180.0; \]
\[ \text{If}(\text{theata}<\text{angle}/2.0) \]
\[ \{ \]
\[ y=\text{y0} \times \cos(\text{th}); \]
\[ x+=-\text{y0} \times \sin(\text{th}); \]
\[ \} \]
\[ \text{Else} \]
\[ \{ \]
\[ y=\text{y0} \times \cos(\text{th}); \]
\[ x+=\text{y0} \times \sin(\text{th}); \]
\[ \} \]
\[ \text{fprintf(fp1,}"t\%10.41f\t%8.41f\t%-.d\n", \text{x*1000.0,}\text{y*1000.0,}\text{bou}); \]
\[ \text{if}(\text{ymax}<\text{y0}) \]
\[ \{ \]
\[ \text{ymax} = \text{y0}; \]
\[ \text{tmax}=\text{t0} - \text{h}; \]
\[ \text{xc}=\text{x}; \]
\[ \text{ymax1}=\text{y}; \]
\[ \} \]
\[ \} \]