CHAPTER 7

AFFINE ARITHMETIC BASED LOAD-FLOW ALGORITHM
FOR 1-PHASE RADIAL AND WEAKLY MESSED DELIVERY
NETWORKS WITH DATA UNCERTAINTIES

7.1 Introduction

This chapter applies ‘Affine Arithmetic’ (AA) tool [37] to estimate range load-flow solution for RWDNs with Data Uncertainties (DU). The AA based power-flow relation gives more accurate solution for the DU problem, than the standard IA [35] tool used in Chapter-6. The work also found efficient range load-flow solution, even for the networks with, ‘π-sections and tap-changing transformer’. The methodology is programmed and successfully demonstrated for both the RWDNs on IEEE Node-15 standard data[8] and test system data with Node-4.

It is not economically advisable to employ DU measuring devices everywhere along laterals/loops of RWDNs. Sometimes, it is unavoidable to operate networks with DU due to measurement errors, manufactures tolerance etc. The reliable ‘theoretical error analysis’ is expected to provide an accurate range-load-flow solution, when a specified system is subjected to DU like, variation in loads, lines, shunts, tap-changing transformer, generating capacities etc. The quantitative IA based load-flow [36] is found to be computationally superior over Monte Carlo simulations and stochastic load-flow methods. The IA algorithm
considers automatically the numerical round off error. Unfortunately, quantitative approach IA often yields an interval that is ‘much wider’ than the ‘exact range’ of the computed function. It means IA-tool utilizes the input data variation within relatively small intervals to obtain feasible solution. The IA-Tool saves the computation time, however, it gives less accurate results. The AA based range-load-flow equations are developed using Concept of Duality. Unlike IA-tool, the quantity representations used by AA-tool are ‘first order approximations’, whose mismatch/error generally is quadratic in the width of input DU. Hence, when compared to IA, higher asymptotic accuracy of AA compensates for the increased cost of its operation.

7.2 Affine and Interval Arithmatic relations

In IA a real quantity $x$ is represented by an interval $x_I = [x_{min} - x_{max}]$ of floating point numbers. Those intervals are added, subtracted, multiplied even for complex numbers for RWDNs, in such a way that each computed interval $x_I$ is confirms to contain the corresponding ideal quantity $x$—the (unknown) real value of the corresponding variable in the exact (error-free, uncertainty free)computation which being approximated. For example: If the ideal quantities $x$ and $y$ lies in the interval $x_I = [2_4]$ and $y_I = [-3_2]$, then the addition of $(x+y)$ lies in the interval i.e. $x_I+y_I = [2-3+0] = [-1_6]$ , whereas the product $x*y$ in [4*
Unfortunately, IA often yields an range that is 'much wider' than the actual range of computed function, As an example, the IA evaluation of $x - x$ given $x \in I_2 \rightarrow [-3_+, 3_+]$ instead of $[0_0]$, which is the actual range of that interval. In cascaded computations, due to the over- prediction factors multiplication of individual steps leads to the results of one-step are inputs for the next step.

Hence, AA were proposed to overcome the error explosion problem of the standard IA [35-36]. Besides noting down the range for each ‘ideal quantity’, AA also acts co-relater tool between those quantities. Because AA-Tool has a quadratic dependency on the dimension of errors, but for range analysis— finding confirmed upper-lower bounds for the value of a function $z = f(x, y)$ over a specified region of its domain.

Conversion between IA and AA: Every affine form, which is a first-degree polynomial form i.e. $\hat{x} = x_0 + x_1 \varepsilon_1 + \ldots + x_n \varepsilon_n$ implies an interval bound for the corresponding ideal quantity $x$: namely, $x \in I = [x_0 - r, x_0 + r]$, where $r$ is the total deviation of $\hat{x}$, $\sum_{i=1}^{n} |x_i|$. This is the smallest interval that contains all possible values of $\hat{x}$, assuming that each $\varepsilon_i$ ranges independently over the interval $U = [-1_+ + 1]$. Note: 'that the bounds of $x_I$ must be round outwards and this conversion discards all the correlation information present in $\hat{x}$.

Conversely, every ordinary interval bound $x_I = [a_+ b]$ for an ideal quantity $x$ can be replaced by an AA-form $\hat{x} = x_0 + x_k \varepsilon_k$, where $x_0$ is the
midpoint of \((a + b)/2\) of \(x_i\), \(x_k\) is the half-width \((b - a)/2\), and \(\varepsilon_k\) is a ‘new noise symbol’, not occurring in any other existing AA-form. The new symbol \(\varepsilon_k\) represents the uncertainty in the value of \(x\) that is implicit in its range \(x_i\). Again, note that \(x_0\) and \(x_k\) rounded carefully, and that the new AA-form, like the interval \(x_i\), carries no correlation information.

The important feature of AA-Tool is that \(\varepsilon_k\) may act as ‘Tuning Tool’ for inputs, outputs or intermediate results with \(\hat{x}\) and \(\hat{y}\) (two or more affine quantities). The detail AA expressions presented in the [Appendix 7A].

### 7.3 AA based load-flow Analysis for RDN with DU

The AA based load flow considers the path oriented RDN Chapter-4 with DU as shown in Fig. 7.1.

![Fig.7.1 Path directed AA representation for RDN with DU.](image-url)
7.3.1 Load data in affine form

A non-iterative AA analysis assumes load-data as constant current model. The Fig.7.1 shows the RDN with line and load DU. Here, the ‘conjugate of power to voltage’ directly taken as constant load currents.

\[ \hat{I}_i = (\hat{S}_i / \hat{V}_i)^* = \hat{I}_{id} - j \hat{I}_{iq} \]  

(7.1)

where,

\( \hat{I}_{id} \) = Real part of load current in AA-form

\( \hat{I}_{iq} \) = Imaginary part load current in AA-form

Further, first order AA-load currents are:

\[ \hat{I}_{id} = I_{od} + I_{1d}e_{1d} + I_{2d}e_{2d} + ... + I_{nd}e_{nd} \]

\[ \hat{I}_{iq} = I_{oq} + I_{1q}e_{1q} + I_{2q}e_{2q} + ... + I_{nq}e_{nq} \]  

(7.2)

7.3.2 Line data in affine form

The DU in line impedance for RDN are arranged in the incremental order of receiving-end nodes

\[
\hat{Z} = \begin{bmatrix}
\hat{Z}_{ji} & 0 & 0 \\
0 & \hat{Z}_{i,i+1} & 0 \\
0 & 0 & \hat{Z}_{i,n}
\end{bmatrix}
\]  

(7.3)

Then, first order AA-form impedances equal to:
\[ \hat{Z}_{ji} = Z_0 + Z_{(ji)} e_1 + ... + Z_{(ji)n} e_n \]
\[ \hat{Z}_{i,i+1} = Z_0 + Z_{(i,i+1)} e_1 + ... + Z_{(i,i+1)n} e_n \]
\[ \hat{Z}_{jn} = Z_0 + Z_{(jn)} e_1 + ... + Z_{(jn)n} e_n \]

where, \( Z_0 \) = Centre-point value; \( e_1, e_2, ..., e_n \) = Noise/co-relation symbols

### 7.3.3 Formulation of Affine Arithmetic based voltage relations

For the 15-Node RDN data[8] the paths \( P_i \) are marked as shown in Fig.7.2. The nodal AA-voltage expression based on the graphical information.

\[
[\hat{V}_i] = [\hat{V}_j] - \sum_{i=2,n}^{\text{Range of voltage drops - paths (Pi)}} [\hat{Z}_{ji}]
\]

\[
[\hat{V}_i] = [\hat{V}_j] - \sum_{i=2,j=1}^{\text{path nodes}} [\hat{B}_{ji}] \times [\hat{Z}_{ji}]
\]  \hspace{1cm} (7.4)

Now, the AA- voltages are given by:

\[
\hat{V}_i = [\hat{e}_i] + j* [\hat{f}_i] \hspace{1cm} \hat{V}_j = [\hat{e}_j] + j* [\hat{f}_j]
\]

Similarly, the affine branch currents can be expressed as

\[
[\hat{B}] = \begin{bmatrix}
\hat{B}_{ji} & j \hat{B}_{jiq} \\
\hat{B}_{i,i+1} & j \hat{B}_{i,i+1q} \\
\hat{B}_{in} & j \hat{B}_{inq}
\end{bmatrix}
\]  \hspace{1cm} (7.5)

In (7.3) and (7.5) receiving nodes for impedance and element currents are in incremental order i.e. \( 2,i, i+1,..n \). In case of sending end nodes may have different numbers.
The real and imaginary parts of (7.4) are:

\[
\hat{e}_i = \hat{e}_j - \sum_{j=1,i=2}^{Paths} (\hat{B}_{jid} \ast \hat{R}_{ji} + \hat{B}_{jiq} \ast \hat{X}_{ij}) \\
\hat{f}_i = \hat{f}_j - \sum_{j=1,i=2}^{Paths} (\hat{B}_{jid} \ast \hat{X}_{ji} - \hat{B}_{jiq} \ast \hat{R}_{ij})
\]

(7.6)

7.3.4 AA based multiplication operation to compute voltage

Consider the AA-voltage drop function as \( \hat{v}_{drop} \leftarrow \hat{B} \ast \hat{R} \) with first order operands: \( \hat{B} = 30 - 4\varepsilon_1 + 2\varepsilon_2 \); \( \hat{R} = 20 + 3\varepsilon_1 + 1\varepsilon_3 \)

Here, the operands are partially correlated the co-relation symbols \( \varepsilon_k \), (k =1,2..n). Then, it is expressed in AA using \( A \) as first order affine product and \( Q \) as pure quadratic residue, i.e.

\[
A(\varepsilon_1\ldots\varepsilon_n) = 600 + 10\varepsilon_1 + 40\varepsilon_2 + 30\varepsilon_3 \\
Q(\varepsilon_1\ldots\varepsilon_n) = (-4\varepsilon_1 + 2\varepsilon_2)(3\varepsilon_1 + 1\varepsilon_3)
\]

The two independent interval of \( Q \) are: \([-6_+ 6]; [-4_+ 4]\). Therefore, a quick estimate for the interval of \( Q \) equal to \([-24_+ 24]\), i.e. \( 0+24\varepsilon_4 \), in the first order affine format.

Then, the first order affine voltage-drop expressed as

\[
\hat{V}_{drop} = 600 + 10\varepsilon_1 + 40\varepsilon_2 + 30\varepsilon_3 + 24\varepsilon_4
\]
The $\hat{V}_{\text{drop}}$ comparisons are presented in Table.7.1 for different methods, which is helpful to estimate the strength of AA.

<table>
<thead>
<tr>
<th>Content</th>
<th>Exact Range</th>
<th>IA Range</th>
<th>AA Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{V}_{\text{drop}}$</td>
<td>[528_675]</td>
<td>[384_864]</td>
<td>[496_704]</td>
</tr>
<tr>
<td>Relative over estimation</td>
<td>1</td>
<td>3.26</td>
<td>1.42</td>
</tr>
<tr>
<td>Approximate error</td>
<td>Zero</td>
<td>Not considered</td>
<td>Quadratic residue</td>
</tr>
<tr>
<td>Correlation of operands</td>
<td>Not requires</td>
<td>Not considered</td>
<td>Noise variables</td>
</tr>
</tbody>
</table>

7.3.5 Development of AA nodal voltage relation

The Concept of Duality based load-flow of Chapter-4 is used to build AA based complex nodal voltage relations. This algorithm contains Forward-Path [$FP$] voltage-drop information.

$$
[FP] = \begin{bmatrix}
P_2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_3 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_4 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_5 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_6 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_7 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_8 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
P_9 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
P_{10} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
P_{11} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
P_{12} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
P_{13} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
P_{14} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
P_{15} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
\end{bmatrix}
$$

(7.7)

The voltage-drop is equal to the affine product of [$FP$] [$\hat{Z}$] and [$\hat{B}_j$]

$$
[\hat{V}_{\text{drop}}] = [FP][\hat{Z}_j][\hat{B}_j]
$$

(7.8)
The element current matrix of $[\hat{B}_{ji}]$ (7.8) contains Backward-Path node-currents information. For that, transpose the [FP] based on Concept of Duality to obtain $BP$ nodal currents.

$$\begin{bmatrix} \hat{B}_{ji} \end{bmatrix} = [FP]^T \begin{bmatrix} \hat{i}_i \end{bmatrix} \quad (7.9)$$

Substitute, (7.8-7.9) in (7.4) to obtain the affine nodal voltage

$$\begin{bmatrix} \hat{V}_i \end{bmatrix} = \begin{bmatrix} \hat{V}_j \end{bmatrix} - [FP] \begin{bmatrix} \hat{Z}_{ji} & 0 & 0 \\ 0 & \hat{Z}_{j,i+1} & 0 \end{bmatrix} [FP]^T \begin{bmatrix} \hat{i}_i \\ \hat{i}_{i+1} \\ \hat{i}_n \end{bmatrix} \quad (7.10)$$

Then first order affine voltage at nodes $i,i+1,\ldots,n$ are:

$$\hat{V}_i = \hat{V}_j - \hat{Z}_{ji} \hat{B}_{ji} \quad (7.11a)$$

$$\hat{V}_i = V_j - \left[ (Z_{j0}(B_{j0} + B_{j1}e_1 \ldots B_{jn}e_n) + B_{j0}(Z_{j1}e_1 \ldots Z_{jn}e_n) ) + (Z_{j1}e_1 \ldots Z_{jn}e_n)(B_{j1}e_1 \ldots B_{jn}e_n) \right] \quad (7.11b)$$

$$\hat{V}_{i+1} = \hat{V}_j - \{ \hat{Z}_{ji} \hat{B}_i + \hat{Z}_{i,i+1} \hat{B}_{i+1} \} \quad (7.12a)$$

$$\hat{V}_{i+1} = V_j - \left[ (Z_{j0}(B_{j0} + B_{jk}e_k) + B_{j0}(Z_{jk}e_k) + (Z_{jk}e_k)(B_{jk}e_k) ) + (Z_{j1}e_1 \ldots Z_{jn}e_n)(B_{j1}e_1 \ldots B_{jn}e_n) \right] + B_{i,i+1}(Z_{i,i+1}e_k + (Z_{i,i+1}e_k)(B_{i+1}e_k)) \quad (7.12b)$$

Similarly, express the voltages at, $i+2, i+3, \ldots, n$ nodes.
In Section 7.3.4) use (7.11-7.12) to perform AA-operation and then substitute in (7.10) to express the nodal voltage variation relation as

\[
\begin{bmatrix}
\hat{V}_i \\
\hat{V}_{i+1} \\
\hat{V}_n
\end{bmatrix} =
\begin{bmatrix}
V_i \\
V_{i+1} \\
V_n
\end{bmatrix} - \{(A_1 + A_2) + Q\}
\]  

(7.13)

where, \( A_1 + A_2 \) = Affinepart; \( Q \) = Quadratic residue part as given below:

\[
A_1 = [FP]
\begin{bmatrix}
Z_{ji0}\epsilon_0 & 0 & 0 \\
0 & Z_{(i,i+1)}\epsilon_0 & 0 \\
0 & 0 & Z_{ii0}\epsilon_0
\end{bmatrix}
[FP]^T
\begin{bmatrix}
I_{ik}\epsilon_k \\
I_{(i+1)k}\epsilon_k \\
I_{nk}\epsilon_k
\end{bmatrix}
\]

\[
A_2 = [FP]
\begin{bmatrix}
Z_{ji0}\epsilon_0 + Z_{jk}\epsilon_k & 0 & 0 \\
0 & Z_{(i,i+1)}\epsilon_0 + Z_{(i,i+1)}\epsilon_k & 0 \\
0 & 0 & Z_{ii0}\epsilon_0 + Z_{ink}\epsilon_k
\end{bmatrix}
[FP]^T
\begin{bmatrix}
I_{ik}\epsilon_k \\
I_{(i+1)k}\epsilon_k \\
I_{nk}\epsilon_k
\end{bmatrix}
\]

\[
Q = [FP]
\begin{bmatrix}
Z_{jk}\epsilon_k & 0 & 0 \\
0 & Z_{(i,i+1)}\epsilon_k & 0 \\
0 & 0 & Z_{ink}\epsilon_k
\end{bmatrix}
[FP]^T
\begin{bmatrix}
I_{ik}\epsilon_k \\
I_{(i+1)k}\epsilon_k \\
I_{nk}\epsilon_k
\end{bmatrix}
\]

In (7.13) the AA terms \( A_1, A_2 \) and quadratic residue term \( Q \) can be simplified using \( k = 1 \ldots n \) for the partially correlated operands through the shared-noise symbol \( \epsilon_k \). Finally, the magnitude of voltage of (7.13) is found using abs command of MATLAB.
7.4 Simulation and Case Studies for RDN with DU

The simulation and case studies for RDN with DU is carried out as:

7.4.1 Line and load parameter uncertainties

If RDN with DU ±5% and ±10% in line and load, respectively, then AA-results are obtained by tuning noise symbols in the equation $A_1$ and $A_2$ of (7.10-7.13). The calculation of quadratic residue impact variation can be seen only after 4th digits accuracy of results. Fig.7.2 shows that voltage variation is maximum at node-11 and minimum at node-2. For the independent cases like the line and load DU affine solution can be found using (7.13).

![Fig.7.2 Voltage variation with DU in the load and line data.](image)

154
7.4.2 RDN with Line, Load and \(\pi\)-section DU

The Fig. 7.3 shows both the RWDNs \(\pi\)-section with dotted line as separator between RDN and WDN.

Fig. 7.3  RWDNs with \(\pi\)-section intervals.

The RDN having \(\pi\)-section modifies the (7.13) at nodal load currents values

\[
\begin{bmatrix}
\hat{B}_{ji}^c \\
\hat{B}_{i,i+1}^c \\
\hat{B}_{in}^c \\
\end{bmatrix} = [FP]^{T} \begin{bmatrix}
\hat{I}_i \cdot \hat{I}_i^c \\
\hat{I}_{i+1} \cdot \hat{I}_{i+1}^c \\
\hat{I}_n \cdot \hat{I}_n^c \\
\end{bmatrix}
\]

(7.14)
In Fig. 7.3, consider the affine shunt currents \( \hat{Y}_{j0} \) and \( V_{io} \) as AA-operands. First, equate the \( \pi \)-section currents as the product element by element of AA-voltages \( (\hat{V}_2, \hat{V}_i, \hat{V}_n) \) with AA-admittance \( (\hat{Y}_j) \) and then express element currents as

\[
\begin{bmatrix}
\hat{B}_{12} \\
\hat{B}_{2i} \\
\hat{B}_{2n}
\end{bmatrix} = [FP]^T \begin{bmatrix}
\hat{I}_2 - \hat{Y}_{j2} \hat{V}_2 \\
\hat{I}_i - \hat{Y}_{j2} \hat{V}_i \\
\hat{I}_n - \hat{Y}_{j2} \hat{V}_n
\end{bmatrix}
\]  

(7.15)

Here, the term \( \hat{Y}_{j2} \hat{V}_i \) assumed as constant mid values due values of admittances are small and voltage at any node is nearly equal to unity.

Then, first order affine terms in \( A_1 \) and \( A_2 \) of (7.13) modified as

\[
A_1 = [FP] \begin{bmatrix}
Z_{120} & 0 & 0 \\
0 & Z_{i0} & 0 \\
0 & 0 & Z_{2n0}
\end{bmatrix} [FP]^T \begin{bmatrix}
I_{20} + I_{2k} e_k \\
I_{i0} + I_{ik} e_k \\
I_{n0} + I_{nk} e_k
\end{bmatrix}
\]

(7.16a-b)

\[
A_2 = [FP] \begin{bmatrix}
Z_{(12)k} e_k & 0 & 0 \\
0 & Z_{(2i)k} e_k & 0 \\
0 & 0 & Z_{(2n)k} e_k
\end{bmatrix} [FP]^T \begin{bmatrix}
I_{20} - Y_{i20} V_{20} \\
I_{i0} - Y_{i20} V_{i0} \\
I_{n0} - Y_{i20} V_{n0}
\end{bmatrix}
\]

Then, the Fig. 7.4 are presents voltage variation for 3-node DS with line, load and \( \pi \)-section DU.
Fig. 7.4 Voltages variation with DU in load, line and π-section.

### 7.4.3 Line, load, tap-transformer DU in RDN

The Fig. 7.5 shows the network having 1: a tap-changing transformer with affine series admittance of $\hat{Y}_t$.

Fig. 7.5 Tap-changing transformer in RDN with DU.
The addition of voltage regulating transformer in between nodes 2\(-n\) of the RDN modifies the impedance and load current relation in (7.13) as

\[
[\hat{Z}] = \begin{bmatrix}
(\hat{Z}_{12}+0) & 0 & 0 \\
0 & (\hat{Z}_{2i}+0) & 0 \\
0 & 0 & (\hat{Z}_{n}+\hat{Z}_{t})/a
\end{bmatrix}
\]

(7.17)

where, \([\hat{Z}_{r}] = \text{transformer leakage impedance}\)

Consider in Fig. 7.5 \((\hat{i}_{20})\) (non-tap side) and \((\hat{i}_{n0})\) (tap side) load-currents are leaving from nodes \(i\) and \(n\), respectively. The AA-element current part of (7.13) is

\[
[\hat{B}_{r}] = \begin{bmatrix}
\hat{B}_{12} \\
\hat{B}_{2i} \\
\hat{B}_{2n}
\end{bmatrix}
= [FP][\hat{i} - (O) - (\hat{i}_{20} - \hat{i}_{n0})]
\]

(7.18a)

In this case, consider the product of constant voltage and constant leakage admittance equal to \(\hat{i}_{20}, \hat{i}_{n0}\) currents.

\[
[\hat{B}] = \begin{bmatrix}
\hat{B}_{12} \\
\hat{B}_{2i} \\
\hat{B}_{2n}
\end{bmatrix}
= [FP]^T
\begin{bmatrix}
\hat{i}_2 - (\frac{a-l}{a})\hat{Y}_t\hat{V}_2 \\
\hat{i}_n - (\frac{l-a}{a'})\hat{Y}_t\hat{V}_n
\end{bmatrix}
\]

(7.18b)
where, \[ \left( \frac{a-l}{a} \right) \hat{Y}_1 \hat{V}_2 = NT; \quad \left( \frac{l-a}{a^2} \hat{Y}_t \hat{V}_n \right) = T \]

Use (7.18a-b) to modify \( A_1 \) plus \( A_2 \) terms of (7.13) containing the impedances and currents as

\[
A_1 = \begin{bmatrix} Z_{(i2)0} & 0 & 0 \\ 0 & Z_{(2i)0} & 0 \\ 0 & 0 & Z_{(2n)0} + Z_{(i1)0} \end{bmatrix} \begin{bmatrix} I_{20} + I_{2k}e_k \\ I_{i0} + I_{ik}e_k \\ I_{n0} + I_{nk}e_k \end{bmatrix}
\]

\[
A_2 = \begin{bmatrix} Z_{(i2)i0}e_0 & 0 & 0 \\ 0 & Z_{(2i)i0}e_0 & 0 \\ 0 & 0 & Z_{(2n)i0}e_0 \end{bmatrix} \begin{bmatrix} I_{20} + I_{2k}e_k \\ I_{i0} + I_{ik}e_k \\ I_{n0} + I_{nk}e_k \end{bmatrix}
\]

(7.18c)

![Fig. 7.6 AA-IA voltages with DU in line, load and regulating transformer.](image)

The AA-nodal voltage is boosted with from tap to non-tap side of the transformer i.e. in between node number 1 and 3 can be seen in Fig. 7.6.
7.5 AA Load Flow for WDN with DU

The open-tie-switches in DS normally close to create loops. In such case, a new element creates the loop in between \(i^{th}\) node (higher potential) to \(j^{th}\) node (lower potential) as in Fig. 7.7.

![Diagram](image)

Fig.7.7 Single loop WDN with DU in line and load.

7.5.1 Affine line data matrix

The element addition records in last row and column of (7.3) as

\[
[\hat{Z}_w] = \begin{bmatrix} \hat{Z} \text{ Rad} \end{bmatrix}
\]

\[
\begin{bmatrix}
0 \\
\vdots \\
0 \\
\end{bmatrix}
\rightarrow
\begin{bmatrix} l & 0 & \cdots & 0 \\
\vdots \\
\end{bmatrix}
\]

\[
[\hat{Z}_{ji}]
\]

(7.19)
7.5.2 Weakly meshed forward path matrix

The forward path matrix $[F_w]$ develops with addition of following two steps in (7.7) as in Chapter-4 Concept of Duality applied for WDN.

\[
\begin{bmatrix}
P_2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_3 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_4 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_5 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_6 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_7 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_8 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_9 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
P_{10} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
P_{11} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
P_{12} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
P_{13} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
P_{14} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
P_{15} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
B_i & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 1
\end{bmatrix}
\]

(7.20)

7.5.3 Affine element current matrix

The element current matrix is computed by inserting the loop element current $B_i$ in the row (new) i.e. $l^{th}$ of (7.5).

\[
\begin{bmatrix}
\hat{B}_w
\end{bmatrix} = \begin{bmatrix}
\hat{B}_{radial} \\
\hat{B}_i
\end{bmatrix}
\]

Then, the element current equals to the product of transpose of $[F_w]$ and $[\hat{i}_w]$ using (7.12), i.e.

\[
\begin{bmatrix}
\hat{B}_w
\end{bmatrix} = [F_w]^T [\hat{i}_w]
\]

(7.21)
### 7.5.4 AA-based Nodal voltage matrix

The AA-voltage modifies for WDN having link impedance between $i^{th}$ to $j^{th}$ node.

\[
\hat{V}_{iw} = \hat{V}_j - [F_w][\hat{Z}_w][F_w]^T \hat{i}_w
\]  

(7.22)

Then, in (7.13) $A_1$, $A_2$ and Q term can be written as:

\[
A_{w1} = [FP_w]^T \begin{bmatrix} Z_{j0} & 0 & 0 & 0 \\ 0 & Z_{(i,j+1)0} & 0 & 0 \\ 0 & 0 & Z_{(i,n)0} & 0 \\ 0 & 0 & 0 & Z_{ji0} \end{bmatrix} [FP_w] \begin{bmatrix} I_{i0} + I_{ik} \varepsilon_k \\ I_{(i+1)0} + I_{(i+1)k} \varepsilon_k \\ I_{n0} + I_{nk} \varepsilon_k \end{bmatrix}
\]  

(7.23)

\[
A_{w2} = [FP_w]^T \begin{bmatrix} Z_{jik} \varepsilon_k & 0 & 0 & 0 \\ 0 & Z_{(i,j+1)k} \varepsilon_k & 0 & 0 \\ 0 & 0 & Z_{nk} \varepsilon_k & 0 \\ 0 & 0 & 0 & Z_{ijk} \varepsilon_k \end{bmatrix} [FP_w] \begin{bmatrix} I_{i0} \\ I_{(i+1)0} \\ I_{n0} \end{bmatrix}
\]

\[
Q_w = [FP_w]^T \begin{bmatrix} Z_{jik} \varepsilon_k & 0 & 0 & 0 \\ 0 & Z_{(i,j+1)k} \varepsilon_k & 0 & 0 \\ 0 & 0 & Z_{nk} \varepsilon_k & 0 \\ 0 & 0 & 0 & Z_{ijk} \varepsilon_k \end{bmatrix} [FP_w] \begin{bmatrix} I_{ik} \varepsilon_k \\ I_{(i+1)k} \varepsilon_k \\ I_{nk} \varepsilon_k \end{bmatrix}
\]
7.5.5 Kron’s reduction algorithm

Arrange (7.23) and Apply the Kron’s reduction in of WDN relation to reduce the $A_1$, $A_2$ and $Q$ dimension and find the affine nodal voltage as

$$
\begin{bmatrix}
\hat{V}_i \\
\theta
\end{bmatrix} = [(A_{w_1} + A_{w_2} + Q_w)]_{n \times n} \begin{bmatrix}
\hat{i}
\end{bmatrix}$$

$$
\begin{bmatrix}
\hat{V}_i \\
\theta
\end{bmatrix} = [(A_{w_1} + A_{w_2} + Q_w)]_{n-1 \times n-1} [I_i]
$$

(7.24)

7.6 Simulation and Case Studies for WDN

AA simulation and case studies are carried out to obtain range solution for weakly meshed distribution networks with Data Uncertainties are discussed as:

7.6.1 Uncertainties in line and load parameters

The line DU ±5% and load DU ±10% variation considered in this case. Then the (7.21-7.24) computes results and Fig.7.8 represents voltage variation for the modified WDN data [8].

![Fig. 7.8 Voltages variation with DU in load and line.](image-url)
7.6.2 Uncertainties in line-load and \( \pi \)-section parameters

The line section with \( \pi \) form can be modeled as shown in Fig. 7. 3. Then consider the opposite sign half line charging currents at node \( i \) and \( n \) in \( \hat{\mathbf{B}}^c_w \) as

\[
\begin{bmatrix}
    \hat{B}^c_{12} \\
    \hat{B}^c_{2i} \\
    \hat{B}^c_{2n} \\
    \hat{B}^c_l
\end{bmatrix}
= 
\begin{bmatrix}
    1 & 1 & 1 & 0 \\
    0 & 1 & 0 & 1 \\
    0 & 0 & 1 & -1 \\
    0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
    \hat{I}^c_2 - \hat{I}^c_{12} \\
    \hat{I}^c_i - \hat{I}^c_{2i} = \left( \frac{i^c_i}{2} \right) \\
    \hat{I}^c_n - \hat{I}^c_{2n} = \left( \frac{i^c_n}{2} \right) \\
    \hat{B}_l
\end{bmatrix}
\]

where
\[
\frac{i^c_i}{2} = \frac{\hat{Y}^c_{gi} \hat{V}_i}{2}; \quad \frac{i^c_n}{2} = \frac{\hat{Y}^c_{gn} \hat{V}_i}{2}
\]

The half line charging currents of network at nodes \( i \) and \( n \) considers as constant voltage sources. From (7.25) compute affine shunt currents and substitute in (7.23) to obtain the result as shown in Fig. 7.9.

![Voltage Variation with input Parameter and Shunt Currents](image)

**Fig. 7.9** Node voltage variation with line, load and shunt DU in WDN.
7.6.3 Uncertainties in line, load and tap-transformer parameters

In Fig.7.8 the WDN with tap-changing transformer is created by placing transformer between node $i$ and $n$. Then in (7.23) WDN impedance and element current relations are can be modified using (7.3, 7.18) as

$$
[\hat{Z}_{wt}] = \begin{bmatrix}
[\hat{Z}_{\text{radial}}] & \text{new} - \text{row} \\
\text{new} - \text{column} & \hat{Z}_t + Z_t / a
\end{bmatrix}
$$  \hspace{1cm} (7.26)

where, $Z_l$ = loop impedance

$Z_t$ = Transformer leakage impedance

$$
[\hat{B}_{wt}] = \begin{bmatrix}
\hat{B}_{i2} \\
\hat{B}_{2i} \\
\hat{B}_{2n} \\
\hat{B}_1
\end{bmatrix} = \begin{bmatrix}
1 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
\hat{I}_2 \\
\hat{I}_{il} - (\hat{I}_{iNT}) \\
\hat{I}_{ln} - (\hat{I}_{nT}) \\
\hat{B}_1
\end{bmatrix}
$$  \hspace{1cm} (7.27)

The AA-currents in Fig. 7.8 Non-Tap ($NT$) and Tap side ($T$) from leaving from nodes $i$ and $n$, respectively. Then element relation is modified as

$$
[\hat{B}_{wt}] = \begin{bmatrix}
1 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
\hat{I}_2 \\
\left(\frac{a-1}{a}\right) \hat{Y}_i \hat{V}_i \\
\left(\frac{a-1}{a}\right) \hat{Y}_n \hat{V}_n \\
\hat{B}_1
\end{bmatrix}
$$  \hspace{1cm} (7.28)

To compute affine result as shown in Fig.7.10 (7.26-7.28) used in (7.23) for the addition of tap-changing transformers in the line section of WDN.
7.7 Results and Discussions for RWDNs

AA based load flow method for RWDNs with DU is proposed to a non-iterative range power flow solution. The AA algorithm obtained the accurate enclosures for computed results, taking into account any variation in the input data as well as all internal truncation and round off errors. The advantages of proposed method are satisfied as per Table.7.1, wherein AA based results are compared with Exact and IA based results for both Radial and Weakly Meshed Electrical Load-Delivery Networks. The voltage profile results considers the DU in line parameter of ± 5% interval and load parameter of ± 10% interval. It computes results for the DU cases like, when line with constant load-data; load-data with constant line-parameter and DU of both line-load using (7.13) for RDN. Similarly, AA-results for line & load DU are obtained by $A_1$, $A_2$ affine terms and $Q$ quadratic residue equation (7.23) for WDN.
The accurate affine solutions are obtained by tuning the ‘shared noise symbols’ (i.e. $\varepsilon_2, \varepsilon_3, \ldots, \varepsilon_n$ in between the range $\{+0.1, -0.1\}$) and by setting quick estimate of quadratic residue $0.15 \times 0.2^*Q$ for the RWDNs under study. Fig.(7.2) shows the results for 15 node [8] RDN having high R/X with line-load parameter uncertainties. The parameter uncertainties effects are found as minimum at $2^{nd}$ node, medium at $4^{th}$ node, and maximum at $11^{th}$ node, respectively. This indicates that the medium effected node can be considered as node-1 to study the impact of parameter uncertainties in the RWDNs. Fig.7.4 presents voltage in the AA form for the 4-node of data given in [Appendix 7B] RDN having shunt currents. The shunt currents computed using ‘constant line shunt admittance’ and ‘voltage as 1.0. p.u’ values. Then, these shunt current values are substituted in (7.13) to compute AA nodal voltages. For the 4-node system (of Appendix 7B). Fig.7.4, 7.6, 7.9.7.10) the affine results are plotted by numbering $1,2,3$, instead $2,i,n$ to show in incremental order of voltage display.

Further, the RDN with tap-changing transformer the model (7.17-7.18) are found useful to consider the uncertainty effects on both line as well as load parameters. The affine voltage values for the fixed tap-ratio i.e. 1:1.15 are presented in Fig.7.6. It is found that at Non-tap the nodal AA voltage variation is more, when compared to Tap-side of the transformer.

In case of WDN with line parameter interval of ± 5% and load parameter of interval ± 10% the results are affine computed using (7.13).
The nodal voltage calculation presented in Fig. 8 for AA and IA using modified DS [8] data and [Appendix 7 B] data., it is seen that in the load and line parameter uncertainties effects are reduced, due to improvement in the voltage profile. The impact changes nodal voltage as: minimum at 2\textsuperscript{nd} node, medium at 13\textsuperscript{th} node and maximum at 2\textsuperscript{nd} node. The accurate AA results considers uncertainties linked with element impedances shown in Fig.7.10. As in RDN load-flow, in weakly meshed network also requires tuning of shared noise symbols i.e. \( \epsilon_2, \epsilon_3, \ldots \epsilon_n \) in between \{+1,-1\} range to find accurate results.

The affine voltages of (7.23) relations are found to obtain results for the 4-node data and [7B-Appendix] data, for WDN with shunt currents. In (7.25), the product of constant shunt admittance and voltage leads to shunt currents values substituted in (7.23) to compute the nodal voltages. In Fig.7.9 presents the AA results with buck voltages. Further, the (7.27-7.28) considers the network with tap-changing transformer the model having uncertainty effects of on both line as well as load parameter. The affine voltage values transformer with fixed tap-ratio i.e. 1:1.15 are presented in Fig.7.10. It is found that at Non-tap the nodal voltage variation is more, when compared to Tap-side of the transformer.

The proposed AA tool can take care of wider range of variation load, line and changing with DU including the tap-changing transformer ratio also. The AA method is programmed and demonstrated using MATLAB Ver. 7.01; Windows; Pentium-IV; 2.8GHz; 248 MB RAM. The opted load
method belongs to Chapter 4, which takes program execution time is less than the conventional FP⇒BP load flow methods. Hence, the time of affine operation also decreases. Now, it is understood that AA acts as best quantitative tool to find accurate and fast for RWDNs range load flow solution.

### 7.8 Conclusions

The self-validated AA tool found superior judge over IA to solve Data Uncertainties for both Radial and Weakly meshed Delivery Networks. The analysis also deals successfully, even for both the networks having π-section and tap-changing transformer. First, affine relation organizes uncertain data and then applies Concept of Duality based load flow to compute the nodal variation of voltage using AA operations on closed intervals. The AA based relations are useful to avoid the overestimation effects of IA on final range solution. The AA method shared noise variable plays a co-relation role with quadratic residue for more practical accurate results. IA may require only during the initial stages of planning load-flow RWDNs. Anytime, IA method saves on time and number of operations required, whereas AA method compensates with results that are more accurate. The AA method is feasible even for large systems and independent of the variation in the input data.
APPENDIX-7 A

If $f$ represents an affine operation, the affine representation for $\hat{z}$ is obtained by expanding and re-arranging into affine form the with noises $\varepsilon_k$.

Affine operation:

\[ \hat{x} \pm \hat{y} = (x_o \pm y_o) + (x_1 \pm y_1)\varepsilon_1 + (x_2 \pm y_2)\varepsilon_2 + \ldots \]
\[ \ldots (x_n \pm y_n)\varepsilon_n = (x_o \pm y_o) + (x_k \pm y_k)\varepsilon_k \]

(A7.1)

\[ \alpha \hat{x} = (\alpha x_o) + (\alpha x_1)\varepsilon_1 + (\alpha x_2)\varepsilon_2 + \ldots + (\alpha x_n)\varepsilon_n \]
\[ = (\alpha x_o) + (\alpha x_k)\varepsilon_k \quad \forall \alpha \in R \]

(A7.2)

\[ \hat{x} \pm \lambda = (x_o \pm \lambda) + x_1\varepsilon_1 + x_2\varepsilon_2 + \ldots + x_n\varepsilon_n \]
\[ = (x_o \pm \lambda) + x_k\varepsilon_k \quad \forall \lambda \in R \]

(A7.3)

where, $k = 1, 2, 3..n$ in eqs(1-3)

Non-affine operations: Now consider a generic non-affine operation $z \leftarrow f(x, y)$. If $x$ and $y$ are expressed by the AA forms, then the ideal quantity $z$ is

\[ z = f(x_o + x_1\varepsilon_1 + \ldots x_n\varepsilon_n, y_o + y_1\varepsilon_1 + \ldots y_n\varepsilon_n) \]
\[ z = f^*(\varepsilon_1\ldots\varepsilon_n) \]

(A7.4)
where \( f^* \) is a function from \( U^n \) to \( \mathbb{R} \). If \( f^* \) is not affine, then \( z \) can be expressed as an affine combination of the noise symbols \( \epsilon_i \), that approximates \( f^a(\epsilon_1...\epsilon_n) \) well over its domain \( U^n \).

\[
\hat{z} = f^a(\epsilon_1...\epsilon_n) = z_0 + z_1\epsilon_1 + ... + z_n\epsilon_n
\]

(A7.5)

Now, we have an additional term \( z_k\epsilon_k \) to compensate the error introduced by this approximation as

\[
\hat{z} = f^a(\epsilon_1...\epsilon_n) = z_0 + z_1\epsilon_1 + ... + z_k\epsilon_k
\]

(A7.6)

Then, approximation error is

\[
z_k\epsilon_k = f^e(\epsilon_1...\epsilon_n) = f^*(\epsilon_1...\epsilon_n) - f^a(\epsilon_1...\epsilon_n)
\]

(A7.7)

The special noise symbol \( \epsilon_k \) of (6-7) this does not occur in any affine form. The \( z_k \) allowed rounding \( z_k \) conservatively i.e. away from zero and it does not imply any correlations. The coefficient \( z_k \) must be an upper bound on the absolute magnitude of the approximation error free, i.e.

\[
|z_k| \geq \max \left\{ \left| f^*(\epsilon_1...\epsilon_n) - f^a(\epsilon_1...\epsilon_n) \right| : (\epsilon_1...\epsilon_n \in U) \right\}
\]

(A7.8)

The error coefficient of \( z_k \) must also consider any round off errors incurred in the computation of the other coefficients such as \( z_0, z_1...z_n \). However, at the same time an approximation function is one, which reasonably accurate and not computationally expensive.
To attain simple and an efficient function, only normal approximation function, which are affine combinations of the input forms as

$$\hat{z} = f^a(\varepsilon_1,...\varepsilon_n) = \alpha \hat{x} + \beta \hat{y} + \gamma$$  \hspace{1cm} (A7.9)

Further, among normal approximates the good selection for $f^a$—in the sense of minimizing the error term $|z_k|$—is the one which minimizes the maximum absolute error as

$$\left|\alpha \hat{x} + \beta \hat{y} + \gamma - f^a(x,y)\right|$$  \hspace{1cm} (A7.10)

where the unknown function coefficients, $\alpha$, $\beta$ and $\gamma$ can be identified based on Chebyshevs approximation theory for univariate function. The $x$ and $y$ range over a polygon $\{\hat{x}, \hat{y}\}$. This is called Chebyshev (minimax) affine approximation of $f$ over $\{\hat{x}, \hat{y}\}$.

**APPENDIX 7-B**

Table B 7.1 Radial and weakly meshed data for 4-node system

<table>
<thead>
<tr>
<th>Nodees</th>
<th>Line-Data = $Z_{ij}$</th>
<th>$jB_{ij}$</th>
<th>Load-Data= $S_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>0.09545 + j*0.07520</td>
<td>j*0.0007</td>
<td>0.0208 + j*0.0021i</td>
</tr>
<tr>
<td>2-3</td>
<td>0.03300 + j*0.01849</td>
<td>j*0.0015</td>
<td>0.0495 + j*0.0051i</td>
</tr>
<tr>
<td>2-4</td>
<td>0.03100 + j*0.01390</td>
<td>j*0.0353</td>
<td>0.0958 + j*0.0098i</td>
</tr>
<tr>
<td>3-4</td>
<td>0.90500 + j*0.28900</td>
<td>j*0.0367</td>
<td>------------------</td>
</tr>
</tbody>
</table>