openXDA at Dominion

Kyle Thomas

2014 GPA 4th Annual User's Forum

8/13/2014

Dominion Traditional FL Method

Step 1

- Download event records and fault summaries from DFRs, TWS, Digital Relays

Step 2

- Check for lightning correlation

Step 3

- Open event records in viewer and perform manual FL analysis

Step 4

- Compare all results, use engineering judgment to determine best FL to provide to field personnel

Dominion Traditional FL Method

Step 3

- Open event records in viewer and perform manual FL analysis

This step takes largest amount of time (assuming Step 1 is automated), but it is critical to getting good FL results, especially when:

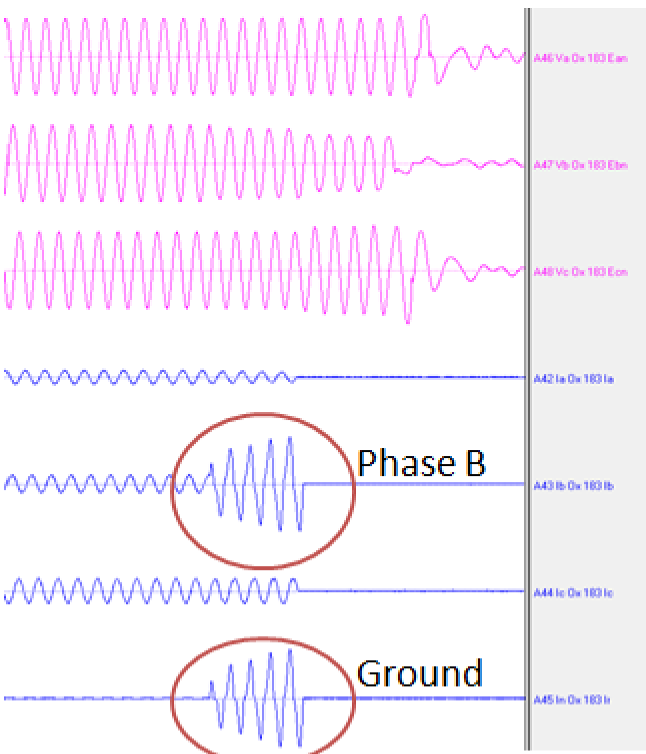
- DFRs and/or Relays fail to auto-calculate FLs
- DFRs and/or Relays fail to auto-calculate FL in fault window
- FL algorithms in the DFRs and/or Relays have significant errors under certain fault types or conditions

Dominion Traditional FL Method

openXDA can really help automate the manual process of Step 3

Here is our manual process for Step 3:

- 3A: Open event record(s) in viewer. We primarily use the WaveWin software
- 3B: Identify and select the faulted waveforms:



Faulted Line: Line 183 (Bristers to Ox)

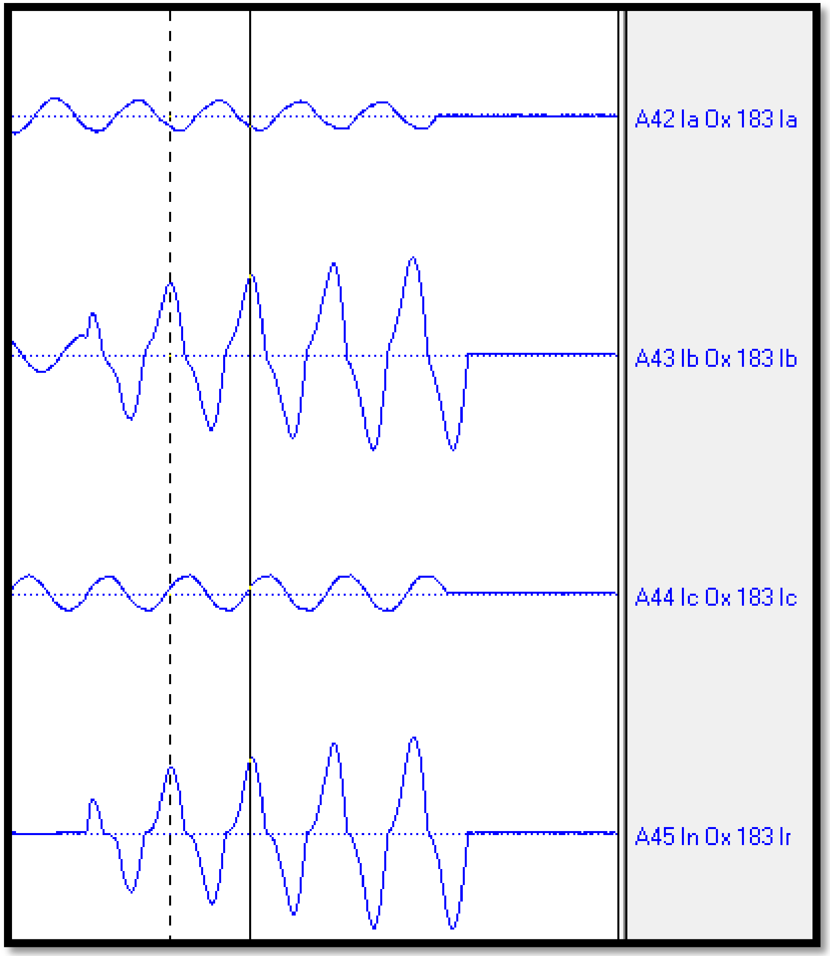
Fault Type: Phase B to Ground

Additional information:

- Faulted current waveforms are growing throughout the fault.

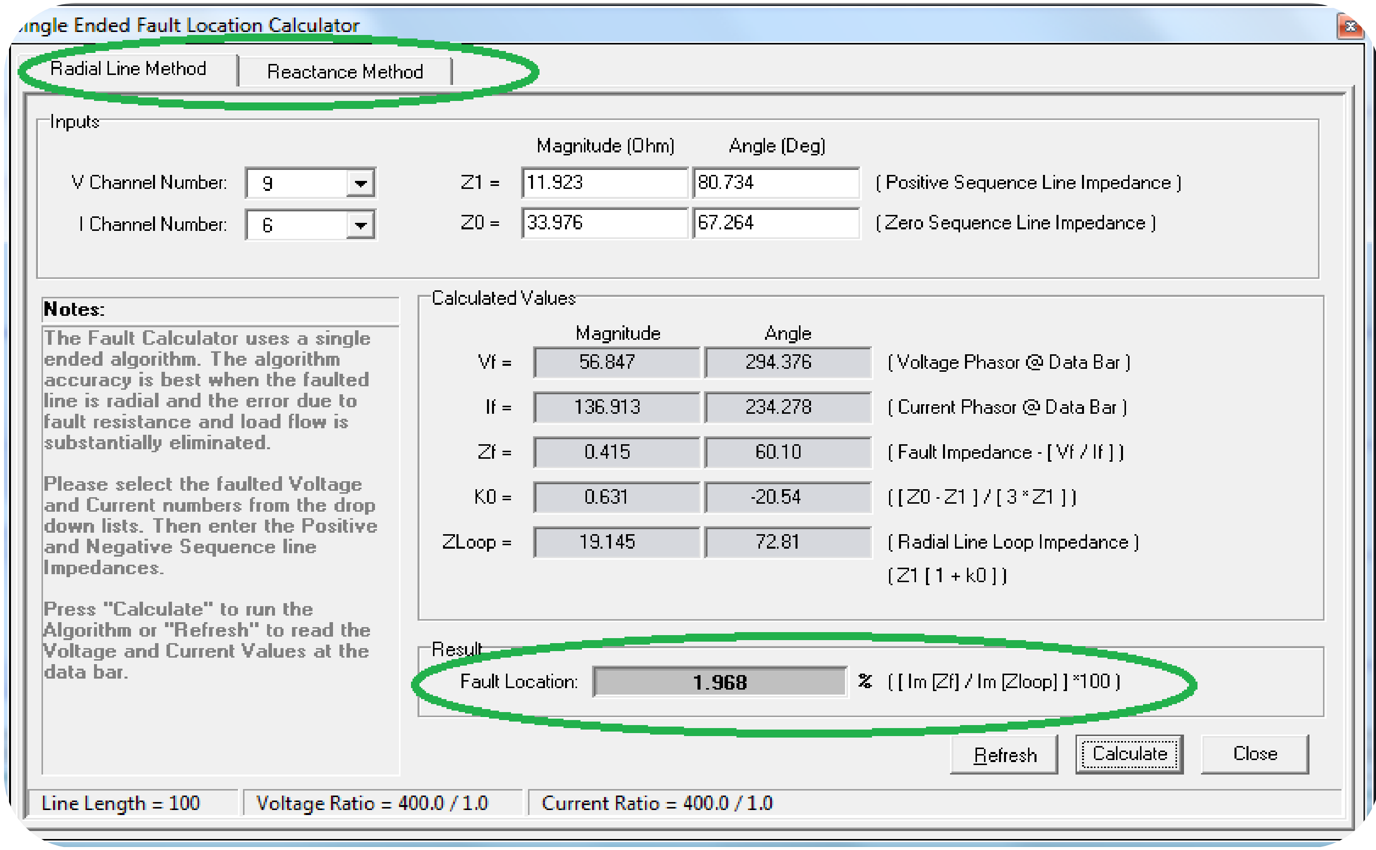
Dominion Traditional FL Method

- 3C: calculate a FL using one algorithm at a specific point in the fault window



Dominion Traditional FL Method

- 3D: use all available algorithms to produce FLs at the same point [Currently 3 algorithms available in WaveWin]

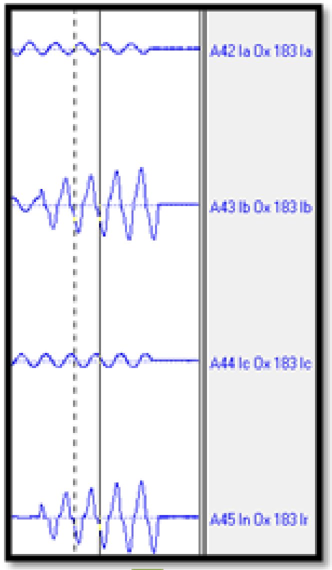


Ex: **Reactance Algorithm 1** = 1.968% * 23.54 miles = 0.46 miles from Ox

Dominion Traditional FL Method

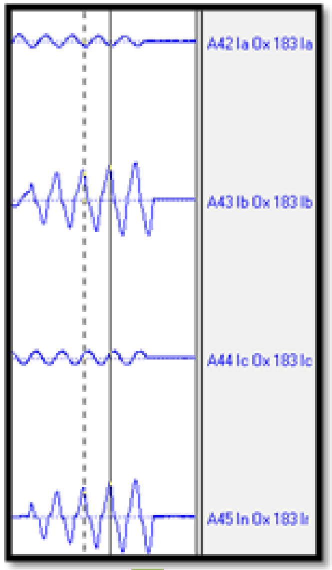
- 3E: repeat steps 3C and 3D using different points along the fault

Point #2



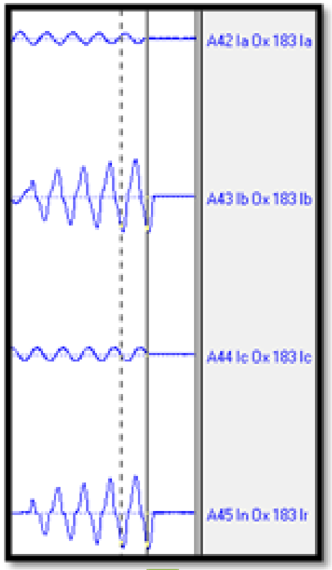
Alg 1 = 0.21 mi from Ox
Alg 2 = 0.51 mi from Ox
Alg 3 = 0.32 mi from Ox

Point #3



Alg 1 = 0.41 mi from Ox
Alg 2 = 0.49 mi from Ox
Alg 3 = 0.39 mi from Ox

Point #4



Alg 1 = 0.50 mi from Ox
Alg 2 = 0.60 mi from Ox
Alg 3 = 0.42 mi from Ox

Dominion Traditional FL Method

- 3F: Use engineering judgment to compare all results and find the best FL

Point #1

Alg 1 = 0.46 mi from Ox
Alg 2 = 0.42 mi from Ox
Alg 3 = 0.36 mi from Ox

Point #2

Alg 1 = 0.21 mi from Ox
Alg 2 = 0.51 mi from Ox
Alg 3 = 0.32 mi from Ox

Point #3

Alg 1 = 0.41 mi from Ox
Alg 2 = 0.49 mi from Ox
Alg 3 = 0.39 mi from Ox

Point #4

Alg 1 = 0.50 mi from Ox
Alg 2 = 0.60 mi from Ox
Alg 3 = 0.42 mi from Ox

Ox DFR Auto-FL =	0.41 mi from Ox
Ox SEL Auto-FL =	0.70 mi from Ox
Bristers DFR Auto- FL =	3.21 mi from Ox
Bristers SEL Auto-FL =	3.51 mi from Ox
FALLS correlation =	N/A
TWS Auto-FL =	N/A

Best Fault Location

0.42 miles from Ox

Report this location
to SOC & Lines Crew

Dominion Traditional FL Method

Problems with this manual FL process:

- Takes time, 15+ minutes, to collect all results for analysis
- Small number of results
- Only 3-5 algorithms used. We have identified 9+ FL algorithms

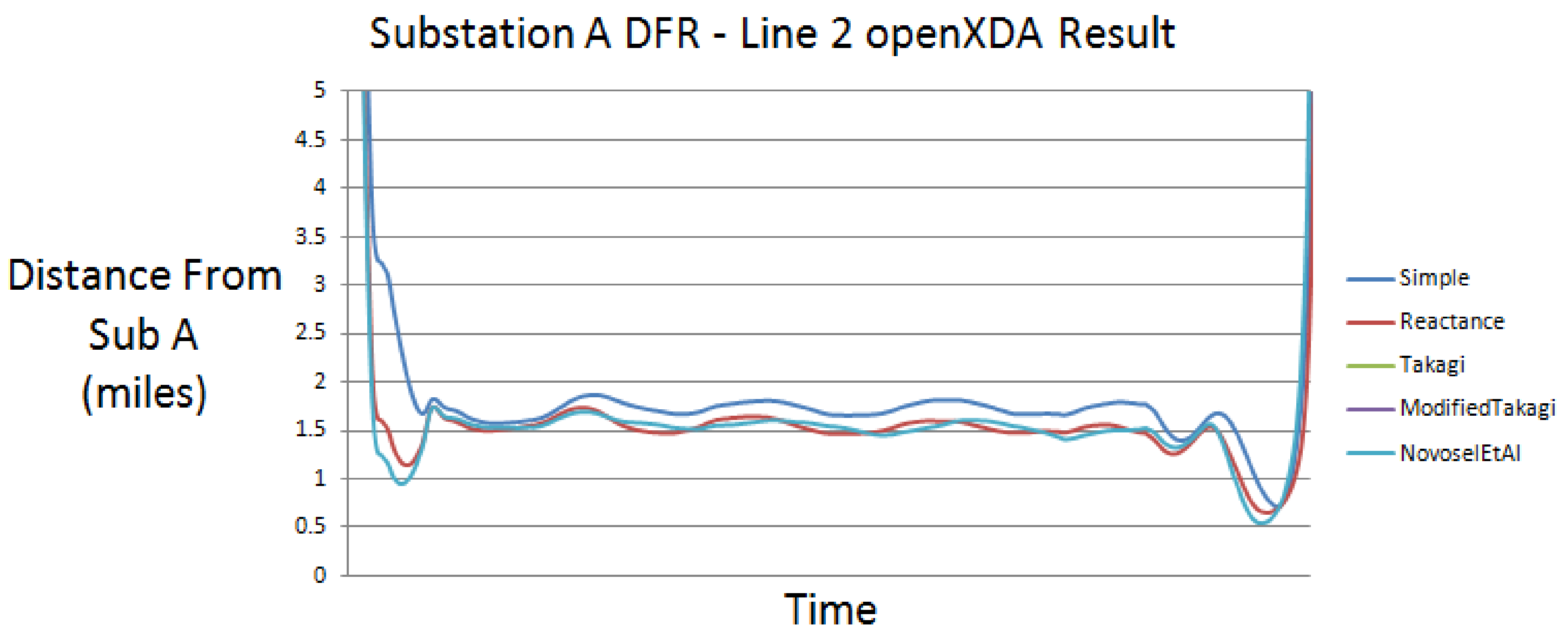
openXDA is designed to calculate fault locations across an entire event record with all available FL algorithms, automatically

Instead of spending time making results, spend that time (or save that time) by analyzing the fault with all the results

openXDA – FL across entire record

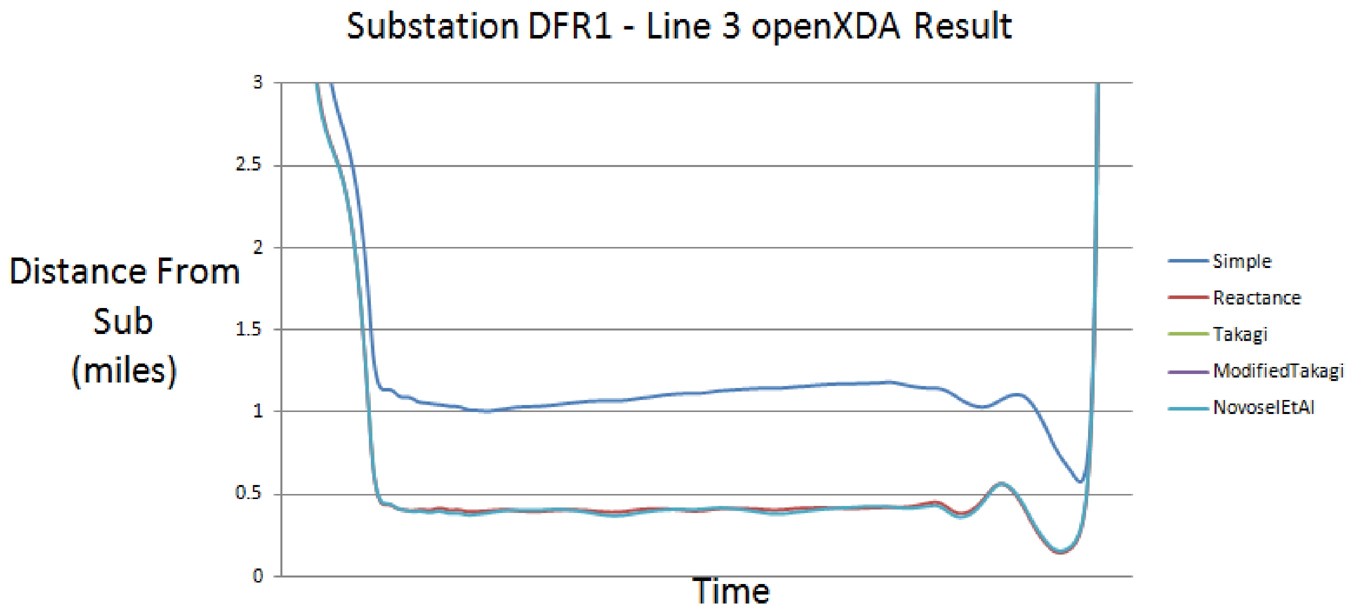
Fault Curves with all algorithms

- Instead of picking a few points in the fault window, openXDA will calculate the FL (with all algorithms) across the entire fault window



openXDA – FL across entire record

1. Provides as many FL results as possible
2. Can be used with all available algorithms, and any future algorithms developed
3. Can help identify when algorithms should and should not be used (or which algorithms should be avoided entirely)



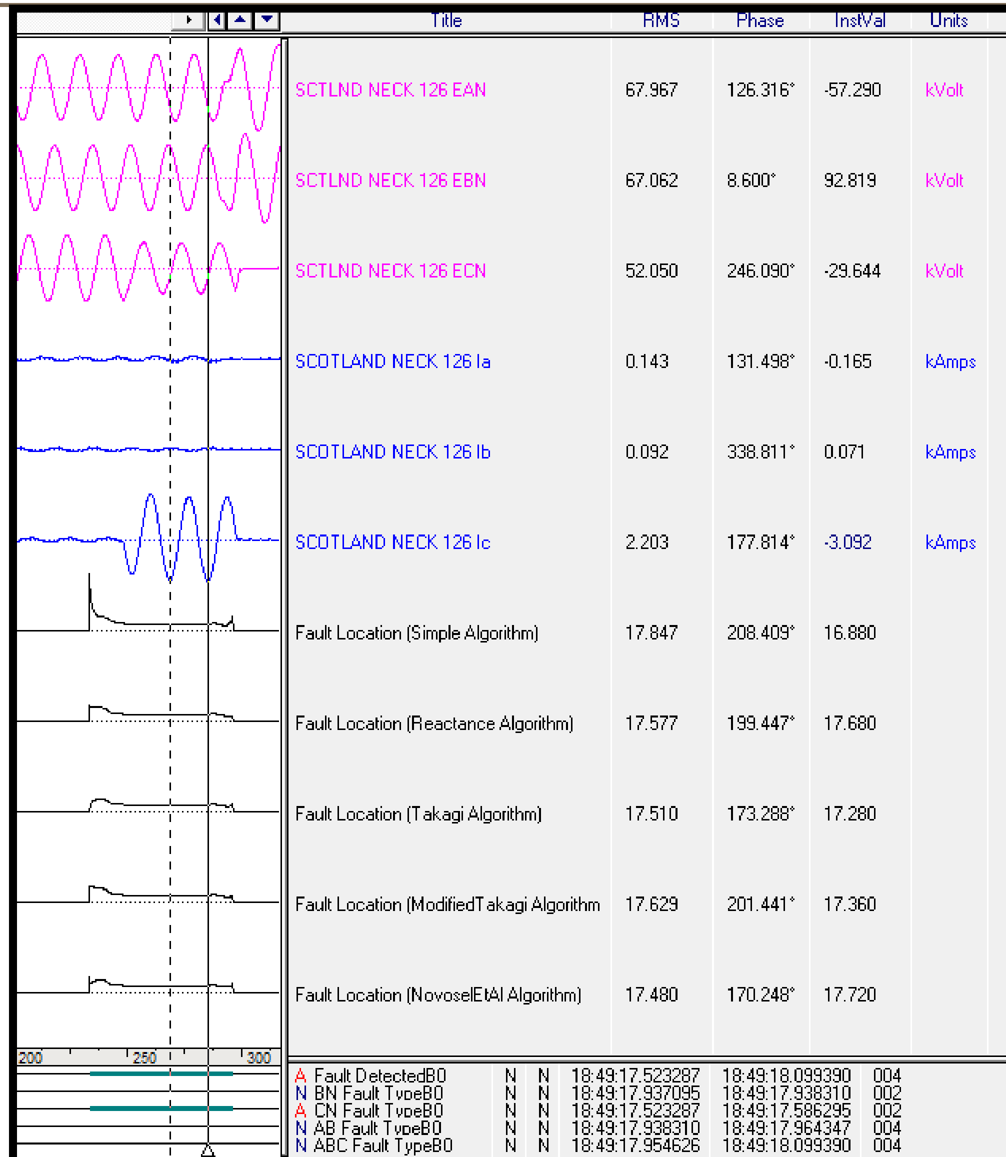
openXDA – Automation

openXDA will produce these FL results automatically

- The application watches a folder, and any new Comtrade file added to the folder or subfolder will be detected and processed by openXDA
- It will go through all line groups in a file and look for a fault
- If a fault is detected on any line, openXDA will run through the calculations on the faulted lines and produce the FL curves
- Data is saved to a SQL database, so plots can be made with Excel
- But that took time to plot in Excel…
- So we dreamed up the idea of having an openXDA COMTRADE Results File
  - For any line that a fault is detected, take the original waveforms and the FL curves and create a new COMTRADE file

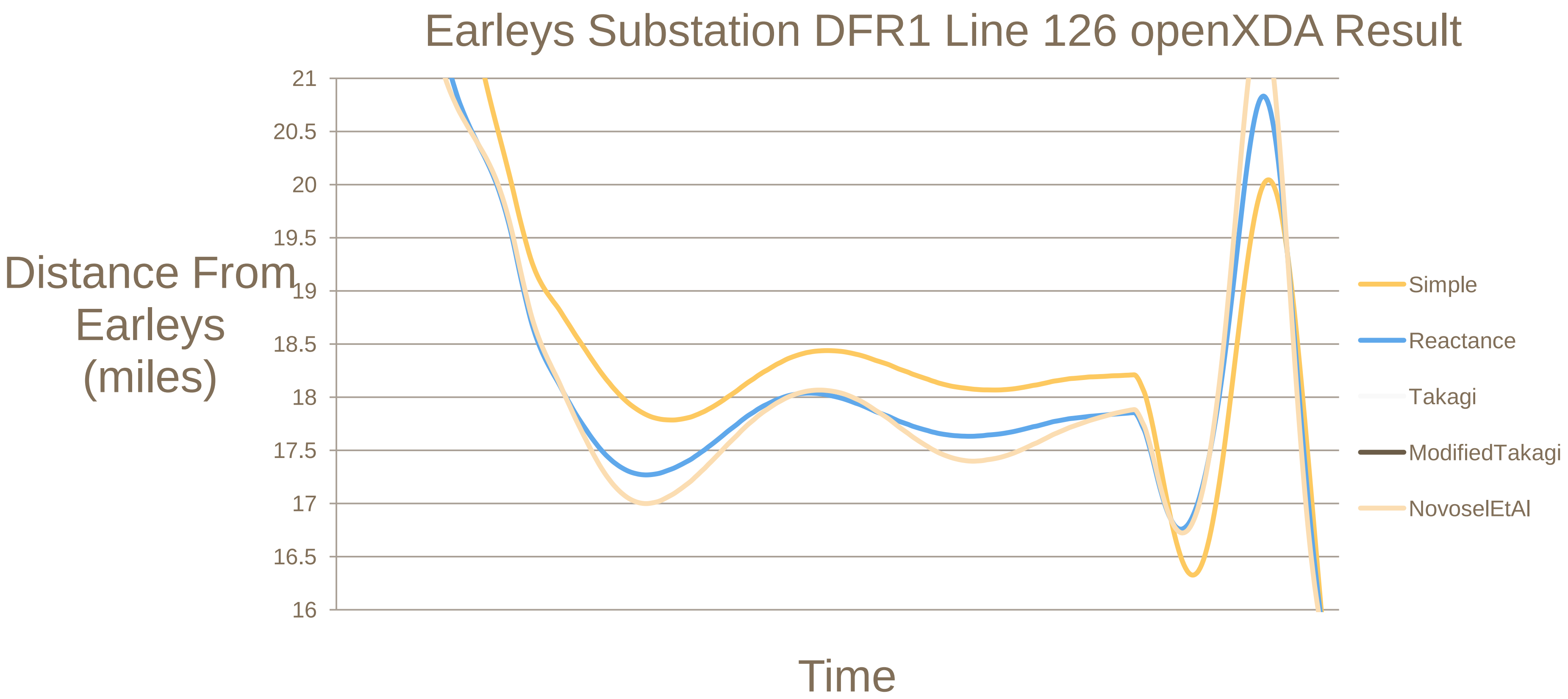
openXDA – Automation

openXDA COMTRADE Results File



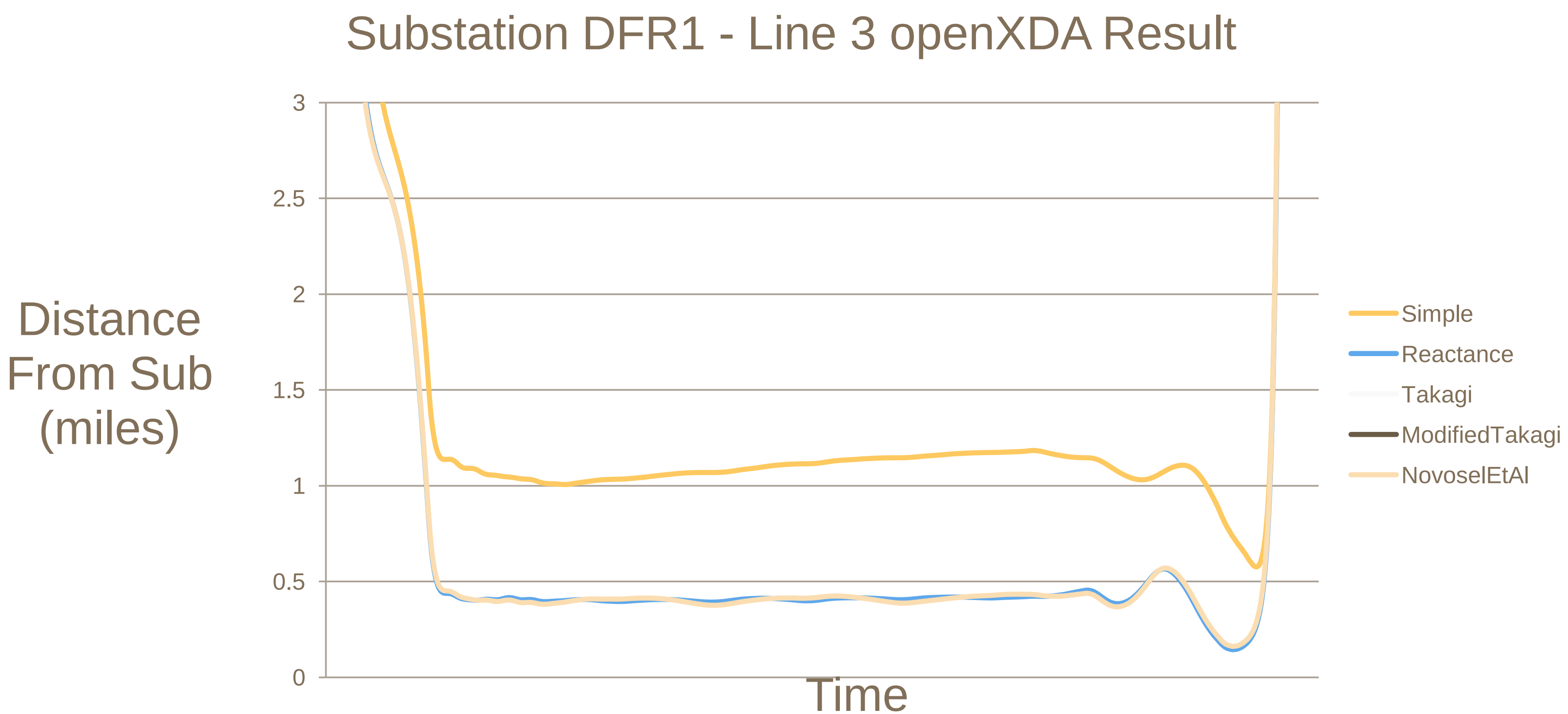
Evaluation Case 1 = Line 126

Fault Type =	Phase C to Ground
Actual fault location =	18.50 miles from Earleys
Traditional FL result =	17.70 miles from Earleys
openXDA FL result =	17.75 miles from Earleys



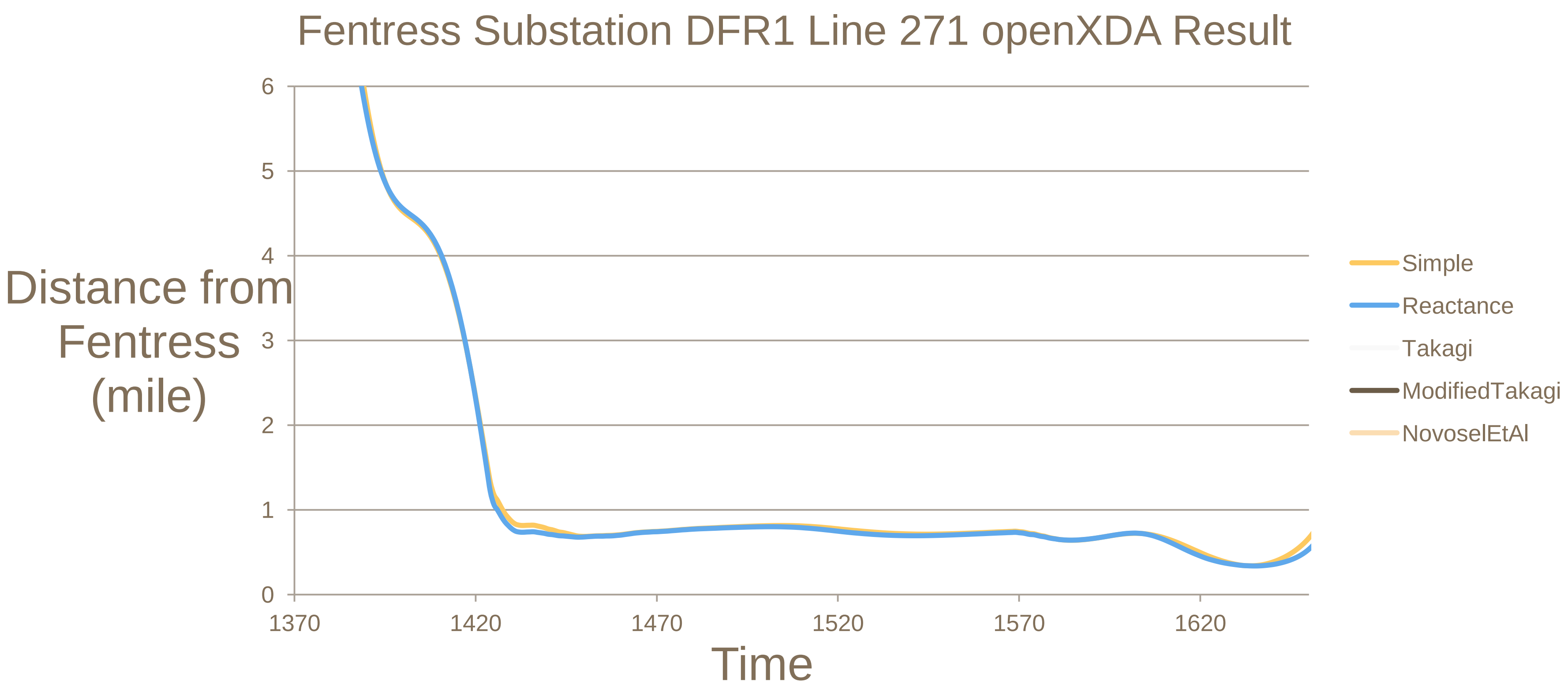
Evaluation Case 2 = Line 247

Fault Type = Phase A to Ground
Actual fault location = 0.58 miles from Suffolk
Traditional FL result= 0.40 miles from Suffolk
openXDA FL result= 0.48 miles from Suffolk



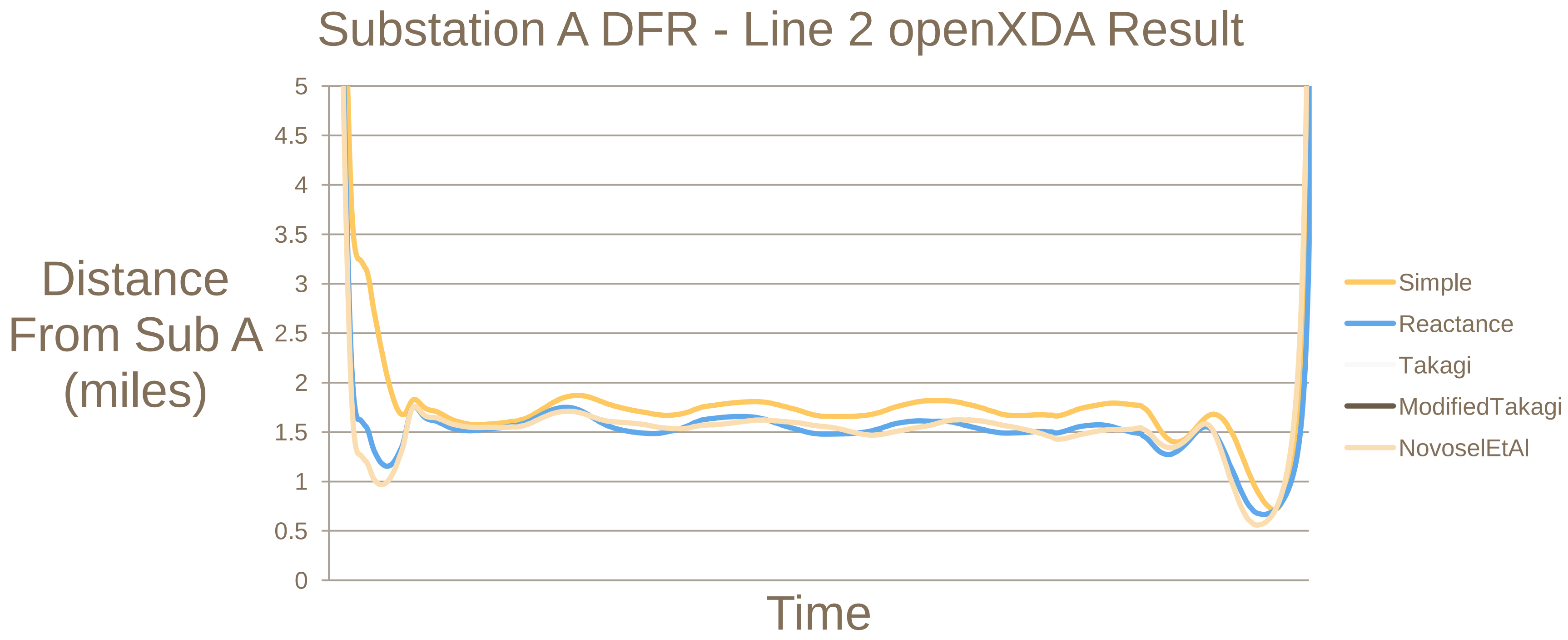
Evaluation Case 3 = Line 271

Fault Type = Phase A to Phase B
Actual fault location = 0.83 miles from Fentress
Traditional FL result= 1.00 miles from Fentress
openXDA FL result = 0.70 miles from Fentress



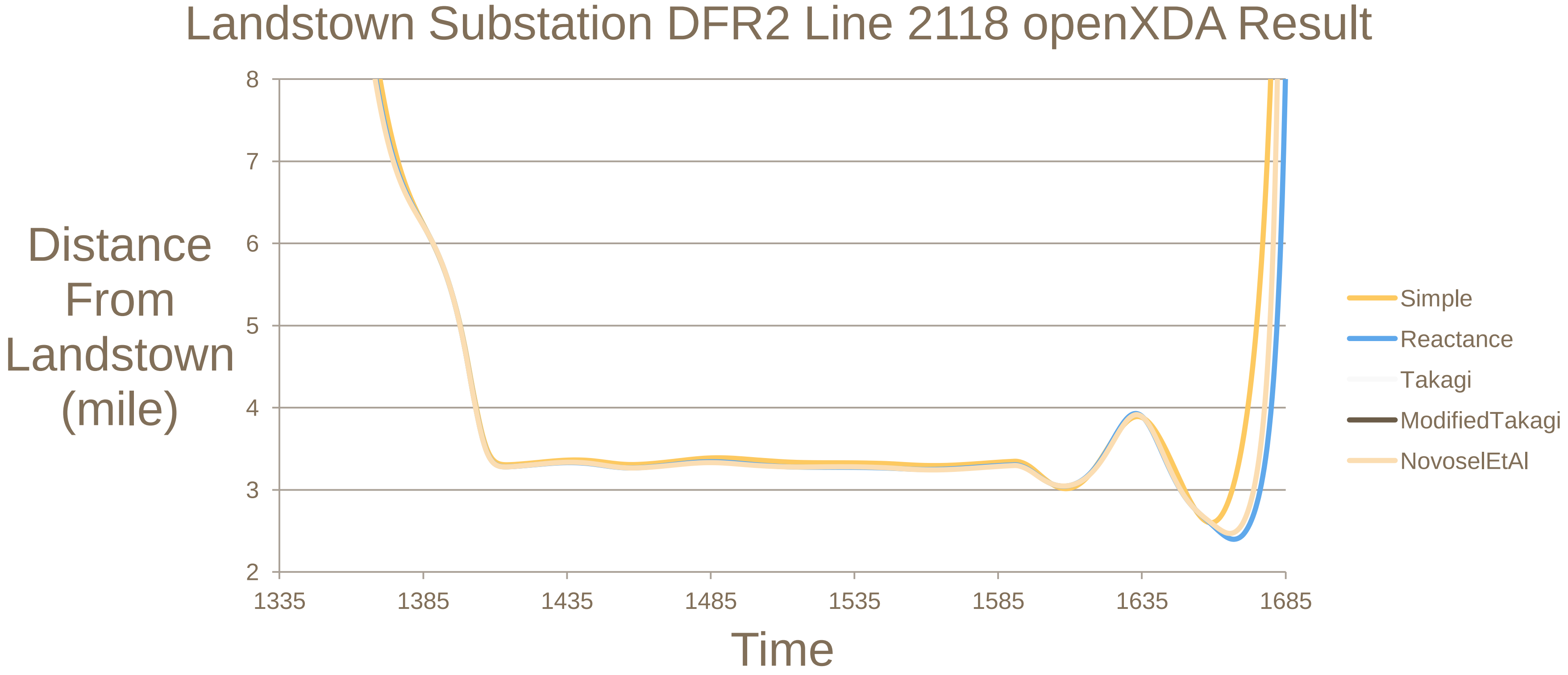
Evaluation Case 4 = Line 2034

Fault Type =	Phase B to Phase C
Actual fault location =	1.39 miles from Trowbridge
Traditional FL result=	1.56 miles from Trowbridge
openXDA FL result=	1.60 miles from Trowbridge



Evaluation Case 5 = Line 2118

Fault Type = Phase C to Ground
Actual fault location = 4.24 miles from Landstown
Traditional FL result= 3.37miles from Landstown
openXDA FL result= 3.41miles from Landstown



openXDA – Future

Extensible is the key

- Double-ended FL Algorithm
- Any new algorithms identified/developed
- Automatically choose only the best algorithms based on OE
- Automatically read DFR config files for line groups
- Create a report with screenshots and statistical analysis
- Extend beyond just FL. Any triggered COMTRADE record could inform you of something significant, such as a pending failure
  - Harmonic analysis
  - Waveform recognition (ex: failing CCVT)
  - Transient analysis
  - Oscillations

Questions?

Kyle Thomas

2014 GPA 4th Annual User's Forum

8/13/2014

So many algorithms…

From

IEEE C37.114

Table 1—Simple impedance equations

Fault type	Positive-sequence impedance equation (mZ_{1L} =)
a–ground	$V_a / (I_a + kI_R)$
b–ground	$V_b / (I_b + kI_R)$
c–ground	$V_c / (I_c + kI_R)$
a–b or a–b–g	V_{ab} / I_{ab}
b–c or b–c–g	V_{bc} / I_{bc}
c–a or c–a–g	V_{ca} / I_{ca}
a–b–c	Any of the following: V_{ab} / I_{ab}, V_{bc} / I_{bc}, V_{ca} / I_{ca}

From

IEEE C37.114

4.3.1 Simple reactance method

$$m = \frac{Im(V_G/I_G)}{Im(Z_L)}$$

For the line-to-ground fault (a-g), the calculation would be as shown in Equation (7):

$$m = Im\left[\frac{V_{Ga}}{I_{Ga} + k_0 I_R)}\right] \Big/ Im(Z_{1L})$$

So many algorithms…

From

IEEE C37.114

Takagi Method

$$m = \frac{Im\left(V_G \Delta I_G^*\right)}{Im\left(Z_L I_G \Delta I_G^*\right)}$$

From

IEEE C37.114

Modified Takagi Method

$$m = \frac{Im\left(V_G\, I_R^*\, e^{-j\beta}\right)}{Im\left(Z_{1L}\, I_G\, I_R^*\, e^{-j\beta}\right)}$$

So many algorithms…

From

IEEE C37.114

Single-Ended Novosel et al Method (from C37.114 Standard)

Equation 2-12

$$\therefore m = \frac{\left(a - \frac{eb}{f}\right) \pm \sqrt{\left(a - \frac{eb}{f}\right)^2 - 4\left(c - \frac{ed}{f}\right)}}{2}$$

The constants in Equation 2-12 are complex multiplications of voltage and current recorded at the substation, transmission line impedance, load, and source impedance. They are defined as follows:

$$\left(\frac{V_G}{Z_{L1} I_G} + \frac{Z_{Load,1}}{Z_{L1}} + 1\right) = a + jb$$

$$\frac{V_G}{Z_{L1} I_G}\left(1 + \frac{Z_{Load,1}}{Z_{L1}}\right) = c + jd$$

$$\frac{\Delta I_G}{Z_{L1} I_G}\left(1 + \frac{Z_{Load,1} + Z_G}{Z_{L1}}\right) = e + jf$$

Solving Equation 2-12 results in two distance estimates of m. The value of m which lies between 0 and 1 pu is the estimated distance to the fault.

So many algorithms…

From

IEEE C37.114

Single-Ended Eriksson et al Method (from C37.114 Standard)

If the source impedances are known, the fault location can be accurately estimated without assumptions. One method discussed in Eriksson, et al. [B2] substitutes Equation (4) in Equation (8). Since the current distribution factor d_S is a function of the source impedances, the line impedance, and the unknown fault location m, a quadratic equation follows:

$$m^2 - mk_1 + k_2 - k_3R_F = 0 \quad (12)$$

where

k_1, k_2, and k_3 are complex functions of local voltage, current, and source impedances

By separating Equation (12) into a real and an imaginary part, one has two equations with two unknowns, m and R_F. The per unit distance to the fault m can be calculated by eliminating R_F and solving for m.

$$V_G = mZ_{1L}I_G + R_F\frac{\Delta I_G}{d_S} \quad (4)$$

$$d_S = \frac{\Delta I_G}{I_F} = \frac{Z_H + (1-m)Z_L}{Z_H + Z_L + Z_G} = |d_S|\angle\beta \quad (8)$$

So many algorithms…

Double-Ended Fault Location Method (from C37.114 Standard & Amir Makki)

$$m = (V_{NS} - V_{FS} + Z_L I_{FS}) / Z_L (I_{NS} + I_{FS})$$

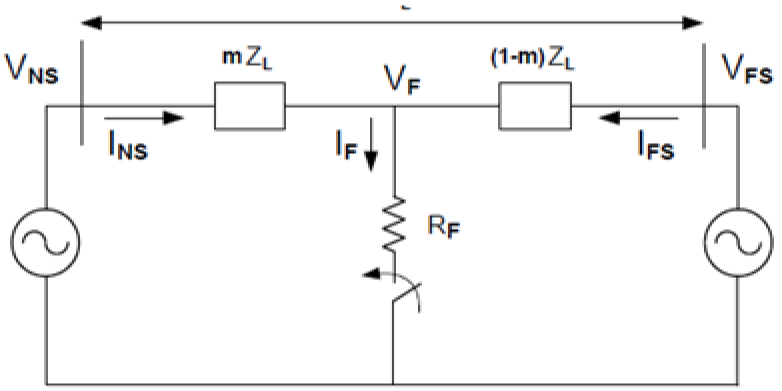


Figure-1; Fault on a Transmission Line with Near and Far Sources