Optimizing the Surge Arresters Location for Improving Lightning Induced Voltage Performance of Distribution Network

Ernesto Perez, member, IEEE
Andres Delgadillo, student member, IEEE,
Diego Urrutia, member, IEEE,
Horacio Torres, senior member, IEEE

NATIONAL UNIVERSITY OF COLOMBIA

June 2007

PhD Thesis
ERNESTO PEREZ GONZALEZ
Presentation Outline

1. Introduction
2. Modelling Lightning Induced Voltage
3. Improving Lightning performance
4. Optimizing Tool
5. Conclusions and Future Work
Introduction - Motivation

- Power quality has become one of the main area of interest around the world for mains, utilities, industries and consumers.
- Lightning causes around 50% of the network electromagnetic disturbances (short interruptions and voltage sags)
Introduction - Motivation

- Millions of USD for losses caused by lightning in distribution network
- Is important to look forward techniques for reducing lightning impact on distribution networks.
- Shielding wire groundings, surge arresters and enhancement of Line BIL are a very useful technique
Lightning Induced Voltage Modelling

By means of accurate calculation of lightning-induced overvoltages on real distribution networks

→

Optimized design of distribution networks against external overvoltages

→

Correct definition of number and location of protective devices (shielding wire groundings and surge arresters)
Link with Transient Analysis Software

- LIOV-EMTP/MATLAB (Methodology A)
  EMTP96 – using TACS on a DLL and MATLAB – using S-function

- YALUK-ATP (Methodology A)
  ATP – Using Foreign Models on a DLL

- LIV-ATP (Methodology B)
  ATP programmed in MODELS
Diminishing Number of Failures

- Surge Arrester (SA) helps to Diminish the fault rate due to induced Voltages
- The best solution for a straight line is locate a SA Every Pole
Diminishing Number of Failures

- Which is the best SA location?
  - If the number of SA is fixed and limited?
  - If it is used a Complex distribution Network with non-homogeneous pole distribution?
Software Tool – Structure

- It was developed an Optimizing Software Tool for SA location.
- Object function is to diminish number of failures for a fixed number of SA.
- Based on Genetic Algorithm (GA) Technique.
- Each ‘individual’ (possible solution) is characterized with a unique SA location.
- For each individual, it should be calculated the lightning-induced voltage performance of the line.
- It is used a certain number of strokes for this task.
What is Genetic Algorithm?

Initial individual generation

C₁ C₂ C₃ C₄ C₅ C₆
What is Genetic Algorithm?

Individuals Mating

C1 C6
C2 C4
C5 C3

Crossing

C7 C8 C9
What is Genetic Algorithm?

Mating

C1  C6
  ↓   ↓
C7  C8  C9

Crossing

Evaluation of Objective Function

Comparison
F(C1)  F(C2)  F(C7)
Comparison
F(C3)  F(C4)  F(C8)
Comparison
F(C5)  F(C6)  F(C9)

New Generation

C7  C1  C2  C4  C5  C9
What is Genetic Algorithm?

Initial generation

- $C_1$
- $C_6$
- $C_2$
- $C_4$
- $C_5$
- $C_3$

Crossing

- $C_7$
- $C_8$
- $C_9$

Evaluation of Objective Function

- Comparison: $F(C_1)$, $F(C_2)$, $F(C_7)$
- Comparison: $F(C_3)$, $F(C_4)$, $F(C_8)$
- Comparison: $F(C_5)$, $F(C_6)$, $F(C_9)$

New Generation

- $C_7$
- $C_2$
- $C_3$
- $C_4$
- $C_5$
- $C_9$
Genetic Algorithm Scheme

Random generation of first individuals

Population

Assigning Probability

Selecting ‘parents’

Crossover and Mutation

Comparison between ‘parents’ and ‘sons’

Evaluating Objective Function

Modifying Data cases

Execution YALUK-ATP

Output Files

Replacement with new Generation

Fitness Values

Reading Data

Chromosomes Population

Chromosomes new individuals

Fitness Values

Reading Data

Execution YALUK-ATP

Output Files

Genetic Algorithm Scheme

Random generation of first individuals

Population

Assigning Probability

Selecting ‘parents’

Crossover and Mutation

Comparison between ‘parents’ and ‘sons’

Evaluating Objective Function

Modifying Data cases

Execution YALUK-ATP

Reading Data

Output Files

Fitness Values

Replacement with new Generation

Execution YALUK-ATP

Reading Data

Output Files

Fitness Values

Chromosomes Population

Chromosomes new individuals
Genetic Algorithm Scheme

- Random generation of first individuals
- Population
- Assigning Probability
- Selecting ‘parents’
- Crossover and Mutation
- Comparison between ‘parents’ and ‘sons’
- Evaluating Objective Function
  - Executing YALUK-ATP
  - Output Files

- Evaluating Objective Function
  - Executing YALUK-ATP
  - Output Files

- Replacement with new generation

Read Data
Fitness Values
Reading Data
Genetic Algorithm Scheme

- Random generation of first individuals
- Population
  - Assigning Probability
    - Selecting ‘parents’
      - Crossover and Mutation
        - Comparison between ‘parents’ and ‘sons’
  - Evaluating Objective Function
    - Chromosomes Population
      - Modifying Data cases
        - Execution YALUK-ATP
          - Reading Data
            - Output Files
        - Fitness Values
          - Modifying Data cases
            - Execution YALUK-ATP
              - Reading Data
                - Output Files
              - Fitness Values
            - Chromosomes new individuals
              - Replacement with new Generation
### Genetic Algorithm Scheme

**Random generation of first individuals**
- Population
  - Assigning Probability
    - Selecting ‘parents’
      - Crossover and Mutation
        - Comparison between ‘parents’ and ‘sons’

**Evaluating Objective Function**
- Chromosomes Population
  - Fitness Values
    - Modifying Data cases
      - Execution YALUK-ATP
        - Output Files

- Reading Data
  - Output Files

- Replacing with new generation

---

**IEEE – PES General Meeting –**
**Tampa June 25th 2007**
Genetic Algorithm Scheme

Random generation of first individuals

Population

Assigning Probability

Selecting ‘parents’

Crossover and Mutation

Comparison between ‘parents’ and ‘sons’

Evaluating Objective Function

Chromosomes Population

Modifying Data cases

Execution YALUK-ATP

Output Files

Fitness Values

Reading Data

Evaluation Objective Function

Chromosomes new individuals

Modifying Data cases

Execution YALUK-ATP

Output Files

Fitness Values

Reading Data
Genetic Algorithm Scheme

1. Random generation of first individuals
   - Population

2. Evaluating Objective Function
   - Execution YALUK-ATP
   - Output Files

3. Assigning Probability
   - Reading Data
   - Fitness Values

4. Selecting ‘parents’
   - Modifying Data cases
   - Chromosomes Population

5. Crossover and Mutation
   - Chromosomes new individuals
   - Fitness Values

6. Comparison between ‘parents’ and ‘sons’
   - Modifying Data cases
   - Execution YALUK-ATP
   - Output Files

7. Replacement with new Generation
Genetic Algorithm Scheme

1. Random generation of first individuals
2. Population
3. Assigning Probability
4. Selecting ‘parents’
5. Crossover and Mutation
6. Comparison between ‘parents’ and ‘sons’
7. Evaluating Objective Function
   - Chromosomes Population
   - Modifying Data cases
   - Execution YALUK-ATP
   - Output Files
8. Evaluating Objective Function
   - Chromosomes new individuals
   - Modifying Data cases
   - Execution YALUK-ATP
   - Output Files

Replacement with new generation
Genetic Algorithm Scheme

- Random generation of first individuals
- Population
  - Assigning Probability
  - Selecting ‘parents’
  - Crossover and Mutation
  - Comparison between ‘parents’ and ‘sons’
- Evaluating Objective Function
  - Chromosomes Population
  - Evaluating Objective Function
  - Modifying Data cases
  - Execution YALUK-ATP
  - Output Files
  - Reproduction with new Generation
**Genetic Algorithm Scheme**

1. **Random generation of first individuals**
   - Population

2. **Assigning Probability**
   - Selecting ‘parents’
   - Crossover and Mutation

3. **Evaluating Objective Function**
   - Execution YALUK-ATP
   - Output Files

4. **Comparison between ‘parents’ and ‘sons’**
   - Fitness Values

5. **Replacement with new Generation**
   - Reading Data
   - Modifying Data cases

6. **Evaluating Objective Function**
   - Execution YALUK-ATP
   - Output Files

- **Population**
- **Chromosomes Population**
- **Fitness Values**
- **Reading Data**
- **Modifying Data cases**
Example of Genetic Algorithm Tool

- 20 Nodes Network (three phase)
- Location of 4 three phase Surge Arresters
- 4845 possible SA locations
- Lightning performance calculated with
  - 40 Strokes
  - 100 Strokes
- It is chosen a base case with a SA located randomly
Effect of Power System Components

- Part of Real Network with a main feeder and three branches with different length.
- Total length 7km aprox.
- Random case diminish the number of outages for 200kV from 35 to 20
Engineering Application

- Random case diminish the number of outages for 200kV from 35 to 20
- Running Genetic Algorithm Tool the number of failures for 200kV is 10
Conclusions

- Here is described a new methodology based on genetic algorithms intended to find an optimal solution for the location of a set of surge arresters.

- This methodology could bring better results when a reasonably probability curve is possible to be obtained for each individual.

- Greater number of strokes should be used for each solution, implying that big efforts should be done in order to reduce induced voltage computation time.
Conclusions

- **This tool allows to find a “good” solution but not always this is the best one.**

- **This proposed methodology contributes on the researching focused on the using of artificial intelligence techniques, such as, genetic algorithms for designing and planning the distribution network systems optimally.**

- **Further work should be done in increasing the number of parameters to simulate meanwhile it is improve the computation time.**
THANK YOU

FOR YOUR ATTENTION