Abstract - In this paper, the parallelization of the Fault Position Method for the stochastic assessment of voltage sags is presented. The parallelization of this stochastic method is made by using multi-thread programming in an algorithm written in C programming language, including functions defined by the POSIX threads header and library in order to perform the parallel calculations desired. All tests were performed using the GNU/Linux operating system. The proposed parallelized method is applied to two electrical systems: a small 5-bus test system and the IEEE 57-bus test system, in order to demonstrate its proper operation. A comparative analysis of the parallelized Fault Position Method with respect to the traditional method is presented, and the reduction in computational time is shown.

Keywords: Voltage sags; fault position method; parallel processing; multi-threading

1 Introduction

A voltage sag is defined as a decrease in RMS voltage at the power frequency for durations from 0.5 cycles to 1 minute, usually reported as the remaining voltage [1]. Short circuits are the main cause of their appearance in electrical systems, but they can also be caused by the starting of large motors, overloads or the disconnection of capacitor banks [1][2].

A voltage sag is a phenomenon of power quality that significantly affects energy users, especially in the industrial sector, where considerable economic losses are originated by this disturbance [2][3]. Furthermore, within the scope of smart grids, an important goal is to design electrical systems that operate in a more automated, secure and efficient way, providing power quality in the range of need [4][5].

In recent decades, several methods have been proposed for evaluating a power system in terms of voltage sags [6][10]. One of the most widely used methods is the well-known Fault Position Method (FPM) [7][11], which allows to estimate the occurrence of sags based on statistical data of faults in the elements and parameters of the electrical network under study.

When applying the Fault Position Method, a large number of fault positions must be considered in order to obtain accurate results [10][12], however, increasing the number of faults will require greater computational resources. This is an important aspect to consider when large power grids are being analyzed. Therefore, it is of interest to employ computational techniques, such as parallel programming, that allow to apply this method in a more efficient way.

In modern power system analysis, it has become necessary to use diverse computational tools such as parallel processing or object-oriented programming, amongst others. Parallel processing is defined as a method of data processing in which two or more processing elements perform calculations to solve a problem working simultaneously together. Parallel processing has been applied in the solution of several problems in the field of power system analysis and industrial applications. For example, in [13], an algorithm for fault detection in DC motor drivers using parallel programming is presented; in [14], parallel processing is employed in the study of electromagnetic transients; in [15], a method for the harmonic power flow analysis is developed; in [16], this technique is applied to the fast steady-state solution of power systems. Recently, fine-grained parallel processing techniques have been applied to the fast steady-state solution of large-scale electric power systems [17]. In previous works, parallel processing has allowed to reduce the computational time required to perform a study, which, in consequence, makes it possible to perform power system analyses in less time and improve the system operation, thus reducing economic losses associated with electrical faults.

In this paper, the Fault Position Method for the stochastic assessment of voltage sags in electrical power systems is implemented employing parallel processing techniques based on multi-thread programming. The proposed method is applied to IEEE test systems in order to demonstrate its performance.

2 Parallelized Fault Position Method

In order to illustrate the procedure to parallelize the Fault Position Method, a brief description of the method is presented first, and then the steps for its parallelization are exposed.
2.1 Fault Position Method Basis

The Fault Position Method is based on the classic short circuit calculation, combined with data of fault probability in nodes and lines, in order to estimate the occurrence of sags in a given period of time. The procedure followed to apply this method is described next [7][11].

Consider the generic system with $n$ buses shown in Fig. 1. When a fault occurs at a generic bus $i$ of the electrical system, according to the classic short circuit calculation, the impedance matrix of the system, $Z_{bus}$, is of order $(n \times n)$, and the voltage at bus $m$ can be calculated by [18]

$$V_m = V_m^{pf} - Z_m I_i$$ (1)

where:

- $V_m^{pf}$ is the pre-fault voltage at bus $m$;
- $Z_m$ is the transfer bus impedance between busbar $m$ of the system and the faulted bus $i$;
- $I_i$ is the fault-current phasor at bus $i$ that can be calculated as

$$I_i = \frac{V_i^{pf}}{Z_i}$$ (2)

where:

- $V_i^{pf}$ is the pre-fault voltage at bus $i$;
- $Z_i$ is the transfer bus impedance of the faulted bus $i$.

Then,

$$V_m = V_m^{pf} - \frac{Z_m}{Z_{ii}} V_i^{pf}$$ (3)

For the calculation of voltage magnitudes due to line faults, fictitious buses along the lines must be considered. For example, in the generic system shown in Fig. 1, one fault position in the middle of the line connecting buses $k$ and $j$ is considered. Then, according to the classic short circuit calculation, the voltage at bus $k$ can be calculated by

$$V_k = V_k^{pf} - \frac{Z_{kp}^*}{Z_{pp}^*} V_p^{pf}$$ (4)

where:

- $Z_{kp}^*$ is the modified transfer bus impedance between busbar $k$ and the faulted bus $p$;
- $Z_{pp}^*$ is the modified transfer bus impedance of the faulted bus $p$.

It is important to notice that in equation (4) the order of the correspondent bus impedance matrix, $Z_{bus}^*$, is $((n+1) \times (n+1))$, due to the addition of the fictitious bus $p$.

Then, considering that the fault at the generic position $p$ (real or fictitious bus) has an associated value of fault rate, $\lambda_p$ (usually faults/year), the accumulated frequency of voltage sags for a magnitude threshold between $V_{low}$ and $V_{up}$, at a generic bus $k$, can be calculated by [2][7]

$$P_k = \sum \lambda_p : V_{low} < V_p \leq V_{up}$$ (5)

where:

- $P_k$ is the voltage sags probability at bus $k$ (voltages sag/year)

2.2 Parallelization of the Method

From the previous description of the Fault Position Method, it is clear that in order to achieve an acceptable level of precision, a large number of fault positions must be included in the calculations. Furthermore, if a fictitious bus is added in each calculation, the dimension of the problem grows significantly when several fault positions in the electrical system are simultaneously analyzed.

For example, for the IEEE 57-bus test system that consists of 63 lines [19], if 10 fault positions are included in the lines, 687 fault calculations are required, with a $(57 \times 57)$ order impedance matrix, or a new $(687 \times 687)$ order impedance matrix must be created, which would include all the fault positions of the system (assuming fictitious buses). Obviously, when large systems are being analyzed, the computational requirements become a more relevant aspect, and it results interesting to apply techniques such as parallel processing in order to perform calculations more efficiently.
In the designed algorithm for the parallelization of the Fault Position Method, each processing element divides the line in a specified number of parts. Then, a fault position is set in each one of these parts and the bus admittance matrix is recalculated for every part. As it was previously mentioned, the dimension of the admittance matrix for each calculation is \((n+1)(n+1)\), due to the addition of the fictitious bus that represents the fault position. Once the bus admittance matrices are obtained, they must be inverted in order to find the impedance matrices and calculate the voltage magnitudes at buses for each assumed fault. Then, based upon these values, the probability of voltage sags occurrence in each bus will be obtained according to specific voltage ranges. To perform this task, parallel processing based on multi-threading was applied.

In multi-threading [20], multiple control threads can solve a large portioned problem; lightweight sub-processes executed within a process share code and data segments, but with their own program counter, machine registers and stack. Global and static variables are common to all threads.

The efficiency of the parallel algorithm is measured in terms of the time it takes to complete the calculations with one processing element in comparison to the time it takes to complete the calculations with \(p\) processing elements; this relation is known as Speed-Up \(S\) [21].

\[
S = \frac{T_1}{T_p} \tag{6}
\]

where:

\(T_1\) is the execution time with one processing element;

\(T_p\) is the execution time with \(p\) processing elements.

Using this metric of performance ensures that the reduction of execution time is independent of the computer characteristics.

In this work, the parallelization algorithm was written in C programming language, including functions defined by the POSIX threads header and library in order to perform the parallel calculations desired. All tests were performed using the GNU/Linux operating system, in particular the Xubuntu 12.10 distribution, and using an Intel (R) Xeon (R) CPU ES40S with two quad-core processors running at 2.0 GHz.

### 3 Case Studies

Two case studies are presented, which allow to verify the operation and efficiency of the method implemented using multi-thread programming.

#### 3.1 Studies in a small test system

In order to analyze in detail the performance of the proposed method, a 5-bus test network (Fig. 2) is adopted. Impedances of the network elements are indicated in Fig. 2 in per-unit values. In this example, uniform distribution of faults along the transmission lines has been assumed. Balanced three-phase faults have been considered with the following statistical rates of faults/year for the different lines: line 2–4, \(\lambda=16\); line 2–5, \(\lambda=8\); and line 4–5, \(\lambda=4\).

In order to demonstrate the performance of the proposed parallelized Fault Position Method, the results are compared to the results obtained from the application of an analytical method [10], in which the accuracy does not depend on discrete values, as is the case of the Fault Position Method.

In Table 1, the voltage sags per year calculated using the parallelized Fault Position Method, considering different number of fault positions, are presented. It can be clearly seen that as the number of fault positions considered increases, the obtained result is closer to the analytical solution. The results obtained using a small number of fault positions are significantly different to the analytical solution, however, when a large number of fault positions is considered, there is no appreciable difference between the results yielded by the Fault Position Method and the analytical method. A similar behavior would be observed when this calculation is made for the rest of the buses.

In Table 2, the speedup results using the proposed technique for different number of fault positions are shown. The efficiency of the parallel algorithm is more evident when the number of fault positions increases. The maximum speedup factor in this case is 2.984 (columns S in Table 2), which means that the calculations are performed almost three times faster than they would be if a sequential algorithm was used.

![Fig. 2. Small test system](image)

<table>
<thead>
<tr>
<th>Voltage sag range</th>
<th>Analytical method</th>
<th>Fault position method</th>
<th>Fault positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0&lt;V&lt;0.1</td>
<td>4.74</td>
<td>0.0</td>
<td>4.64 4.64 4.74 4.74</td>
</tr>
<tr>
<td>0.1&lt;V&lt;0.2</td>
<td>10.37</td>
<td>8.0</td>
<td>10.24 10.48 10.45 10.46</td>
</tr>
<tr>
<td>0.2&lt;V&lt;0.3</td>
<td>7.12</td>
<td>4.0</td>
<td>7.04 7.12 7.12 7.12</td>
</tr>
<tr>
<td>0.3&lt;V&lt;0.4</td>
<td>5.77</td>
<td>16.0</td>
<td>6.08 5.76 5.77 5.77</td>
</tr>
</tbody>
</table>


### Table 2. Speedup of the parallel solution of the 5-bus test case using threads and different number of fault positions

<table>
<thead>
<tr>
<th>Threads</th>
<th>100 faults</th>
<th>1,000 faults</th>
<th>10,000 faults</th>
<th>100,000 faults</th>
<th>1,000,000 faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>2</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>3</td>
<td>1.588</td>
<td>1.404</td>
<td>1.499</td>
<td>1.493</td>
<td>1.506</td>
</tr>
<tr>
<td>4</td>
<td>1.500</td>
<td>2.403</td>
<td>2.870</td>
<td>2.938</td>
<td>2.984</td>
</tr>
</tbody>
</table>

#### 3.2 Studies in the IEEE-57 buses test system

The IEEE 57-bus test system consists of 57 buses (see Fig. 3) which are interconnected by means of 63 lines, 15 transformers and 7 generating units [15]. Balanced three-phase faults have been considered and a fault rate of 1.0 for all lines; fault rates at buses have been neglected.

Fig. 4 shows the results of voltage sags for the voltage range from 0.5 to 0.6 p.u. when different number of fault positions at lines are considered. In order to visualize the differences more clearly, graphs are shown for 1, 10 and 100 fault positions. It is observed that as one increases the number of fault positions, results tend to values that would be obtained without the discretization of fault positions. Similarly, Fig. 5 shows the results for the voltage sags range of 0.6 to 0.7, and once again it can be observed that the results with 10 positions are closer to the obtained with 100 positions than the obtained with one fault position.

In Fig. 6 and Fig. 7, the number of voltage sags/year for a voltage range up to 0.7 p.u. and 0.8 p.u. are shown, respectively. It can be seen that the differences are smaller for the more widely voltage sags range.

In Table 3, the execution times considering different number of fault positions for the 57-bus system are shown. It can be seen that when the number of threads used to perform parallel calculations is increased, a significant reduction in execution time is achieved. The speedup obtained with 8 threads and 40 faults indicates that the problem is solved approximately 7.8 times faster than when a single processing element is used.
4 Conclusions

In this paper, parallel processing techniques have been applied to the parallelization of the Fault Position Method for the stochastic assessment of voltage sags. The applied parallelization is based on multi-thread programming, which allows to divide the task of short circuit calculations for the assumed faults in the electrical system.

The proposed parallelized method was applied to a small 5-bus test system and to the IEEE 57-bus test system. In the case studies performed, it is demonstrated that the proposed parallelized method can be a helpful tool to estimate the voltage sag performance of the system in a more efficient manner than the traditional Fault Position Method.

5 References


