MDPI

Article

A fast decomposition for solving a Security-Constrained Optimal Power Flow (SCOPF) problem through constraint handling

Tomas Valencia[1], Daniel Agudelo-Martinez [1,2], Dario Arango [1,2], Camilo Acosta[1], Sergio Rivera[2], Diego Rodriguez [1,2] and Juan Gers[1]

[1] GERS USA, Weston, FL 33331, USA
[2] Electrical Engineering Department, Universidad Nacional de Colombia, Bogotá 110111, Colombia
* Correspondence: diego.rodriguez@gers.com.co; Tel.: +057 4857100 (F.L.)

Version January 16, 2020 submitted to Energies

Abstract: This paper presents a decomposition methodology using constraint handling to improve the computation time of a security-constrained optimal power flow (SCOPF) problem. In order to evaluate methodology performance, tests over small (500 buses), medium (4,918 buses), and large scale (11,615 buses) transmission networks were carried out. The methodology consists of the decomposition of the SCOPF problem into a base case problem and contingency sub-problems, using constraint handling rules to solve the complete problem in an iterative fashion. The first stage involves computing an OPF problem using a base case network. The second stage deals with the modification of the initial base case by updating some of the constraint limits according to the evaluation of potentially relevant contingencies. The entire algorithm resorted to parallel computing tools. The methodology successfully solved the tested networks with the set of proposed constraints, including active and reactive power re-dispatch in post-contingency scenarios.

Keywords: Security-constrained optimal power flow (SCOPF); real-time optimal power flow (OPF); interior point method; parallel processing; Constraint handling; Matpower; Interior Point Optimizer (IPOPT)

1. Introduction

The security-constrained optimal power flow (SCOPF) problem deals with finding an optimal operating cost of power systems while ensuring security criteria (usually N-1) from a plausible set of contingencies [1–5]. SCOPF is a non-linear, non-convex, static, large-scale optimization problem that might also have integer decision variables (generally known as a Mixed Integer Nonlinear Programming -MINLP- optimization problem) [3,4,6,7].

The importance of a fast and optimal (or near-optimal) solution to the OPF and SCOPF problems is remarked by organizations such as the North American Electric Reliability Corporation (NERC) and the Federal Energy Regulatory Commission (FERC) [6,8,9]. The latter stated that a 5% increase in the efficiency of the algorithms for OPF will yield six billion dollars in savings per year in the United States alone [9]. However, in addition to the complexity of the SCOPF optimization problem, computational burden hugely increases when large-scale networks and many contingencies are evaluated [10,11], becoming an NP-Hard problem in the worst case [12]. For these reasons, the optimal global condition for the SCOPF problem cannot be ensured in its generic formulation with constrained time [2].

The security-constraint OPF problem is addressed through the preventive (P-SCOPF) and the corrective (C-SCOPF) approaches, or a combination of both (PC-SCOPF). The preventive and corrective models aim to find a minimum cost of operation that is also feasible for all the considered contingencies

[5,7]. P-SCOPF considers no corrective actions in post-contingency states other than those with an automatic response to contingencies (e.g. active power of generators participating in frequency control, automatic tap-changers, reactor banks switching, secondary voltage control) [3,11]. Its major drawbacks are the resulting high operating cost due to the over-tightened feasible region (conservative solution) and a high computation time as a consequence of a large amount of contingencies [5,7,11,13, 14].

In contrast, C-SCOPF allows the system operator to re-adjust control variables after a contingency actually occurs in order to eliminate any violations caused by the evaluated contingencies [5,7]. This is based on the fact that some power system components (i.e. transmission lines, transformers) would not be affected by short period violations [5,7]. However, most of the proposed formulations for the C-SCOPF problem have not considered the reactive power re-dispatch, the lack of modeling the possible failure of corrective controls, and the cost of the overall corrective actions [13–15].

Implementation examples for P-SCOPF can be found in [10,16], for C-SCOPF in [2,7,10], and for PC-SCOPF in [13,14]. Other SCOPF models include risk assessment [1,17,18], time constraints [19], and stochastic models [13,15]. However, many of these models must tackle computational limitations (e.g. computation time and memory) when solving the SCOPF problem considering medium- and large-scale networks as well as a large number of contingencies [7,15].

A set of strategies have been used to make the SCOPF optimization problem more tractable from a computational point of view. Some of these techniques, used in both OPF and SCOPF problems, use *linearization* and *convexification* of the optimization problem. One kind of *linearization* is solving the SCOPF problem through the DC-OPF approximation [8,16,17,20]. However, the linear approximation may be inaccurate when using reactive power control variables (shunt reactance, voltage at generator buses) or under highly loaded conditions [3]. Other linearization techniques act directly on the objective function [17], and others include the Successive Linear Programming (SLP) method [2].

Both the OPF and SCOPF are not convex problems. Therefore, it is not possible to ensure a global minimum through mathematical programming [21]. Different attempts have been made to reach solutions close to the global minimum through genetic (GA), metaheuristic (MA) and machine learning algorithms. Strategies such as the earthworm optimization algorithm, firefly algorithm tuned through fuzzy logic and approaches based on historical data have been presented in [22–24] to solve OPF. However, the size of the networks analyzed exceed no more than 300 buses and the number of iterations to reach the solution through these strategies cannot be ensured.

Decomposition strategies have been proposed that allow large problems to be divided into subproblems to solve them with parallel computing. The most commonly used algorithms include the Augmented Lagrangian Method (ALM), Alternating Direction Multipliers Method (ADMM), and Benders Decomposition (BD).

ALM in [25] was used to solve a distributed OPF, while in [26] it was used to solve the reactive OPF from network splitting. However, its application to the SCOPF has not been implemented yet.

ADMM has been widely used because it allows the total problem to be divided and makes it parallelizable and easy to implement. This method was used in [17,25,27] to solve an OPF. The ADMM was also implemented in [5] to solve a C-SCOPF by testing a set of networks of up to 3,012 buses. However, the number of contingencies and the time to reach a solution was not promising for a real time requirement (3,582 seconds for 4 contingencies).

BD was presented in [10] as an appropriate methodology to divide the C-SCOPF problem, but it was applied only for a 6-bus network. In [1,6], a 118-bus and 2,351-bus systems were validated, but using a DC model of power flow equations. In [5] BD was also used in a 3,012-bus network but it considered only 4 contingencies and took 1,165 seconds to reach the solution.

On the other hand, several works have tried to solve non-convex problems through a hybrid optimization strategy. For example, the OPF problem in [21] was solved by using GA to group the chromosomes in a search space close to the absolute minimum, and then a continuous Newton-Rhapson method was used to mathematically reach the global minimum. However, overload constraints were

not considered. In [7], ALM and ADMM were used to solve the SCOPF in DC. In [13], BD was used to solve PC-SCOPF along with an evolutionary algorithm (EA) to select the relevant contingencies; however, only a 118-bus system was tested.

Another strategy widely used to reduce the size of SCOPF is contingency filtering [3,4]. For example, in [28] the umbrella contingencies method was used to select the most relevant contingencies taking into account the magnitude of the lagrangian multipliers associated with the post-contingency balance constraints. The disadvantage of this approach is that it is necessary to solve the SCOPF first, which makes it infeasible in real time. In [29], a method for contingency evaluation in real time was proposed using weighted digraphs and the central eigenvector of the Laplacian matrix. However, the Laplacian matrix was filled based on the number of overloads caused by the outaged lines, which took too long for a real time approach. EA was used in [13] for contingencies filtering, but it demands a full iteration to identify insecure contingencies. In [2], vulnerability and critical measures were used to select the relevant contingencies in the problem. However, it is necessary to evaluate a power flow for each contingency to carry out this selection, so it makes it infeasible in real time.

According to the state of the art, the main contributions of this paper can be summarized as follows:

- Consideration of novel constraints with respect to common formulations of SCOPF, since reactive power re-dispatch and area spin reserve are considered.
- A contingency filtering algorithm is proposed according to the base case OPF to reduce the problem size.
- Large and complex networks as well as numerous contingencies are analyzed to increase the success of the tool in real systems.
- The methodology solves a power flow problem in each contingency, therefore, it might be faster than other decomposition approaches that demand solving an optimization problem.
- A novel algorithm to active and reactive power re-dispatch is employed to compute a post contingency state.
- An algorithm is proposed to iteratively solve the SCOPF by modifying the constraint limits of the base case.

The remainder of this paper is organized as follows. Section 2 presents the complete formulation of the SCOPF problem and explains the proposed approach to address it. The methodology and flow diagrams of the implemented algorithms are described in Section 3. Sections 4 and 5 describe the results obtained and the corresponding discussion. Finally, Section 6 presents the conclusions and future works from this research.

2. Formulation

2.1. Problem formulation

The SCOPF problem [30] is focused on minimizing the total cost C_{tot}:

$$min(C_{tot}) = min\left[\left(\sum_{g\epsilon G} c_g\right) + \delta c^{\sigma} + \frac{1-\delta}{|k|}\sum_{k\epsilon K} c_k^{\sigma}\right] \tag{1}$$

where G is the set of generators, c_g is the generation cost of generator g, c^{σ} is the total constraint violation penalty in base case and c_k^{σ} is the total constraint violation penalty in contingency k. K is the set of all contingencies. δ is a weight assigned to the penalty cost in the base case.

For all constraints, sc denote a particular scenario of the set SC:

$$SC = \{0, 1, 2, 3, ..., k-1, k\} \tag{2}$$

where $sc = 0$ means the base case and $sc = i$ with $i > 0$ denote the i^{th} scenario, which corresponds when the i^{th} contingency is applied.

The constraints associated to objective function are:

$$\underline{v_i} \leq v_i \leq \overline{v_i} \quad \forall i \, \epsilon \, I^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{3}$$

where I^{sc} is the set of active buses in the scenario sc. v_i is the voltage magnitude on bus i.

$$\underline{p_g} \leq p_g \leq \overline{p_g} \quad \forall g \, \epsilon \, G^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{4}$$

$$\underline{q_g} \leq q_g \leq \overline{q_g} \quad \forall g \, \epsilon \, G^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{5}$$

where G^{sc} is the set of active generators in the scenario sc. p_g and q_g are the active and reactive power of generator g.

The considered transmission lines overload constraints were:

$$\sqrt{(p_e^o)^2 + (q_e^o)^2} \leq \overline{R}_e v_{i_e}^o + \sigma_e^{sc,s} \quad \forall e \, \epsilon \, E^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{6}$$

$$\sqrt{(p_e^d)^2 + (q_e^d)^2} \leq \overline{R}_e v_{i_e}^d + \sigma_e^{sc,s} \quad \forall e \, \epsilon \, E^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{7}$$

$$\sigma_e^{sc,s} \geq 0 \; \forall sc \, \epsilon \, SC \tag{8}$$

where in p_e^o , q_e^o , p_e^d and q_e^d are the active (p) and reactive (q) power in the e line extremes. The upper index means the origin (o) or destiny (d) bus of the line e. E^{sc} is the set of active transmission lines in the scenario sc. $v_{i_e}^o$ and $v_{i_e}^d$ are the voltage magnitudes in the origin and destiny buses respectively. $\overline{R}_e$ is the line e maximal apparent current. $\sigma_e^{sc,s}$ is the upper bound penalty for line current rating violation in the origin and destiny buses in the scenario sc.

The transformers overload constraints considered were the following:

$$\sqrt{(p_f^o)^2 + (q_f^o)^2} \leq \overline{s}_f + \sigma_f^{sc,s} \quad \forall f \, \epsilon \, F^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{9}$$

$$\sqrt{(p_f^d)^2 + (q_f^d)^2} \leq \overline{s}_f + \sigma_f^{sc,s} \quad \forall f \, \epsilon \, F^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{10}$$

$$\sigma_e^{sc,s} \geq 0 \; \forall sc \, \epsilon \, SC \tag{11}$$

$$\sigma_f^{sc,s} \geq 0 \; \forall sc \, \epsilon \, SC \tag{12}$$

where in p_f^o , q_f^o , p_f^d and q_f^d are the active (p) and reactive (q) power in the f transformer extremes. The upper index means the origin (o) or destiny (d) bus of the transformer f. F^{sc} is the set of transformers in the scenario sc . The apparent power rating is defined as $\overline{S}_f$ for the f transformer. $\sigma_f^{sc,s}$ is the upper bound penalty for the f transformer in the scenario sc.

The commutable shunts were modeled as generators with null real power. The reactive power constraints were considered as:

$$\underline{b_i^{cs}} v_i^2 \leq q_i^{cs} \leq \overline{b_i^{cs}} v_i^2 \quad \forall i \, \epsilon \, I^{sc} \; \wedge \forall sc \, \epsilon \, SC \tag{13}$$

where q_i^{cs} and b_i^{cs} are the reactive power and the susceptance value of commutable shunt on bus i respectively (if it has one). v_i is the voltage magnitude on the same bus.

The power balance constraints considered were:

$$\sum_{g \epsilon G_i^{sc}} p_g - p_{L_i}^{sc} - g_{fs_i}^{sc} v_i^2 - \sum_{e \epsilon E_i^{sc,o}} p_e^o - \sum_{e \epsilon E_i^{sc,d}} p_e^d - \sum_{f \epsilon F_i^{sc,o}} p_f^o - \sum_{f \epsilon F_i^{sc,d}} p_f^d = \sigma_i^{sc,P+} - \sigma_i^{sc,P-} \; \forall sc \; \epsilon \; SC \quad (14)$$

$$\sigma^{sc,P+} \geq 0 \; \forall sc \; \epsilon \; SC \quad (15)$$

$$\sigma^{sc,P-} \geq 0 \; \forall sc \; \epsilon \; SC \quad (16)$$

for active power, and

$$\sum_{g \epsilon G_i^{sc}} q_g - q_{L_i}^{sc} - (-b_{fs_i}^{sc} - b_{cs_i}^{sc}) v_i^2 - \sum_{e \epsilon E_i^{sc,o}} q_e^o - \sum_{e \epsilon E_i^{sc,d}} q_e^d - \sum_{f \epsilon F_i^{sc,o}} q_f^o - \sum_{f \epsilon F_i^{sc,d}} q_f^d = \sigma_i^{sc,Q+} - \sigma_i^{sc,Q-} \; \forall sc \; \epsilon \; SC \quad (17)$$

$$\sigma_i^{sc,Q+} \geq 0 \; \forall sc \; \epsilon \; SC \quad (18)$$

$$\sigma_i^{sc,Q-} \geq 0 \; \forall sc \; \epsilon \; SC \quad (19)$$

for reactive power.

where $p_{L_i}^{sc}$ and $q_{L_i}^{sc}$ are the active and reactive load power on bus i in the scenario sc. $g_{fs_i}^{sc}$ and $b_{fs_i}^{sc}$ are the conductance and susceptance of the fixed shunts on bus i in the scenario sc. $E^{sc,o}$ and $E^{sc,d}$ are the set of active lines in the scenario sc in the origin and destination buses. $F^{sc,o}$ and $F^{sc,d}$ are the set of active transformers in the scenario sc in the origin and destination buses.

$\sigma_i^{sc,P+}$ and $\sigma_i^{sc,P-}$ ($\sigma_i^{sc,Q+}$ and $\sigma_i^{sc,Q-}$) are the positive and negative parts of the violation of active (reactive) power balance for bus i in the scenario sc.

An area spin reserve constraint was considered to prevent the lack of power when generation contingencies occur.

$$\sum_{g \epsilon A} (\overline{P}_g - P_g) + \sigma_{A,sc} \geq max_{g \epsilon (\chi_{sc} \cap A)} \overline{P}_g \quad (20)$$

where χ_{sc} is the set of generators that appear in a contingency of scenario sc and $\sigma_{A,sc}$ is an area spin reserve slack variable for each affected area A in the sc contingency.

The penalization cost of 1 for the scenario sc is computed according to slack variables as follows:

$$C_{sc}^{\sigma} = \sum_{i \epsilon I} \alpha_p (\sigma_i^{sc,P+} - \sigma_i^{sc,P-}) + \sum_{i \epsilon I} \alpha_q (\sigma_i^{sc,Q+} - \sigma_i^{sc,Q-}) + \sum_{f \epsilon F} \alpha_{ef} (\sigma_f^{sc,s}) + \sum_{e \epsilon E} \alpha_{ef} (\sigma_e^{sc,s}) + \alpha_{\sigma} \sigma_{A,sc}$$

According to secondary frequency regulation in multi-area systems, each control area should be able to recover the frequency according to participation factors predefined by the network operator [31]. After the contingency has occurred, in steady state the active power generated must satisfy :

$$P_g^k = \begin{cases} \underline{P_g} & \text{if } P_g^0 + a_{pf} \Delta P \leq \underline{P_g} \\ P_g^0 + a_{pf} \Delta P & \text{if } \underline{P_g} < P_g^0 + a_{pf} \Delta P < \overline{P_g} \\ \overline{P_g} & \text{if } P_g^0 + a_{pf} \Delta P \geq \overline{P_g} \end{cases} \quad (21)$$

where P_g^k is the vector of active power generated in the affected area to the the k contingency. a_{pf} is the vector of participation factors. P_g^0 is the vector of active power in the base case, ΔP is the difference of real generation between the base case and the contingency.

The reactive power re-dispatch is performed according to voltage stability criteria. In principle, the voltage control tries to maintain the pre-fault voltage magnitude unless reactive limits are violated [32]:

$$\begin{cases} V_g^k = V_g^0 & \text{if } \underline{Q_g} \leq Q_g^k \leq \overline{Q_g} \\ \underline{V_g} < V_g^k < V_g^0 & \text{if } Q_g^k = \overline{Q_g} \\ \overline{V_g^0} < V_g^k < \overline{V_g} & \text{if } Q_g^k = \underline{Q_g} \end{cases} \tag{22}$$

Where Q_g^k is the reactive power generated vector in the contingency k, V_g^0 and V_g^k are voltage magnitude vectors in the generators in contingency k and in the base case, respectively.

2.2. Proposed approach

Considering the set of constraints corresponding to each contingency is a computationally demanding problem when a huge amount of contingencies are to be evaluated. In order to simplify the complete SCOPF problem, the methodology applied in this research pretends to cover the post-contingency restrictions by means of constraint handling of the base case. That to say, this methodology only employs optimization for the base case subject to constraints only considering sc=0, that is, without considering c_k^σ terms in Equation 1, neither constraints with $sc > 0$.

According to the latter reasoning, the objective function and constraints equations for the base case in the m iteration is given by minimize :

$$C = (\sum_{g \epsilon G} C_g + \delta C^\sigma) \tag{23}$$

The lower and upper bounds of the variable x for the iteration m is denoted respectively as $\underline{x}(m)$ and $\overline{x}(m)$, so the constraints are:

$$\underline{v_i}(m) \leq v_i \leq \overline{v_i}(m) \quad \forall i \, \epsilon \, I \tag{24}$$

$$\underline{p_g}(m) \leq p_g \leq \overline{p_g}(m) \quad \forall g \, \epsilon \, G \tag{25}$$

$$\underline{q_g}(m) \leq q_g \leq \overline{q_g}(m) \quad \forall g \, \epsilon \, G \tag{26}$$

$$\sqrt{(p_e^o)^2 + (q_e^o)^2} \leq \overline{R}_e(m) v_{i_e}^o + \sigma_e^s \quad \forall e \, \epsilon \, E \tag{27}$$

$$\sqrt{(p_e^d)^2 + (q_e^d)^2} \leq \overline{R}_e(m) v_{i_e}^d + \sigma_e^s \quad \forall e \, \epsilon \, E \tag{28}$$

$$\sqrt{(p_f^o)^2 + (q_f^o)^2} \leq \overline{s}_f(m) + \sigma_f^s \quad \forall f \, \epsilon \, F \tag{29}$$

$$\sqrt{(p_f^d)^2 + (q_f^d)^2} \leq \overline{s}_f(m) + \sigma_f^s \quad \forall f \, \epsilon \, F \tag{30}$$

$$\underline{b_i^{cs}} v_i^2 \leq q_i^{cs} \leq \overline{b_i}^{cs} v_i^2 \quad \forall i \, \epsilon \, I \tag{31}$$

$$\sum_{g \epsilon G_i} p_g - p_i^L - g_i^{FS} v_i^2 - \sum_{e \epsilon E_i^0} p_e^o - \sum_{e \epsilon E_i^d} p_e^d - \sum_{f \epsilon F_i^o} p_f^o - \sum_{f \epsilon F_i^d} p_f^d = \sigma_i^{P+} - \sigma_i^{P-} \tag{32}$$

$$\sum_{g \epsilon G_i} q_g - q_i^L - (-b_i^{FS} - b_i^{CS}) v_i^2 - \sum_{e \epsilon E_i^0} q_e^o - \sum_{e \epsilon E_i^d} q_e^d - \sum_{f \epsilon F_i^o} q_f^o - \sum_{f \epsilon F_i^d} q_f^d = \sigma_i^{Q+} - \sigma_i^{Q-} \tag{33}$$

$$\sum_{g \epsilon A} (\overline{P}_g(m) - P_g) + \sigma_A \geq max_{g \epsilon (\chi \bigcap A)} \overline{P}_g \tag{34}$$

The penalization cost for the base case is computed according to slack variables as follows:

$$C^{\sigma} = \sum_{i \epsilon I} \alpha_p (\sigma_i^{P+} - \sigma_i^{P-}) + \sum_{i \epsilon I} \alpha_q (\sigma_i^{Q+} - \sigma_i^{Q-}) + \sum_{f \epsilon F} \alpha_{ef} (\sigma_f^s) + \sum_{e \epsilon E} \alpha_{ef} (\sigma_e^s) + \alpha_{\sigma} \sigma_A \tag{35}$$

Where α_p, α_q, α_{ef} and α_{σ} are penalization functions. Each of these penalization functions satisfies that:

$$\alpha(x) = \begin{cases} k_1|x| & \text{if } 0 \leq |x| \leq x_1 \\ k_2|x| & \text{if } x_1 < |x| < x_2 \\ k_3|x| & \text{if } |x| \geq x_2 \end{cases} \tag{36}$$

The last optimization problem can be summarized as the following vector optimization problem:

$$\begin{aligned} \underset{x_c,\, u_0}{\text{minimize}} \quad & f(x_0, u_0) \\ \text{subject to} \quad & g(x_c, u_0) = 0, \qquad c\, \epsilon \{0\} \cup C, \\ & h(x_c, u_0) \leq B(m), \quad c\, \epsilon \{0\} \cup C \end{aligned} \tag{37}$$

The IPOPT solver [33] was used to compute the optimal base case in each iteration.

3. Methodology

The first step of the methodology consists of the decomposition of the SCOPF problem into a base case problem and contingency sub-problems. Here, several OPF problems were solved using a base case network. The second stage dealt with the modification of the base case by updating some of the constraints limits (constraints handling) according to the evaluation of potentially relevant contingencies.

An algorithm based on the Matpower toolbox [34] and the Interior Point Optimizer (IPOPT [35][33]) was proposed to solve the SCOPF problem described in Section 2 [30][36]. The algorithm consisted of *Pre- and Post-processing* stages (input and output data) and other stages classified into three groups: *Parallel OPF*, *Contingencies*, and *Constraints Handling*.

Figure 1 shows the approach used for updating the constraint limits and solving the SCOPF problem in a iterative way. The *Pre- and Post-processing* stages include two blocks named **SwShunts2Gen** and **SwGen2Shunts**, see Figure 1. These blocks treated the switched shunts as generators with null active power before the OPF computation, and then these "reactive generators" were converted back to switched shunts before the algorithm delivered the final solution. The other stages, named *Parallel OPF*, *Contingencies*, and *Constraints Handling* are described in the following subsections. Finally, two stop rules for the loop depicted in Figure 1 were implemented: when the penalty cost is lower than a certain percentage of the objective function, or when a number of iterations had been reached.

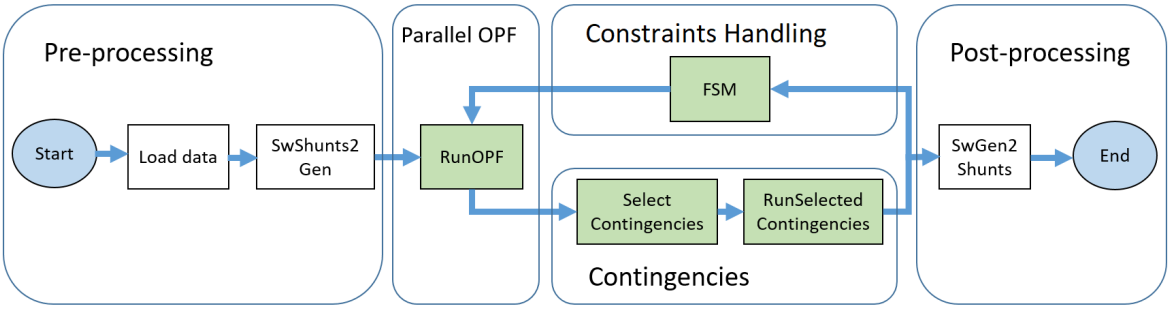

Figure 1. Overview of sections composing the main algorithm.

3.1. Parallel OPF (Optimal Power Flow)

The **RunOPF** block was executed in parallel and is summarized in Figure 2. The **load parallel seeds function** sets different configuration parameters for interior point solver (IPOPT). The initial conditions of optimization variables, the linear solver used (like the Multifrontal Massively Parallel sparse direct Solver (MUMPS), MA57, or MA86), and tolerance levels were combined in multiples workers due to their influence in the convergence time.

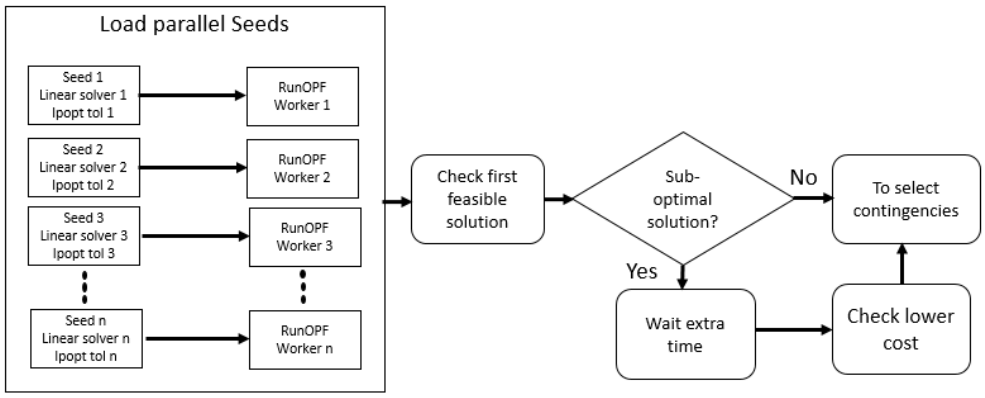

Figure 2. Parallel optimal power flow.

If a configuration with a sub-optimal cost is the first in reach a feasible solution, extra time is given to find better solutions. After that, the solution with the lowest cost is chosen to continue towards the **screening contingencies** stage.

3.2. Contingencies

3.2.1. Contingencies ranking and screening

The contingencies ranking and screening was used as a speed-up strategy for medium- and large-scale networks (i.e. more than 1,000 buses). The strategy was used separately for branches and generation contingencies, as explained in Figure 3.

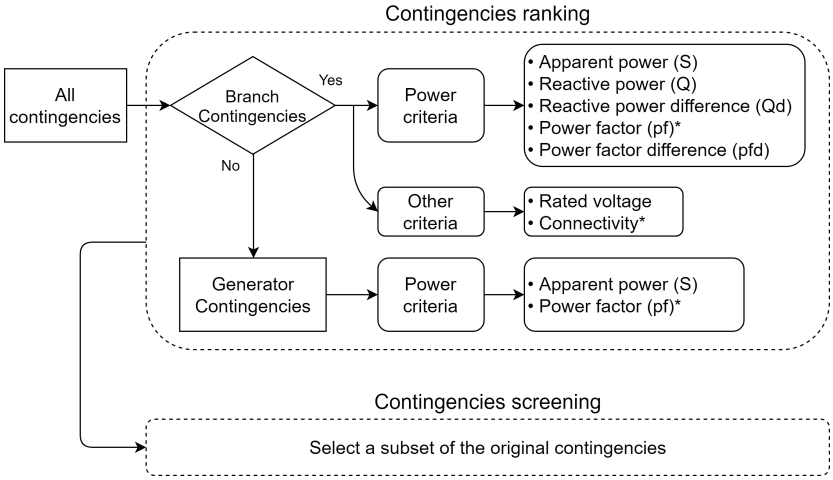

Figure 3. Contingencies selection flowchart.

The criteria used in Figure 3 are described below:

- Apparent Power (S): Main criteria for both branch and generator contingencies.
- Reactive Power (Q): Used in branch contingencies.
- Reactive Power Difference (Qd): Computed for branch and generator contingencies as Q difference Between "origin" and "destiny" buses.
- Power factor (pf): computed using Equation 38 for both branch and generator contingencies.

$$pf = cos\left(atan\left(\frac{Q}{P}\right)\right) \tag{38}$$

Where:

  - pf: Power factor
  - $atan$: Four-quadrant inverse tangent function
  - P, Q: Active and Reactive power

- Power factor difference (pfd): Similar to Reactive power difference, the Power factor difference was computed between "from" and "to" buses.
- Rated voltage: This criteria sorted branch contingencies according to the highest rated voltage between "origin" and "destiny" buses.
- Connectivity: It sorted branches according to the amount of elements their buses were connected to.

All lists from selected criteria shown in Figure 3 were sorted in descendent order but *Power Factor* and *Connectivity*, which were sorted in ascendant order and marked with * in Figure 3.

After all lists were sorted according to the aforementioned criteria, a unique list of contingencies were computed from combination of those criteria. Each branch and generator were respectively ranked on each of the 7 and 2 lists depicted in Figure 3. Given this, all the positions a single branch or generator occupied in the different lists were averaged to create a unique list of sorted contingencies for branches and generators respectively. From these two lists, one for branches and one for generators, the top x contingencies were selected to continue as the input for *Contingency evaluation* stage.

3.2.2. Contingency evaluation

Different workers simultaneously executed the steps summarized in Figure 4 through parallel computation. First, the contingency element is outaged and conventional power flow is executed with the same load and generated active power as in the base case.

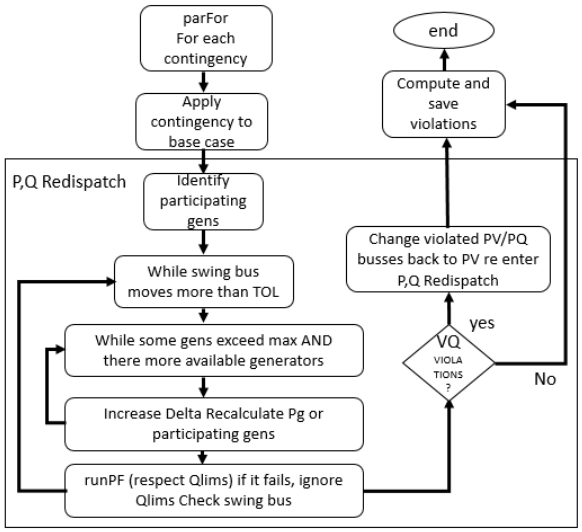

Figure 4. Active and reactive re-dispatch algorithm.

The difference between total active power generated in the base case and the result of power flow is a first ΔP (Delta) estimation (Equation 21). According to this value, the active power set point in post contingency are adjusted taking into account the participation factors. If a generator tried to exceed its limit, it is saturated and the ΔP value is increased neglecting the participating factor of saturated generators.

The differences of active power between the programmed value and power flow result are due to QV violations in some generators. To fix the reactive power violations in PV buses these are converted to PQ to satisfy the equation 22. Nevertheless, this lead to changes in voltage magnitude in PQ buses and leads power losses system .

According to slack deviation, the delta value is recalculated and the power flow is executed again until reaching a minimal deviation.

3.3. Constraints Handling rules

3.3.1. Finite State Machine

This stage was created to update constraints limits of the base case according to the no convergences, soft (overloads) and hard (voltage) violations identified by the **run selected Contingencies** stage in Figure 1. After the set of the contingencies selected was evaluated and the summary of the largest violations was established (overloads, voltage violations, and non-convergences), the Finite State Machine updates the limits of the initial constraints for the OPF of next iteration. Figure 5 shows the different states and transitions of the **FSM** stage.

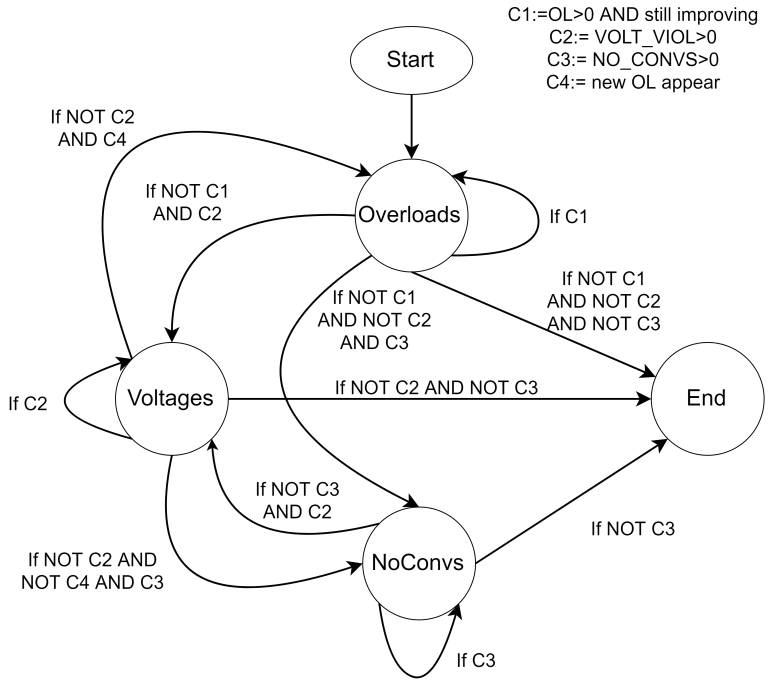

Figure 5. FSM transitions

A detailed explanation of the three states composing the Finite State Machine are explained below.

3.3.2. Updating constraints limits

The constraints limits were tuned in function of the current state of FSM (Figure 5). These rules affect the $B(m)$ term in Equation 37.

If the current state is **overloads**, the rules for updating limits shown in Figure 6 will be applied. *Branch* is used to refer both transmission lines and transformers.

If the current state is **Voltages**, the limit update rule is shown in Figure 7 will be applied. In this case, marginal violation are defined as violation that satisfied:

$$((v_k - v_{min} < 0.125)||(v_{max} - v_k < 0.125))\& \, |v_{base} - v_{lim}| < 0.01 \tag{39}$$

The power factor of complex power in the destination or origin extreme was validated as another criteria to update the voltage limit. In this case the voltage can update both undervoltages and overvoltages similarly.

If the marginal voltage or power factor criteria is not met, a new set of constraints is added in the case base:

$$q_f^d < q_{f_{max}} \tag{40}$$

$$q_f^o < q_{f_{max}} \tag{41}$$

If is a transformer contingency, and if is a line contingency:

$$q_e^d < q_{e_{max}} \tag{42}$$

$$q_e^o < q_{e_{max}} \tag{43}$$

If the state present is voltages, the limits are updated according the figure 7:

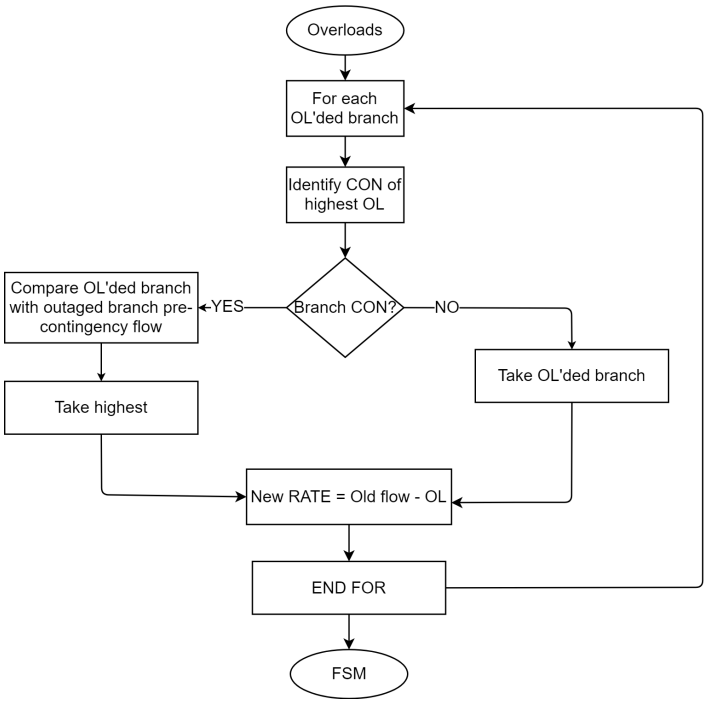

Figure 6. Branch limits updating

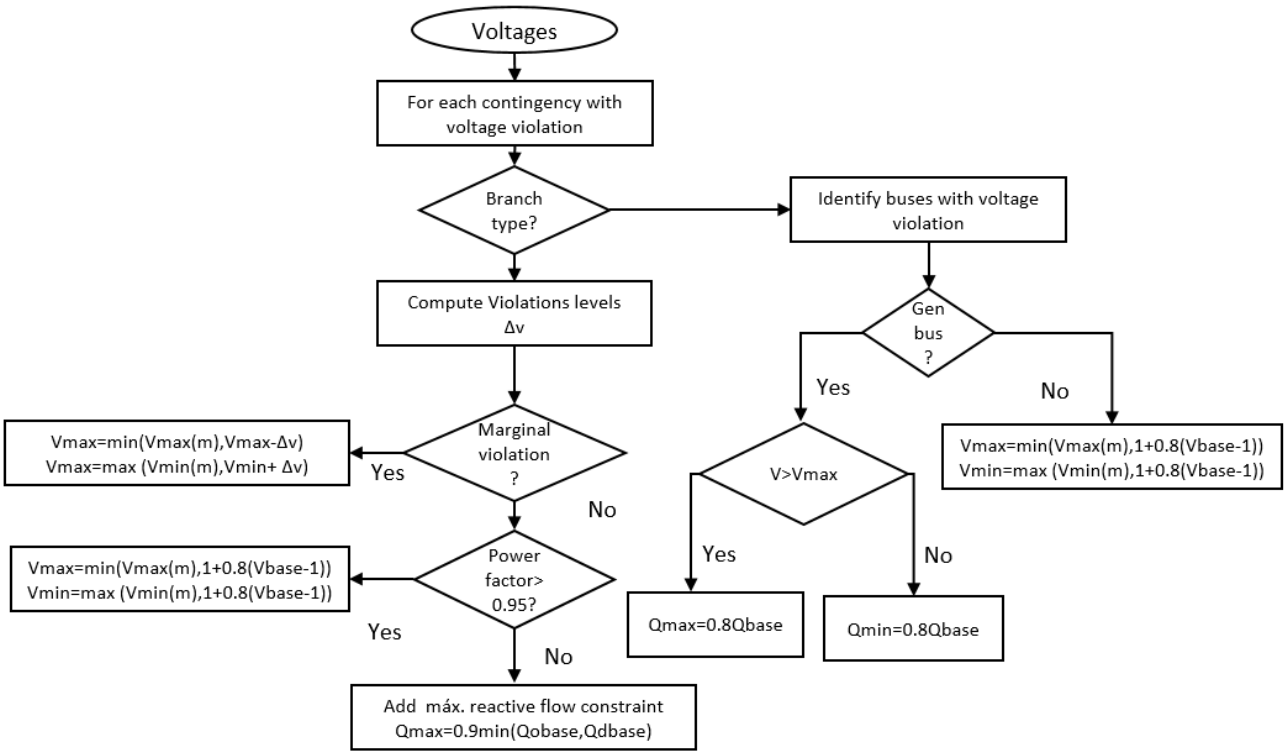

Figure 7. Voltage Limits Updating

Finally, if the present state is **non convergence**, the strategy is reduce the upper limits $\overline{P_g}(25\%), \overline{R_e}(10\%), \overline{s}_f(10\%)$ depending on whether the element that is in contingency is a generator, a transmission line or a transformer.

4. Results

The algorithm proposed in this research was implemented through the Matpower toolbox [34] and IPOPT solver [33][35] to solve the SCOPF problem described in Section 2. The set of tested networks are described in Table 1. These datasets were used in the Grid Optimization Competition - Challenge 1. More information about these and other datasets can be found in [36]. Results from the evaluation of the networks listed in Table 1 are presented and discussed below. All the networks tested were run in a 64-bits Linux distribution of Matlab® 2019, Intel(R) Xeon(R) CPU E5-2680 @ 2.70GHz core i7, 128 GB RAM memory.

Table 1. Description of networks tested.

Network	Buses	Generators	Loads	Branches	Transformers	Contingencies	Shunts	Areas
1	500	224	281	540	193	786	44	1
2	4,918	1,340	3,070	4,412	2,315	5,085	732	31
3	11,615	899	19,272	13,967	5,936	8,747	1,332	1

A set of different initial points were used for the optimal power flow computation. The solver used for the OPF problem was IPOPT [33]. Since the IPOPT methodology strongly depends of the initial point [11], a set of 16 different initial points were proposed to run the OPF algorithm in parallel. Each combination was executed at a different CPU core. These different combinations changed the initial point for IPOPT solver and included the following fields :

- Linear Solver: mumps, ma57 [33].
- Strategy: monotone (default), adaptive [33].
- Oracle: quality function (default), Loqo [33].
- Seed: initialization of decision variables from a base case (warm starting), from "zero" condition (cold starting) or from previous algorithm iterations [37],
- Initial voltage: from a base case (warm starting) or set to 1 p.u.

To test the performance of contingencies selection (CS), a scenario was tested for each network of Table 1. CS was made for each iteration of the algorithm, leaving a fixed percentage of selection (25%, 50%, 75%, 100%) of the total number of sorted generation and branch contingencies.

In each iteration, an evaluation of 100% of the contingencies was carried out in order to determine the total number of violations (overvoltage, undervoltage, overloads) and non-convergences. In addition, the operating cost of the system was computed by taking into account the penalties for overload and power unbalance.

Voltage violations for the k contingency are translated into a power unbalance, according to 32 and 33 equations. To compute the penalization cost, the equation 36 was used with $k_1 = 1000$, $k_2 = 5000$, $k_3 = 10e6$, $x_1 = x_2 = 2$, and $x_3 = 50$

Figure 8 shows the cost normalized in logarithmic scale ($Cost^*$) and violations number in function of the algorithm iterations for the network 1. The transformed cost was computed because the unbalance penalization was higher with respect to the minimum cost in the first three iterations, so it was computed as follows:

$$Cost^* = log_{10}(\frac{Cost}{C_{min}}) \tag{44}$$

In all cases, the algorithm ends within 5 iterations. The minimal cost C_{min} was $2.63e5$\$. A 50% of CS is enough to achieve the minimal cost.

The computation time using 25% was 33.08 seconds, and for 100% was 42 seconds. Therefore just a 26% of computation time was increased when 75% more contingencies are evaluated.

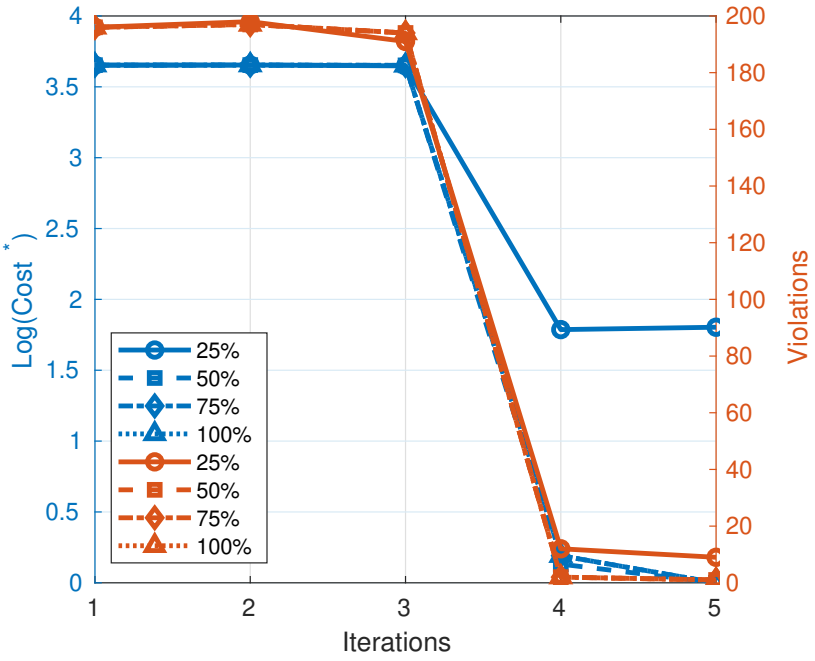

Figure 8. Cost and violations in function of iterations for network 1

Figure 9 depicts the costs and number of violations in function of the iteration of the algorithm for 4,918-buses network. In this case, the operation cost is minimal for 50% and highest for 75% of CS. For 100% the algorithm save the best iteration, so the minimal cost is reached in the iteration 5.

In terms of violations, with 50% of CS there were 33 violations and with 100% the algorithm ended with 23. This shows that it is not always cheaper to have fewer violations. In terms of time, for 25% of CS, the algorithm ended in the lowest time (682s) and final cost was lower than for 100%. With 75% of CS, the iterations were only 4, but the computation time was higher than with 25%. In general terms for a real time approach 25% of CS had an acceptable performance for this scenario.

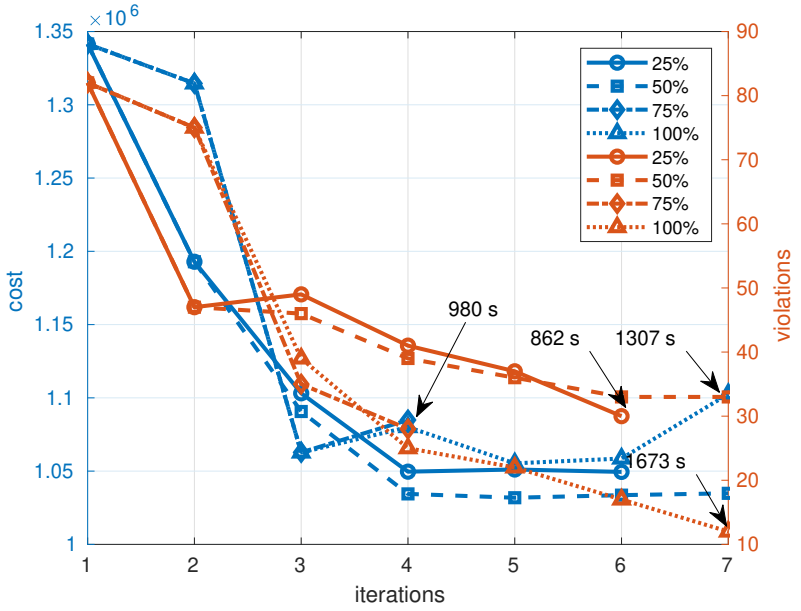

Figure 9. Cost and violations in function of iterations for network 2

In Figure 10, the transformed cost 44 is used again. In this case, a low cost is reached with 50%, 75%, and 100% of CS. With 25%, the penalization is approximately of 64 times higher than the minimal cost.

In Table 2 is summarized the results for the network 3. With 25% the algorithm ends with 22 violations of voltages. Although only 0.18% of the total number of the buses is violated the unbalance cost could be unacceptable. With 50%, 75%, and 100%, the penalty cost of violation could be acceptable. In terms of time, the best performance was reached with 75% of CS.

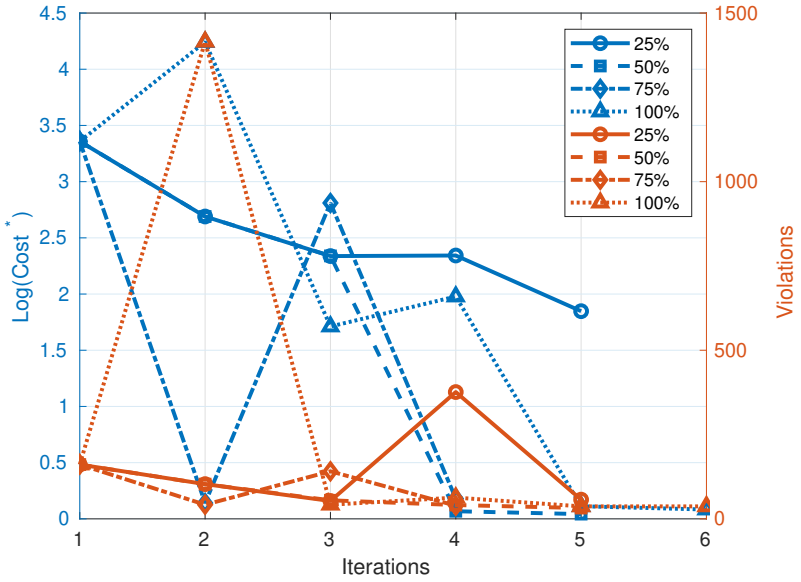

Figure 10. Cost and violations in function of iterations for network 3

Table 2. Summary Results for different percentage of CS

Screening percentage	25	50	75	100
Convergence Time [s]	2200	2817	2542	5352
Cost (1e8) [$]	1.3467	0.0211	0.0264	0.0231
Violations [$]	22	2	0	2

Figure 11 shows the average time per contingency in function of the network size by using 16 and 72 workers. The time differences between them for a 11,615 network is 52 ms/cont, therefore, for 75% (6,560 contingencies) the time would decrease in around 341 s per iteration. For four iterations, the computation time would have been about 1,178 s (19.6 minutes) instead of 2,542s, reducing the calculation time by more than twice.

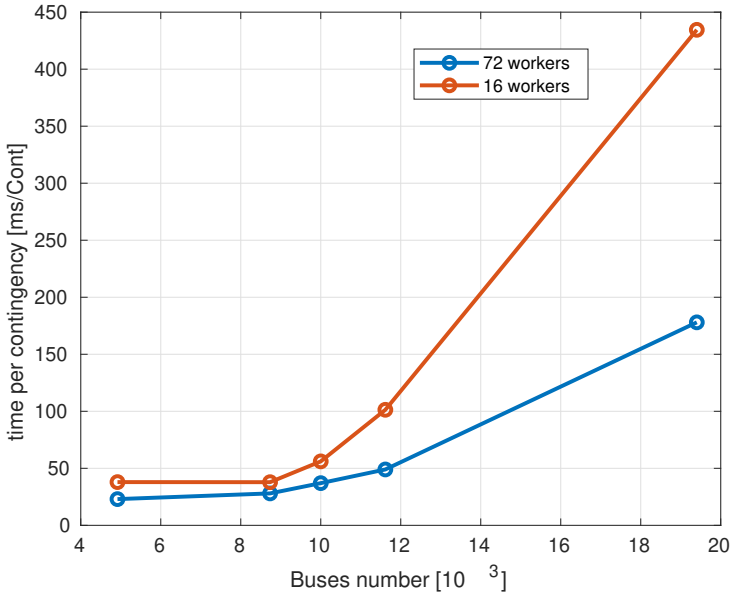

Figure 11. Average time per contingency in function of number of buses.

5. Discussion

The selection of contingencies strongly depends on the size of the analyzed network. Indeed, the results obtained from network 1 (500-bus system) showed that the selection of contingencies is not worth it for a real-time approach using the proposed methodology. The relevance of contingency selection increases as the network size also do.

The percentage of CS can be chosen according to the calculation time available by the user. For example, it is possible to estimate with less uncertainty how many contingencies should be chosen to arrive at a desired number of iterations of the algorithm.

Some times the algorithm can steeply increase the costs from one iteration to another (as in the third iteration for 75% selection in Figure 10). However, in the next iteration it can return to a better cost. In case it does not improve, the best solution is saved from one iteration to another until the pre-set time of calculation is finished.

The number of cores available for the calculation is another determining factor, since the evaluation of contingencies is the bottleneck of the algorithm. The desired percentage of contingencies could be selected from the available time, the number of cores and the number of busses in the network. Once the desired percentage is known, the criteria of Figure 3 would be applied.

6. Conclusions

An algorithm has been developed that is capable of solving the problem of SCOPF for large networks with a large number of contingencies when compared with the dimensions of the networks that are usually used in the cited literature. The advantages of using parallel computing have been highlighted: both to reduce the uncertainty of the seed and the solver in the OPF solution, and to quickly perform the contingency evaluation stage of the algorithm.

Solving an optimization problem only for the base case of the OPF requires less time compared with strategies such as Bender's Descomposition that requires solving an optimization problem for each contingency. Our strategy only executes conventional power flows, which saves even more time when the number of contingencies is as high as in the evaluated networks.

The criteria for selecting contingencies using the proposed algorithm show good performance. All cases showed that it was enough to select 50% of the contingencies to arrive at final costs similar to when 100% of the contingencies were evaluated.

It is suggested for future work to introduce uncertainty costs to model renewable injection, restrictions in storage and ramp elements to model the limitation of actuators. Additionally, new strategies to update the limits could be researched, looking for less iterations to reach a lower cost. A network partitioning algorithm could be used to reach a better performance.

Author Contributions: For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "conceptualization, X.X. and Y.Y.; methodology, X.X.; software, X.X.; validation, X.X., Y.Y. and Z.Z.; formal analysis, X.X.; investigation, X.X.; resources, X.X.; data curation, X.X.; writing–original draft preparation, X.X.; writing–review and editing, X.X.; visualization, X.X.; supervision, X.X.; project administration, X.X.; funding acquisition, Y.Y.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: Please add: "This research received no external funding" or "This research was funded by NAME OF FUNDER grant number XXX." and and "The APC was funded by XXX". Check carefully that the details given are accurate and use the standard spelling of funding agency names at https://search.crossref.org/funding, any errors may affect your future funding.

Acknowledgments: The information, data, or work presented herein was funded in part by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0001090. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Authors also would like to thank the Electrical Engineering Department of Universidad Nacional de Colombia

Conflicts of Interest: Declare conflicts of interest or state "The authors declare no conflict of interest." Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript, or in the decision to publish the results must be declared in this section. If there is no role, please state "The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results".

References

1. Wang, Q.; McCalley, J.D.; Zheng, T.; Litvinov, E. A Computational Strategy to Solve Preventive Risk-Based Security-Constrained OPF. *IEEE Transactions on Power Systems* **2013**, *28*, 1666–1675. doi:10.1109/tpwrs.2012.2219080.
2. Sadat, S.A.; Haralson, D.; Sahraei-Ardakani, M. Security versus Computation Time in IV-ACOPF with SOCP Initialization. 2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS). IEEE, 2018. doi:10.1109/pmaps.2018.8440287.
3. Capitanescu, F.; Ramos, J.M.; Panciatici, P.; Kirschen, D.; Marcolini, A.M.; Platbrood, L.; Wehenkel, L. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electric Power Systems Research* **2011**, *81*, 1731–1741. doi:10.1016/j.epsr.2011.04.003.
4. Capitanescu, F. Critical review of recent advances and further developments needed in AC optimal power flow. *Electric Power Systems Research* **2016**, *136*, 57–68. doi:10.1016/j.epsr.2016.02.008.
5. Phan, D.; Kalagnanam, J. Some Efficient Optimization Methods for Solving the Security-Constrained Optimal Power Flow Problem. *IEEE Transactions on Power Systems* **2014**, *29*, 863–872. doi:10.1109/TPWRS.2013.2283175.
6. Mohammadi, J.; Hug, G.; Kar, S. A benders decomposition approach to corrective security constrained OPF with power flow control devices. 2013 IEEE Power & Energy Society General Meeting. IEEE, 2013. doi:10.1109/pesmg.2013.6672684.
7. Phan, D.T.; Sun, X.A. Minimal Impact Corrective Actions in Security-Constrained Optimal Power Flow Via Sparsity Regularization. *IEEE Transactions on Power Systems* **2015**, *30*, 1947–1956. doi:10.1109/tpwrs.2014.2357713.
8. Al-Saffar, M.; Musilek, P. Distributed Optimal Power Flow for Electric Power Systems with High Penetration of Distributed Energy Resources. 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE). IEEE, 2019. doi:10.1109/ccece.2019.8861718.
9. Louca, R.; Bitar, E. Robust AC Optimal Power Flow. *IEEE Transactions on Power Systems* **2019**, *34*, 1669–1681. doi:10.1109/tpwrs.2018.2849581.

10. Li, Y.; McCalley, J. Decomposed SCOPF for Improving Efficiency. *IEEE Transactions on Power Systems* **2009**, *24*, 494–495. doi:10.1109/tpwrs.2008.2002166.
11. Capitanescu, F.; Glavic, M.; Ernst, D.; Wehenkel, L. Applications of security-constrained optimal power flows. In Proceedings of Modern Electric Power Systems Symposium, MEPS06, 2006.
12. Sojoudi, S.; Lavaei, J. Physics of power networks makes hard optimization problems easy to solve. 2012 IEEE Power and Energy Society General Meeting. IEEE, 2012. doi:10.1109/pesgm.2012.6345272.
13. Xu, Y.; Yang, H.; Zhang, R.; Dong, Z.; Lai, M.; Wong, K. A contingency partitioning approach for preventive-corrective security-constrained optimal power flow computation. *Electric Power Systems Research* **2016**, *132*, 132–140. doi:10.1016/j.epsr.2015.11.012.
14. Xu, Y.; Dong, Z.Y.; Zhang, R.; Wong, K.P.; Lai, M. Closure to Discussion on "Solving Preventive-Corrective SCOPF by a Hybrid Computational Strategy". *IEEE Transactions on Power Systems* **2014**, *29*, 3124–3125. doi:10.1109/tpwrs.2014.2359354.
15. Karangelos, E.; Wehenkel, L. An Iterative AC-SCOPF Approach Managing the Contingency and Corrective Control Failure Uncertainties With a Probabilistic Guarantee. *IEEE Transactions on Power Systems* **2019**, *34*, 3780–3790. doi:10.1109/tpwrs.2019.2902486.
16. Hinojosa, V.; Gonzalez-Longatt, F. Preventive Security-Constrained DCOPF Formulation Using Power Transmission Distribution Factors and Line Outage Distribution Factors. *Energies* **2018**, *11*, 1497. doi:10.3390/en11061497.
17. Javadi, M.; Nezhad, A.E.; Gough, M.; Lotfi, M.; Catalao, J.P. Implementation of Consensus-ADMM Approach for Fast DC-OPF Studies. 2019 International Conference on Smart Energy Systems and Technologies (SEST). IEEE, 2019. doi:10.1109/sest.2019.8848992.
18. Attarha, A.; Amjady, N. Solution of security constrained optimal power flow for large-scale power systems by convex transformation techniques and Taylor series. *IET Generation, Transmission & Distribution* **2016**, *10*, 889–896. doi:10.1049/iet-gtd.2015.0494.
19. Werner, A.; Duwadi, K.; Stegmeier, N.; Hansen, T.M.; Kimn, J.H. Parallel Implementation of AC Optimal Power Flow and Time Constrained Optimal Power Flow using High Performance Computing. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, 2019. doi:10.1109/ccwc.2019.8666551.
20. Lee, S.; Kim, W.; Kim, B.H. Performance Comparison of Optimal Power Flow Algorithms for LMP Calculations of the Full Scale Korean Power System. *Journal of Electrical Engineering and Technology* **2015**, *10*, 109–117. doi:10.5370/jeet.2015.10.1.109.
21. Stankovic, S.; Soder, L. Optimal Power Flow Based on Genetic Algorithms and Clustering Techniques. 2018 Power Systems Computation Conference (PSCC). IEEE, 2018. doi:10.23919/pscc.2018.8442583.
22. Zamzam, A.; Baker, K. Learning optimal solutions for extremely fast ac optimal power flow. *arXiv preprint arXiv:1910.01213* **2019**.
23. Ghosh, I.; Roy, P.K. Application of Earthworm Optimization Algorithm for solution of Optimal Power Flow. 2019 International Conference on Opto-Electronics and Applied Optics (Optronix). IEEE, 2019. doi:10.1109/optronix.2019.8862335.
24. Lastomo, D.; Widodo.; Setiadi, H. Optimal Power Flow using Fuzzy-Firefly Algorithm. 2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). IEEE, 2018. doi:10.1109/eecsi.2018.8752903.
25. Guo, J.; Hug, G.; Tonguz, O.K. A Case for Nonconvex Distributed Optimization in Large-Scale Power Systems. *IEEE Transactions on Power Systems* **2017**, *32*, 3842–3851. doi:10.1109/tpwrs.2016.2636811.
26. Grabada Echeverri, M.; Rider Flores, M.J.; Mantovani, J.R.S. Dos técnicas de descomposición aplicadas al problema de flujo de potencia óptimo multi-areas. *DYNA* **2010**, *77*, 303 – 312.
27. Guo, J.; Hug, G.; Tonguz, O. Asynchronous ADMM for Distributed Non-Convex Optimization in Power Systems. *arXiv preprint arXiv:1710.08938* **2017**.
28. Bouffard, F.; Galiana, F.D.; Arroyo, J.M. Umbrella contingencies in security-constrained optimal power flow. 15th Power systems computation conference, PSCC, 2005, Vol. 5.
29. Eftekharnejad, S. Selection of multiple credible contingencies for real time contingency analysis. 2015 IEEE Power Energy Society General Meeting, 2015, pp. 1–5. doi:10.1109/PESGM.2015.7286593.
30. ARPA-E. SCOPF Problem Formulation: Challenge 1. Technical report, Advanced Research Projects Agency–Energy), 2018.

31. Bevrani, H. *Robust Power System Frequency Control*; Power Electronics and Power Systems, Springer US, 2008.
32. Zhao, J.; Chiang, H.D.; Li, H.; Ju, P. On PV-PQ bus type switching logic in power flow computation. Proceedings of the 16th power systems computation conference, 2008, Vol. 16, p. 7.
33. HSL. *A collection of Fortran codes for large scale scientific computation*, 2019 (accessed December 5th, 2019). Availabel in: http://www.hsl.rl.ac.uk/.
34. Zimmerman, R.D.; Murillo-Sánchez, C.E. MATPOWER, 2019. doi:10.5281/ZENODO.3251119.
35. Wächter, A.; Biegler, L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming* **2005**, *106*, 25–57. doi:10.1007/s10107-004-0559-y.
36. ARPA-E. *Grid Optimization (GO) Competition*, 2019 (accessed December 5th, 2019). Availabel in: https://gocompetition.energy.gov.
37. Zimmerman, R.D.; Murillo-Sánchez, C.E. MATPOWER User's Manual, 2019. doi:10.5281/ZENODO.3251118.

Sample Availability: Samples of the compounds ..... are available from the authors.