A Generic Approach for Solving Nonlinear-Discrete Security-Constrained Optimal Power Flow Problems in Large-Scale Systems

Ludovic Platbrood, Florin Capitanescu, Christian Merckx, Horia Crisciu and Louis Wehenkel

***Abstract*—This paper proves the practicality of an iterative algorithm for solving realistic large-scale SCOPF problems. This algorithm is based on the combination of a contingency filtering scheme, used to identify the binding contingencies at the optimum, and a network compression method, used to reduce the complexity of the post-contingency models included in the SCOPF formulation. We show that by combining these two complementary ideas, it is possible to solve in a reasonable time SCOPF problems on large power system models with a large number of contingencies. Unlike most results reported for large-scale SCOPF problems, our algorithm uses a non-linear AC network model in both pre-contingency and post-contingency states, optimizes both active/reactive powers flows jointly, and treats the discrete variables. The proposed algorithm is implemented with state-of-the-art solvers and applied to two systems: a national grid with 2563 buses and 1297 contingencies, and a model of the European transmission network with 9241 buses and 12000 contingencies.**

***Index Terms*—Security-constrained optimal power flow, contingency filtering, network compression, network equivalents.**

I. Introduction

A. Motivation and Related Works

THE security-constrained optimal power flow (SCOPF) problem is in its general form a nonlinear, non-convex, static, large-scale optimization problem with both continuous and discrete variables [1,2,3,4]. The efficient solution of SCOPF problems is crucial for system operators, in the context of planning, operational planning and real-time operation. SCOPF problems have been formulated in the "preventive only" mode [1] and in the "corrective also" mode [2], the difference between these modes being that the former does not consider the possibility of re-scheduling controls in post-contingency states, except for automatic system response to contingencies (e.g., generators participation in frequency control, automatic tap-changers, etc.).

One of the main challenges of SCOPF problems is their huge size, especially when they are formulated for large-scale systems and/or when a large number of contingencies have to be considered [3,4]. Obtaining the direct solution of these problems for large-scale power systems is impossible due to memory limitation and/or prohibitive computation times.

However, in real-world applications a very large proportion of contingencies are generally not constraining the optimum, and for most contingencies, most operational constraints are inactive. This sparse nature of the problem means that contingency selection and network reduction methods could be applied effectively, simplifying the SCOPF computations on large-scale power systems.

In the SCOPF literature, four main classes of approaches have been proposed in order to mitigate the drawbacks of the direct approach [3]: (i) iterative contingency selection schemes [1,5,6,7,8,9], (ii) decomposition methods (e.g. Benders, Dantzig-Wolfe, Talukdar-Giras, etc.) [2,14], (iii) network compression [10], and (iv) iterative methods combining contingency selection and network compression [17].

These approaches require first solving a *continuous relaxation* of the SCOPF problem, for which the following state-of-the-art methods have been used [3]: sequential linear programming (SLP) [1,6,7,18], Newton method [2,5], interior-point method (IPM) [8,9,10,13,17,19], and conic programming [20].

SCOPF problems are solved by either decomposition [4] into the separate and successive optimization of the so-called *active-power sub-problem* [5,6,7,8,9,19,21] and *reactive power sub-problem* [5,18,20,21], or by a *full* approach that optimizes jointly both active and reactive powers [1,2,8,9,10,13,17].

Although several commercial SCOPF packages are available from various vendors and are routinely used by many system operators, the scientific literature reporting on experiments using SCOPF solvers on large-scale systems is still quite limited. Indeed most publications reporting on SCOPF algorithms provide only results obtained on small to medium sized power systems, rarely exceeding 1000 buses, except of the following works: Ref. [7] uses an SLP approach for a 12,965-bus system, Ref. [18] uses a SLP for the 3551 buses UK system, Ref. [19] relies on the IPM for a 2746 buses Polish system, Ref. [20] employs a conic programming for a 2383-bus Polish system, and Ref. [21] provides experiments with the IPM for a 3012-bus Polish system, and a 8387-bus European grid model. However, other than [18], these works rely on a *continuous relaxation of the discrete variables*.

B. Paper Contribution and Organization

We reported in [17] preliminary results with our iterative method, and specifically how some modules behave at the first iteration. The main contribution of this paper is to *prove the practicality of this unified iterative approach for solving SCOPF problems for large-scale systems*. This approach combines contingency filtering (CF) techniques used in an iterative SCOPF algorithm [8,9] with a network compression (NC) method [10]. Our algorithm thereby acts simultaneously along two complementary directions to reduce the overall complexity of the SCOPF problem. Indeed, the CF technique is used to quickly identify the binding contingencies at the optimum and hence limit the number of contingencies included in the SCOPF computations carried out at successive iterations. On the other hand, the NC method is used to reduce the size of each post-contingency model included in the SCOPF computations, by identifying the potentially affected areas for each contingency and by modeling only their corresponding constraints. In the rest of this paper we will use the acronym ISCOPF-NC (Iterative Security-Constrained Optimal Power Flow with Network Compression) to denote the proposed approach.

Importantly, and in contrast to most results reported for large-scale SCOPF problems, *the implementation of our ISCOPF-NC uses a non-linear AC network model in both pre-contingency and post-contingency states, optimizes both active/reactive powers flows jointly, and treats the discrete variables.* It exploits also recent developments in the context of nonlinear programming solvers [12].

We demonstrate the approach validity for both a national power system (with 2563 buses and 1297 contingencies) as well as the whole European Transmission Network (ETN) that contains 9241 buses and considers 12000 contingencies. We further implement the method using appropriate parallel dedicated computing architectures.

The rest of the paper is organized as follows. Section II provides an abstract formulation of the SCOPF problem and describes the contingency filtering and network compression approaches respectively. Section III details the proposed ISCOPF-NC algorithm. Section IV describes the dedicated computing architecture used. Section V provides a detailed formulation of the SCOPF problem and reports numerical results. Section VI concludes.

II. SCOPF PROBLEM AND MAIN ALGORITHMIC FEATURES

A. General SCOPF Problem Formulation

The SCOPF problem that we address in this paper can be compactly formulated in the following way [1,2], whereas a detailed formulation is provided in Section V.C:

$$\min f(x_0,u_0)$$
$$s.t.\begin{cases} g_0(x_0,u_0)=0 \\ h_0(x_0,u_0)\le 0 \\ g_k(x_k,u_k)=0, k\in C \\ h_k(x_k,u_k)\le 0, k\in C \\ -\tau\left[\dfrac{du_k}{d\tau}\right]_{\max} \le u_k-u_0-\Delta u_k \le \tau\left[\dfrac{du_k}{d\tau}\right]_{\max}, k\in C \end{cases} \quad (1)$$

where C is the set of postulated contingencies, subscript k refers to variables and constraints of the k^{th} post-contingency state, subscript "0" refers to variables and constraints of the base case (pre-contingency state), x and u denote the vectors of state and (continuous and discrete) control variables, $f(x,u)$ is the objective function, $g(x,u)$ and $h(x,u)$ denote the vectors of equality and inequality constraints, Δu_k is the vector of changes in control variables due to the automatic control of the system following the k^{th} contingency, τ is the time allowed to reach a feasible post-contingency state (equal to 0 in "preventive only" approach) and $\left[\dfrac{du_k}{d\tau}\right]_{\max}$ is the vector of rate of change of controls (e.g. ramp up/down rate of generator active power, transformer tap changer rate of change, etc.).

To ease the comprehension of the ISCOPF-NC algorithm of Section III we introduce hereafter two of its salient features: contingency filtering and network compression.

B. Contingency Filtering Technique

The contingency filter exploits the amount of constraints violation obtained after simulating all contingencies with a classical load flow program.

The filter uses the Non Dominated Contingency (NDC) approach [8]. According to the latter a contingency k is *dominated* by contingency j if contingency j leads to a larger or equal violation for every constraint than contingency k, and a strictly larger violation for at least one constraint. Hence, a contingency is *non-dominated* if no other contingency dominates it.

The principle of the NDC contingency filtering approach is to select the non-dominated contingencies among the given set of *critical* contingencies (i.e. contingencies which lead to constraint violation). The reader is encouraged to refer to [8,9] for a detailed description of this filtering technique.

C. Network Compression Method

The network compression method [10] identifies, for each selected contingency, a limited area called the active region. The variables and elements of the active region are kept in

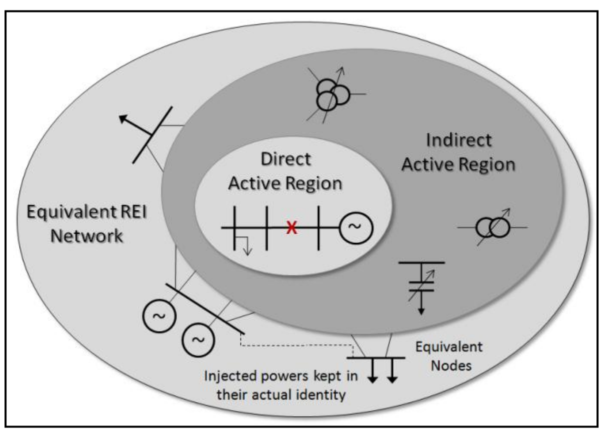

Fig. 1. Network Compression method concept

their real identity. The nodes and elements that do not belong to the active region are replaced by a REI-Dimo (Radial-Equivalent-Independent) equivalent network proposed by Paul Dimo [11].

The active region is composed of two sub-regions: the direct and the indirect regions (see Fig. 1).

The direct active region is defined as the set of buses and branches where the contingency has a significant impact in terms of voltages, angles, and power flow deviations with respect to base case values. Buses are selected as part of the direct active region if the variation of voltages, angles, or power flows between the pre-contingency state and post-contingency state, simulated by a power flow program, exceed chosen thresholds.

The indirect active region is motivated by the fact that there are control means located outside the direct active region that may significantly impact during the optimization process the constraints related to the elements selected in the direct active region. Once the direct active region has been identified, linear sensitivity analysis is performed to assess the impact of control means located outside the direct active region on the elements of the latter. Buses relative to those control means whose sensitivities exceed chosen thresholds are selected as part of the indirect active region.

The criteria and formulas used to set up the direct and indirect active region are described in details in [10,17].

It is important to notice that equipment that have been reduced by the Network Compression do not exist anymore in post-contingency situation. Therefore, constraints or control variables cannot be activated on such equipment as they have been replaced by the REI-Dimo equivalent.

III. ITERATIVE SCOPF WITH NETWORK COMPRESSION (ISCOPF-NC)

A. Overview of the Computational Modules Used by the ISCOPF-NC Approach

The flowchart in Figure 2 gives a global overview of the proposed ISCOPF-NC algorithm. An *iteration of the algorithm* represents the transition through the following modules:

1) A Load Flow (**LF**) computation module is used to obtain a reasonable starting point of the ISCOPF-NC algorithm.
2) A **SCOPF** module[1] is used to solve the main SCOPF problem (1). The latter includes only the contingencies selected by the filter and further compressed by the NC method. The module uses an interior-point method solver [12] which guarantees a local optimal solution.
3) A Security Analysis (**SA**) module checks by means of a conventional power flow program, whether, at the optimal solution provided by the SCOPF module, some contingencies lead to violations of constraints (branch flows or voltage limits). The so identified contingencies are called *critical* contingencies. If the set of critical contingencies is empty an acceptable solution of the SCOPF problem is found and computations terminate.
   Notice that because we use the network compression method within the SCOPF (see Section II.C), it may be possible that when running the SA module at the optimal solution provided by the SCOPF, some of the potentially binding contingencies (those already included in SCOPF) *still lead to small constraint violations.* If these violations are above a given tolerance, the compression factor used in the NC module (see Section II.C) will be reduced for the concerned contingencies for the subsequent loops.
4) A Contingency Filter (**CF**) module (see Section II.B) selects among the critical contingencies those that are candidates to be added to the current SCOPF problem. These contingencies are called *selected* contingencies.
5) A Post-Contingency OPF (**PCOPF**) module which, in the "corrective also" approach, keeps among the selected contingencies only those unable to reach a feasible post-contingency state within the allowed time.
   For this purpose, each selected contingency is assessed using the PCOPF module which minimizes the degree of relaxation of corrective actions to reach a feasible state. For contingency k, the PCOPF problem is formulated as:

$$\min \sum z_k$$
$$s.t.\begin{cases} g_k(x_k,u_k)=0 \\ h_k(x_k,u_k)\le 0 \\ \left|u_k-u_0^*-\Delta u_k\right| \le \tau\left[\dfrac{du_k}{d\tau}\right]_{\max}+z_k \\ z_k\ge 0 \end{cases} \quad (2)$$

   where u_0^* is the vector of base case optimal values of control variables (stemming from the optimal solution of the SCOPF problem at the current iteration) and z_k is the vector of positive slack variables aimed to relax coupling constraints. The other variables have the same meaning as in formulation (1).
   If the value of the PCOPF objective is nonzero, which means that the corrective actions are insufficient to ensure post-contingency state feasibility, the contingency is called *uncontrollable*. Otherwise, the contingency is called *controllable*. Only uncontrollable contingencies are added to the SCOPF problem.
   The uncontrollable contingencies that must be added to the SCOPF problem are called potentially binding contingencies. If the set of potentially binding contingencies is empty an acceptable solution of the SCOPF problem is found and computations terminate.
   Note that the PCOPF is not needed in the "preventive only" approach where all selected contingencies are obviously considered as potentially binding.

[1] Note that at the first iteration, the set of potentially binding contingencies is empty and the SCOPF is reduced to an OPF computation that includes only the pre-contingency constraints.

6) For each potentially binding contingency, an *optional* Network Compression (**NC**) module (see Section II.C) can be used in order to reduce as much as possible the size of the post-contingency power system model that has to be added to the current SCOPF problem. Note that the compression of previously added contingencies *is re-assessed every iteration*.
7) Although not explicitly shown in Fig. 2, there is also a module which updates of the SCOPF problem (i.e. it adds newly selected contingencies but also refreshes previously added contingencies).

B. Treatment of Discrete Variables

In order to provide meaningful SCOPF solutions, discrete variables (e.g. transformer ratio, phase shifter angle, and shunt reactive power) are treated by a progressive round-off technique [13].

In this technique all discrete variables are first treated as continuous variables. Then they are progressively rounded-off to their nearest discrete values. The SCOPF solution is re-assessed every time a set of discrete variables are fixed to discrete values.

This technique offers a good trade-off between accuracy and speed (e.g. compared to accurate but very computationally intensive techniques such as MINLP, or fast but less accurate techniques based on linear approximation).

IV. HIGH PERFORMANCE ENVIRONMENT

In this Section we describe the computational environment we used in our simulations, without claiming superiority or optimal choices compared to the state-of-the-art, but only to interpret the gains obtained in computational time.

Although the SA, the NC, and (in the "corrective also" mode) the PCOPF modules are used intensively during the ISCOPF-NC process, each contingency is addressed independently from the others in these modules:

1) The SA module receives the full list of contingencies but each of them is evaluated separately.
2) In corrective mode, contingencies selected by the CF lead to separate PCOPF sub-problems.
3) Potentially binding contingencies are reduced separately by the NC module.

This intrinsic property has been exploited in each module in order to reach computational times compatible with TSO (Transmission System Operator) operational requirements.

Two main approaches can be considered to deal with such kind of trivially parallelizable problems [22]:

- Either launching automatically N instances of the executable with different inputs mastered by a workload management system;
- Or introducing distributed memory parallelization directly in the executable code.

This latter approach has been chosen for the Security Assessment. The Contingency Filter, which computational effort is negligible compared to SA, is embedded in the Security Assessment tool, and exchanges information with the SA with as less as possible I/O interactions.

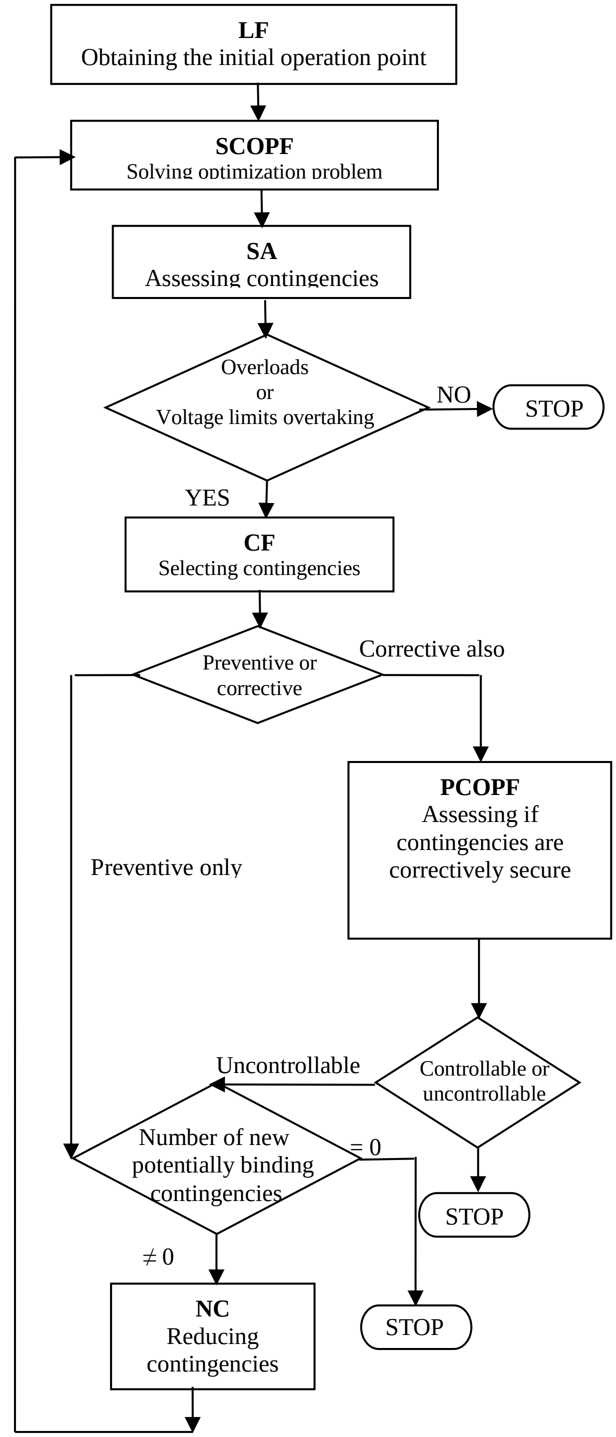

Fig. 2. ISCOPF-NC Flowchart

Using the Message Passing Interface (MPI), a master process controls all input data and output results operations and distributes tasks among slave processes, which perform computations and send results back to the master process, without any communication between them.

All data, representing a few Megabytes, are transferred through the network and read in memory by slave processes. Results of slave processes are stored in memory and transferred to the master process, which interacts directly with the CF without using the hard disk drive.

In other words, the hard disk drive is only accessed by the master process, and slave processes as well as the CF only access the physical memory.

For the Network Compression, the first approach has been chosen. Although MPI techniques could have been envisaged, as the NC uses and generates a lot of information (hence files), a Workload Management System (WMS) was found more appropriate in this case, as it manages efficiently the files transfer between distant working directories and the user local working directory.

Finally the SCOPF problems are solved in a sequential way and hence do not take advantage of the parallelization of the high performance environment.

V. Numerical results

A. Tests Cases Description

The ISCOPF-NC method has been tested on two systems: a large national network, similar in size to most grid models considered by the TSOs in ETN, and the European extra-high voltage network that stems from the model provided by the ENTSO-E for the purpose of the PEGASE project [15,16]. Details about the ETN test case can be found in [26], *the data being available on request* by contacting the authors of [26]. Table I summarizes the characteristics of both networks.

For the national network, we assume as contingency the loss of any equipment connected on voltage level of at least 150kV, which leads to 1297 contingencies.

For the European network, the set of postulated contingencies has been built choosing a large variety of equipment (e.g. generators, lines, transformers, shunts), which leads to approximately 12000 contingencies.

Referring to discussions of Section IV, it should be noticed that both problems have been solved using the parallel version of the ISCOPF-NC algorithm. However, this parallel version does not show significant improvements of computational speed on the "national network" system (less than 2%).

B. Computer Architecture

A BladeCenter of 8 blades with 2 processors of 4 cores (hence 8 cores per blade) with 2.6 Ghz clock rate, each blade sharing 12 GB of memory, has been used to perform the ISCOPF-NC algorithm on both test cases.

TABLE I
TEST NETWORKS CHARACTERISTICS

Number of	National network	European network
Busbars	2563	9241
Lines	1762	14044
Transformers	921	2217
Phase shifters	3	79
Generators	137	2060
Capacitor banks	91	322
Postulated contingencies	1297	12000

C. Detailed SCOPF Problem Description

We detail hereafter the abstract SCOPF problem formulation of Section II.

In order to test our approach in more stringent conditions, we consider the *joint optimization of both active and reactive powers, i.e. a full SCOPF*.

Minimum generation cost or the economic dispatch is a major objective function in SCOPF. However, since generators cost curves are confidential for both power system models used in the paper, instead of making arbitrary assumptions about these costs, we use a generic quadratic objective function, namely least squares of generators active power deviations with respect to their initial values, which can be seen as a particular case of a (pseudo) economic dispatch problem with classical quadratic cost functions for generators:

$$\min_{P_i^s, Q_i^s, \alpha_i^s, \varphi_{jk}^s, \rho_{jk}^s} \sum_{i \in G^0} (P_i^0 - Pinit_i^0)^2$$

subject to:

$$\sum_{i \in G_j^s} P_i^s - \sum_{i \in L_j^s} PL_i^s - \left(V_j^s\right)^2 Gjj + V_j^s \sum_{k \in N_j^s} (V_k^s/\rho_{jk}^s) \cos\left(\theta_j^s - \theta_k^s - \varphi_{jk}^s\right) G_{jk} + V_j^s \sum_{k \in N_j^s} (V_k^s/\rho_{jk}^s) \sin\left(\theta_j^s - \theta_k^s - \varphi_{jk}^s\right) B_{jk} = 0, \quad \forall j \in N^s, \forall s \in S \qquad (3)$$

$$\sum_{i \in G_j^s} Q_i^s - \sum_{i \in L_j^s} QL_i^s + \left(V_j^s\right)^2 Bjj + \sum_{i \in C^s} \alpha_i \left(V_j^s\right)^2 dBi + V_j^s \sum_{k \in N_j^s} (V_k^s/\rho_{jk}^s) \sin\left(\theta_j^s - \theta_k^s - \varphi_{jk}^s\right) G_{jk} + V_j^s \sum_{k \in N_j^s} (V_k^s/\rho_{jk}^s) \cos\left(\theta_j^s - \theta_k^s - \varphi_{jk}^s\right) B_{jk} = 0, \quad \forall j \in N^s, \forall s \in S \qquad (4)$$

$$F_{jk}^s\left(V_j^s, V_k^s, \theta_j^s, \theta_k^s\right) \le Fmax_{jk}^s, \forall jk \in B^s, \forall s \in S \qquad (5)$$
$$Vmin_j^s \le V_j^s \le Vmax_j^s, j \in N^s, \forall s \in S \qquad (6)$$
$$Pmin_i^s \le P_i^s \le Pmax_i^s, i \in G^s, \forall s \in S \qquad (7)$$
$$Qmin_i^s \le Q_i^s \le Qmax_i^s, i \in G^s, \forall s \in S \qquad (8)$$
$$\alpha_i^s \in \alpha_i = \{\alpha_{1i}^s, \alpha_{2i}^s, \dots\}, i \in C^s, \forall s \in S \qquad (9)$$
$$\varphi_{jk}^s \in \varphi_{jk} = \left\{\varphi_{1jk}^s, \varphi_{2jk}^s, \dots\right\}, jk \in PST^s, \forall s \in S \qquad (10)$$
$$\rho_{jk}^s \in \rho_{jk} = \left\{\rho_{1jk}^s, \rho_{2jk}^s, \dots\right\}, jk \in LTC^s, \forall s \in S \qquad (11)$$
$$\left|P_i^s - P_i^0\right| \le DP_i, i \in G^s, \forall s \in S\backslash\{0\} \qquad (12)$$
$$\left|\alpha_i^s - \alpha_i^0\right| \le D\alpha_i, i \in C^s, \forall s \in S\backslash\{0\} \qquad (13)$$
$$\left|\varphi_{jk}^s - \varphi_{jk}^0\right| \le D\varphi_{jk}, jk \in PST^s, \forall s \in S\backslash\{0\} \qquad (14)$$
$$\left|\rho_{jk}^s - \rho_{jk}^0\right| \le D\rho_{jk}, jk \in LTC^s, \forall s \in S\backslash\{0\} \qquad (15)$$

where:

- S is the set of system states, indexed by s, where 0 denotes the pre-contingency state and $1, \dots, n$ denote the n post-contingency states;
- N^s is the set of buses in state s;
- B^s is the set of branches in state s;
- G^s is the set of generators in state s;
- C^s is the set of shunt banks in state s;
- PST^s is the set of phase shifter transformers in state s;
- LTC^s is the set of on load tap changer transformers in state s;
- G_j^s is the set of generators connected to bus j in state s;
- L_j^s is the set of loads connected to bus j in state s;
- N_j^s is the set of nodes adjacent to bus j in state s;
- α_i is the set of discrete step positions of shunt bank i;
- φ_{jk} is the set of discrete angle values of phase shifter jk;
- ρ_{jk} is the set of discrete ratio values of LTC transformer jk;
- P_i^s and Q_i^s are respectively the active and reactive powers of generator i in state s;
- $Pinit_i^0$ is the initial active power of generator i in pre-contingency state;
- PL_i^s and QL_i^s are respectively the active and reactive powers of load i in state s;
- V_j^s and θ_j^s are the magnitude and phase angle of the voltage at bus j in state s;
- Gjj and Bjj are respectively the shunt conductance and susceptance at bus j of the branch jk (according to the quadruple classical branch model);
- G_{jk} and B_{jk} are the respectively the conductance and susceptance the branch linking bus j and bus k;
- dBi is the total susceptance of shunt bank i;
- α_i^s is the step position of shunt bank i in state s;
- φ_{jk}^s is the discrete phase angle value of phase shifter jk in state s;
- ρ_{jk}^s is the ratio of LTC transformer jk in state s (in particular $\rho_{jk}^s = 1$ if the branch jk is a line);
- $Pmin_i^s$ and $Pmax_i^s$ are respectively the bounds on active power of generator i in state s;
- $Qmin_i^s$ and $Qmax_i^s$ are respectively the bounds on reactive power of generator i in state s;
- F_{jk}^s is the apparent power flow in branch jk and $Fmax_{jk}^s$ is its MVA limit;
- $Vmin_j^s$ and $Vmax_j^s$ are respectively the bounds on voltage magnitude at node j in state s;
- DP_i is the maximal allowed rate of change of active power of generator i following a contingency;
- $D\alpha_i$ is the maximal allowed rate of change of step position of shunt i following a contingency;
- $D\varphi_{jk}$ is the maximal allowed rate of change of the angle of phase shifter jk following a contingency;
- $D\rho_{jk}$ is the maximal allowed rate of change of the ratio of LTC transformer jk following a contingency.

The control variables of the problem are: generators active and reactive powers (P_i^s and Q_i^s), step position of capacitor banks (α_i^s), angle of phase shifters (φ_{jk}^s), and ratio of transformers (ρ_{jk}^s). Note that the three latter types of control variables are properly modeled as *discrete variables* in Eq. (3)-(15).

The equality constraints are the power flow equations (3)-(4). The inequality constraints are the apparent power flow limits on every branch (5), voltage limits (6), bounds on the control variables (7)-(11), and coupling constraints between pre-contingency state and post-contingency state (12)-(15).

The SCOPF problem is solved hereafter in "preventive only" mode. The Contingency Filter uses the Non Domination criterion and focus only on flows violations.

D. SCOPF Problems Size

Table II summarizes the size of the initial OPF network and gives a rough estimation of the corresponding SCOPF size if a full model of the post-contingency states is considered.

TABLE II
SIZES OF OPF/SCOPF PROBLEMS

	National network		European network	
	OPF	SCOPF	OPF	SCOPF
Variables	4960	6,300,000	23096	277,000,000
Inequalities	3786	5,000,000	32098	385,000,000
Equalities	4910	6,300,000	18482	220,000,000

Such huge size SCOPF problems are unmanageable directly with nowadays computational tools.

For the national network, it is however possible to obtain a SCOPF problem of manageable size with full representation of post-contingency states by using the ISCOPF-NC algorithm without performing the network compression step, as shown hereafter.

E. Results for the National Network

The solution obtained with the ISCOPF-NC algorithm is called hereafter as "Compressed", whereas its variant obtained without performing the network compression step i.e. using the full representation of the selected post-contingency states is called "Full".

The main results of the optimization process are summarized in the Table III.

Computational times refer to the SCOPF problem in Section V.C., namely the quadratic objective function on the deviation of active power.

The compressed case is, as expected, the fastest. Furthermore, in this case it leads to a better objective value than the reference full case but at the expense of some slight constraints violation that will be discussed in the next subsection and shown in Table VI.

TABLE III

RESULTS ON NATIONAL NETWORK WITH BOTH APPROACHES

		Full	Compressed
Number of outer SCOPF iterations		4	4
Final SCOPF size	Variables	77993	19444
	Equalities	75706	18190
	Inequalities	98154	22496
	Contingencies	19	20
f* (MW^2)	$\sum_{i\in G^0}(P_i^0 - Pinit_i^0)^2$	99546	71755
f* (MW)	$\sum_{i\in G^0}\lvert P_i^0 - Pinit_i^0\rvert$	1753	1168
Computation time		37 minutes	14 minutes

Table IV summarizes the number of contingencies added to the SCOPF at each outer loop of the algorithm. As the very first OPF is the same for each process, the first 13 selected contingencies are the same in each case, since the Security Assessment starts from the same optimal solution and the Contingency Filter uses the same input in the two cases. Both approaches identified the 3 binding and the 7 nearly binding contingencies at the optimum, among the 19 and respectively 20 contingencies included in the SCOPF. This proves that the CF performed satisfactorily.

TABLE IV

NUMBER OF CONTINGENCIES ADDED TO THE SCOPF PROBLEMS

Iteration	Full	Compressed
1	13	13
2	5	6
3	1	1
4	-	-
Total	19	20

We noticed that the final sets of contingencies selected by the full and compressed cases are slightly different. However, no more than 22 different contingencies have been selected overall in the two cases. Table V establishes the relationship between these 22 contingencies. In particular one can observe that 17 contingencies appear in both cases.

Table VI shows the flows on both sides of the branches (labeled 1->2 and 2->1) that lead to binding contingencies of the full case. They have been computed by performing a load-flow-based security analysis at the SCOPF solution of the selected cases.

TABLE V

CONTINGENCIES ADDED TO THE SCOPF PROBLEMS (DETAILS)

Contingency Number	Full	Compressed
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	-
15	15	15
16	16	16
17	17	-
18	18	18
19	19	19
20	-	20
21	-	21
22	-	22

One can remark that the flows on the branches with (nearly) active constraints are close and coherent. Some small overloads (less than the chosen 2% tolerance) appear for the compressed variant, as assumed in order to significantly speed-up computations and/or in larger cases to enable the inclusion of relevant contingencies into SCOPF.

Figure 3 presents the values of the active power of the generators obtained in these two cases. It can be observed that the general trend of the objective function is respected. The difference between both solutions in terms of generators active power is of 11.4 MW in average with a maximum value of 39.6 MW. Furthermore, 57 generators out of 98 have a deviation of less than 10 MW. Hence, although final values of active powers are slightly different, this approximated solution can be deemed acceptable especially given the number of generators involved in optimization and their final output.

TABLE VI
POWER FLOWS AT FINAL SCOPF SOLUTION

Contingency number	Limit (MVA)	Full		Compressed	
		1->2	2->1	1->2	2->1
2	1350	1348.9	1335.0	1351.4	1340.2
4	39	37.7	38.8	38.5	39.8
5	56	56.0	54.4	56.6	54.9
7	1351	1346.3	1347.4	1349.5	1350.6
8	40	40.0	38.8	38.6	38.3
10	40	39.9	39.8	40.0	39.8
11	20	19.8	19.8	19.8	19.8
13	345	341.0	343.8	348.7	347.0
15	1351	1345.2	1346.3	1349.0	1350.3
18	73	72.5	70.1	68.7	66.5

We can conclude that the trade-off between accuracy and speed due to the ISCOPF-NC method approximation is acceptable as it allows reducing significantly the computational time offering a national system operator the possibility to use the algorithm not only in day-ahead but also in intra-day and close to real-time operation contexts.

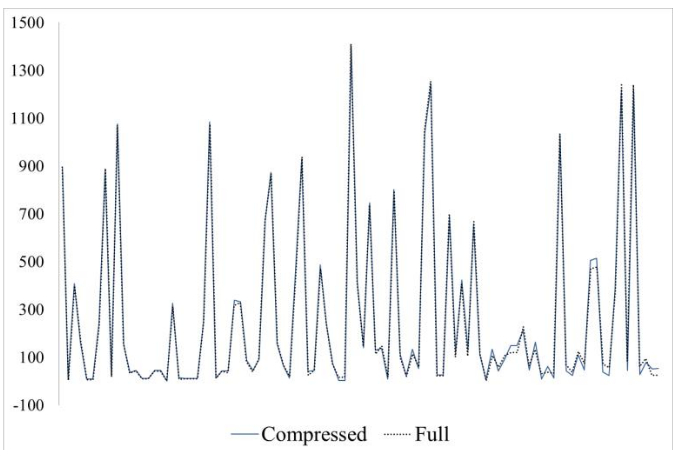

Fig. 3. Generators active powers (MW) in both full case (dashed line) and compressed case (continuous line).

Although small violations may occur, they are only allowed in post-contingency states, where network reduction applies, but not in the pre-contingency state which is modeled using the whole grid model. This is a reasonable controllable risk since, if a contingency really occurs its effects can be also counteracted by other last resort control means (e.g. load shedding).

The algorithm offers the possibility to set the tolerance accepted on post-contingency constraints violations. In particular, if one looks for a solution without violations of post-contingency constraints then the full model of the contingency states is used, at the expense of larger computational time, and provided that memory requirements are met. The algorithm has been devised to provide flexible choices to satisfy the trade-off between (post-contingency) constraints satisfaction accuracy and computational speed, within available computer memory limit.

For this grid we observed that the parallelized ISCOPF-NC does not bring significant gains in terms of performances. Specifically, for the MPI within the SA, a rough overall gain of 3 minutes is observed. Each SA decreases its computation time from 60s to 15s, which is under the gain expected considering the number of computation cores used. This can be explained by the latency due to the repartition of work among computation cores, latency which is not recovered by gains of SA calculations on this relatively low number of contingencies.

Using the WMS for managing the Network Compression brings even worse results as performances decrease with respect to the sequential execution of the ISCOPF. In addition to the latency related to the repartition of work, the transfer of input and output files to and from distant hard disk drives slows down the overall process. The Network Compression which takes about 20s in its sequential form falls to a rough 40s when managed by the WMS.

Further details about this test case and further SCOPF results for this system can be found in [23].

F. Results for the European Network

First, we briefly illustrate the identification of the active region for two critical contingencies using the 9241-bus model of the European Network described in Section V.A.

The loss of a critical line results in a direct active region of 291 buses selected as follows: 69 buses according to the voltage criterion, 19 according to the angle criterion, and 204 according to the power flow criterion. The indirect active region contains 14 buses, 2 buses according to the voltage criterion and 12 according to power flow criterion. They stem from 2 phase shifters and 12 on load tap changing transformers which may have a significant impact on the elements of the direct active region. The total number of nodes kept in the active region is therefore 305, counting for a compression factor of 96.7%. Furthermore, the REI-Dimo equivalent generates 52 equivalent nodes and 229 equivalent lines.

The loss of another critical line has comparatively a more local impact as the direct active region comprises only 34 buses. However, the indirect active region is comparatively larger as it contains 21 additional buses stemming from a shunt bank capacitor and 20 on-load tap changing transformers. The total number of active region nodes is 55, which counts for a compression factor of 99.4%. The REI-Dimo equivalent generates additionally 53 equivalent nodes and 179 equivalent lines.

We now present results of the ISCOPF-NC process on this ETN test case.

Tables VII and VIII provide the number of variables, constraints, contingencies in the SCOPF problem, the CPU time for the major processes of the ISCOPF-NC loop, and the contingencies selected by the filtering process.

These results show that 57% of the ISCOPF-NC process is spent solving the SCOPF problem. Note that the SCOPF module itself deals with *both continuous and discrete variables*, as explained in Section III.B, and ends up at every outer loop to *computing three SCOPF problems of different sizes* and with different continuous relaxations as follows. The first SCOPF considers all variables as continuous, then shunt reactive powers are set to discrete values and the remaining variables are optimized, next the transformers ratio and phase shifter angle are set to discrete values and finally the continuous variables are optimized.

TABLE VII
STATISTICS OF THE ISCOPF-NC ALGORITHM

Iteration	Variables	Constraints	Computation time (s)		
			SCOPF	SA	NC
1	23000	50000	70	130	60
2	30000	64000	485	130	140
3	33000	70000	940	130	140
4	34000	72000	710	130	0
			2205 (57%)	520 (13%)	340 (9%)
			Overall process: 65 min		

TABLE VIII
CONTINGENCIES INCLUDED IN THE SCOPF

Iteration	Contingencies in SCOPF	Critical contingencies	Selected Contingencies
1	0	30	23
2	23	23	14
3	37	3	3
4	40	0	0

The security analysis and the network compression take respectively 13% and 9% of the overall process time, thanks to the parallel computations. Finally, 21% of time is spent in refreshing and updating the SCOPF problem.

This example clearly demonstrates that our approach allows solving SCOPF problems in large systems with a large number of postulated contingencies within acceptable computational times.

Given a 2% tolerance on overloads, no branch appears to be overloaded when performing Security Analysis at the final solution on the whole set of contingencies.

The active constraints at the final solution show that 23 contingencies out of 40 are binding, proving again the good performance of the contingency filter.

Note that a SCOPF problem including the full model of the 23 binding contingencies on the ETN network is intractable with the current optimization tools and computation architecture because the very large size of the problem exceeds memory limit. The network compression remains thus essential as it allows reducing the number of variables and constraints related to the post-contingency states so as to cope with the memory limit.

VI. Conclusions and Future Works

This paper has demonstrated that using appropriate dedicated computing architectures our method allows solving SCOPF problems, which rely on the accurate AC network model and treat discrete variables, in large-scale systems within acceptable computational times.

The proposed algorithm can be used in the context of day-ahead operational planning in large scale systems comparable in size with the European network. The algorithm can even be used in intraday or close to real-time for systems of medium size such as the national network.

Results obtained with the prototype implementing the ISCOPF-NC algorithm have shown that the major gains in performances could be expected by improving the resolution of the SCOPF problem. Gains can also be expected by accelerating the update of the SCOPF problem definition but this is related to a better implementation of the prototype, for instance by minimizing files exchanges.

Note that, in the absence of knowledge about the grid to optimize, our algorithm starts “from scratch.” However, faster convergence is expected if system operator knowledge about binding/critical contingencies is used, the algorithm being more likely to converge in one-two loops. Also real-world SCOPF problems generally model only relevant system thermal constraints for each contingency which further reduces the number of constraints. Likewise, as in real-time the operating state is not likely to differ drastically from one time-window to the other, the ISCOPF-NC algorithm can be accelerated by using the information of the previous solution (e.g. the solution itself, the list of binding contingencies and binding or nearly binding constraints) to start quicker than “from scratch” assumptions.

Further speed-up is expected for customized decomposed SCOPF problems (e.g. solving only the active power sub-problem or the reactive power sub-problem) and if discrete variables are not included in optimization or their continuous relaxation is deemed acceptable.

In particular, the solution of the OPF mode, which optimizes a single system state at a time (e.g. to remove thermal congestions or severe voltage violations), including both continuous and discrete variables meets the stringent real-time requirements.

Although the progressive round-off strategy of discrete variables provides acceptable results, as the efficient treatment of discrete variables is a major challenge in SCOPF [3,4], we plan future work to evaluate the performances of other existing alternative approaches [27] for large-scale problems.

Further work is also planned to develop a scalable method to pre-select a limited number of efficient corrective actions to be used in the SCOPF “corrective also” mode [19].

ACKNOWLEDGMENT

The authors acknowledge the support of the European FP7 project PEGASE. They also acknowledge Sylvain Leclerc, Fortunato Villella, Vincent Sermanson and Ariadne Szekut for preparing optimization data and problems referred in this paper.

Ludovic Platbrood acknowledges Francois-Xavier Bouchez for his precious help in developing the parallel version of the ISCOPF-NC algorithm.

Florin Capitanescu has been supported for this work by the National Research Fund Luxembourg in the frame of the CORE project ``Reliable and efficient distributed electricity generation in smart grids'', C11/SR/1278568.

Louis Wehenkel acknowledges the support of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Program, initiated by the Belgian State, Science Policy Office.

The scientific responsibility rests with the authors.

REFERENCES

[1] O. Alsac and B. Stott, "Optimal load flow with steady-state security", IEEE Transactions on PAS, vol. PAS-93, no. 3, 1974, pp. 745-751.

[2] A.J. Monticelli, M.V.P. Pereira, and S. Granville, "Security-constrained optimal power flow with post-contingency corrective rescheduling", IEEE Transactions on Power Systems, vol. PWRS-2, no. 1, February 1987, pp. 175-182.

[3] F. Capitanescu, J.L. Martinez Ramos, P. Panciatici, D. Kirschen, A. Marano Marcolini, L. Platbrood, and L. Wehenkel, "State-of-the-art, challenges, and future trends in security-constrained optimal power flow", Electric Power System Research, vol. 81, no. 8, pp 1731-1741, 2011.

[4] B. Stott, O. Alsac, Optimal power flow - basic requirements for real-life problems and their solutions (White Paper), SEPOPE XII Symposium, Rio de Janeiro, Brazil, May 2012, available online.

[5] T.J. Bertram, K.D. Demaree, and L.C. Dangelmaier, "An integrated package for real-time security enhancement", IEEE Transactions on Power Systems, vol. 5, no. 2, 1990, pp. 592-600.

[6] O. Alsac, J. Bright, M. Prais, and B. Stott, "Further developments in LP-based optimal power flow"", IEEE Transactions on Power Systems, vol. 5, no. 3, 1990, pp. 697-711.

[7] T.J. Overbye, Xu Cheng, and Yan Sun, "A comparison of the AC and DC power flow models for LMP calculations", Proc. of the 37th Annual HICSS conference, Hawaii, 2004.

[8] F. Capitanescu, M. Glavic, D. Ernst and L. Wehenkel, "Contingency filtering techniques for preventive security-constrained optimal power flow", IEEE Transactions on Power Systems, vol. 22, no. 4, 2007, pp. 1690-1697.

[9] F. Capitanescu and L. Wehenkel, "A New Iterative Approach to the Corrective Security-Constrained Optimal Power Flow Problem", IEEE Transactions on Power Systems, Vol. 23, No. 4, Nov. 2008, pp. 1533-1541.

[10] K. Karoui, H. Crisciu, A. Szekut, and M.Stubbe, "Large Scale Security Constrained Optimal Power Flow", 16th Power System Computation Conference, Glasgow Scotland, 14-18 July, 2008.

[11] P. Dimo, "Nodal analysis of power systems", Abacus Press, 1975, Tunbridge Wells, Kent, England.

[12] R. H. Byrd, M. E. Hribar, and J. Nocedal, "An interior point algorithm for large-scale non-linear programming", SIAM Journ. Optimization, vol. 9, pp. 877-900, Apr. 1999.

[13] K. Karoui, L. Platbrood, H. Crisciu and R. Waltz: "New Large-scale Security Constrained Optimal Power Flow program using a new Interior point algorithm", 2008, 5th International Conference on the European Electricity Market.

[14] L. Lasdon, "Optimization theory for large systems", The Mac-millan Company, 1970, London.

[15] M. Stubbe, K. Karoui, T. Van Cutsem, and L. Wehenkel "Le projet PEGASE", Revue E tijdschrift, (4), pp.37-41, Dec. 2008.

[16] European FP7 project PEGASE: available on-line at: http://www.fp7-pegase.com/.

[17] L. Platbrood, H. Crisciu, F. Capitanescu, L. Wehenkel, "Solving very large scale SCOPF problems by combining iterative algorithms and network compression", PSCC conference, Sweden, August 2011.

[18] P.J. Macfie, G.A. Taylor, M.R. Irving, P. Hurlock, Wan Hai-Bin, "Proposed Shunt Rounding Technique for Large-Scale Security Constrained Loss Minimization" ,IEEE Trans. Power Syst., vol. 25, no. 3, 2010, pp. 1478-1485.

[19] A. Marano Marcolini, F. Capitanescu, J.L. Martinez Ramos, L. Wehenkel, "Exploiting the use of DC SCOPF approximation to improve iterative AC SCOPF algorithms", IEEE Trans. Power Syst., vol. 27, no. 3, 2012, pp. 1459-1466.

[20] R. Jabr, N. Martins, B. Pal, S. Karaki, "Contingency Constrained VAr Planning Using Penalty Successive Conic Programming", IEEE Trans. Power Syst., vol. 27, no. 1, 2012, pp. 545-553.

[21] F. Capitanescu and L. Wehenkel, "Experiments with the interior-point method for solving large scale optimal power flow problems", Electric Power System Research, vol. 95., no. 2, February 2013, pp. 276-283.

[22] F.-X. Bouchez, B. Haut, L. Platbrood, K. Karoui, "HPC for power systems in the framework of the PEGASE project", IEEE PES GM, USA, July 2012.

[23] L. Platbrood, S. Fliscounakis, F. Capitanescu, P. Panciatici, C. Merckx, and M. Ortega-Vazquez, "Deliverable D3.2: Development of prototype software for steady state optimisation of the ETN ", PEGASE project, http://www.fp7-pegase.com/, 2011.

[24] S.Leclerc (task leader), "Deliverable 6.3 – Part 2 Demonstration report for Steady State Optimization", PEGASE project, http://www.fp7-pegase.com/, 2012.

[25] R. Waltz, T. Platenga, Knitro 6.0 user's manual, 2009.

[26] F.Villella, S.Leclerc, I.Erlich and S.Rapoport, "PEGASE Pan-European Test-beds for testing of Algorithms on Very Large Scale Power Systems", IEEE PES ISGT, Germany, July 2012.

[27] F. Capitanescu and L. Wehenkel, "Sensitivity-based approaches for handling discrete variables in optimal power flow computations", IEEE Transactions on Power Systems, Vol. 25, No. 4, November 2010, pp. 1780-1789.

Ludovic Platbrood was born in Belgium in 1977. He graduated as mathematician from the University of Namur, Belgium. After obtaining his Master degree in Operational Research and Applied Mathematics in 2002, he joined the Power System Consulting group of Tractebel Engineering where he is Design Engineer. His main areas of interest are optimization, numerical analysis and numerical algebra.

Florin Capitanescu was born in Romania in 1973. He graduated in Electrical Power Engineering from the University ``Politehnica'' of Bucharest in 1997. He obtained the Ph.D. degree from the University of Liège in 2003. His main research interests include optimization methods application in power systems, in particular optimal power flow, and voltage stability.

Christian Merckx graduated as mathematician in 1982 and received the Ph.D. degree in 1990, both from the University of Namur (Belgium). Since, he works for the Power System Consulting group of Tractebel Engineering in Belgium where he is coordinator of the software development. His work is focused on application of numerical analysis, optimization and computer sciences in the field of power systems.

Horia Crisciu was born in Romania in 1945. He graduated from the Polytechnic University of Bucharest, Romania. After obtaining his PhD degree in Electrical Engineering in 1982, he joined Polytechnic University of Bucharest as Professor. Since 1989, he joined the Power System Consulting group of Tractebel Engineering as Principal Engineer. His main areas of interest are power system optimization, state estimation, and EMS algorithmic.

Louis Wehenkel graduated in Electrical Engineering (Electronics) in 1986 and received the Ph.D. degree in 1990, both from the University of Liège, where he is full Professor of Electrical Engineering and Computer Science. His research interests lie in the fields of stochastic methods for systems and modeling, optimization, machine learning and data mining, with applications in complex systems, in particular large scale power systems planning, operation and control, industrial process control, bioinformatics and computer vision.