A New Iterative Approach to the Corrective Security-Constrained Optimal Power Flow Problem

Florin Capitanescu and Louis Wehenkel, *Member, IEEE*

Abstract—This paper deals with techniques to solve the corrective security-constrained optimal power flow (CSCOPF) problem. To this end, we propose a new iterative approach that comprises four modules: a CSCOPF which considers only a subset of potentially binding contingencies among the postulated contingencies, a (steady-state) security analysis (SSSA), a contingency filtering (CF) technique, and an OPF variant to check post-contingency state feasibility when taking into account post-contingency corrective actions. We compare performances of our approach and its possible variants with classical CSCOPF approaches such as the direct approach and Benders decomposition (BD), on three systems of 60, 118, and 1203 buses.

Index Terms—Benders decomposition, contingency filtering, optimal power flow, security-constrained optimal power flow.

I. INTRODUCTION

THE security-constrained optimal power flow (SCOPF) problem is a nonlinear, nonconvex, large-scale optimization problem [1], [2]. The SCOPF has been formulated under two modes: "preventive" [1] and "corrective" [2], called hereafter PSCOPF and CSCOPF, respectively. In this paper we focus on the CSCOPF which, unlike the PSCOPF, considers the possibility of rescheduling control means in post-contingency states, other than those with automatic response to contingencies (e.g., active power of generators participating in frequency control, automatic tap-changers, capacitor/reactor bank switching, secondary voltage control, etc.). The underlying assumption of CSCOPF approach is that operational limits violation (e.g., power flows, bus voltages, etc.) can be generally endured up to (at least) several minutes without damaging the corresponding equipment, which lets some time for (automatic or human) corrective actions to be implemented. The major difficulty of the SCOPF problem is its high dimensionality, especially for large systems and/or when many contingencies are considered [3]–[5]. Trying to solve this problem directly for a large power system, by imposing simultaneously all post-contingency constraints, would lead to prohibitive memory and CPU times requirements. Moreover, because in real life applications most contingencies do not constrain the optimum, including them all into the SCOPF problem increases the complexity of the computations by shrinking the feasible region, and can lead to

algorithmic/numerical problems. This is especially true under stressed operating conditions, i.e., when the SCOPF solution is most useful.

Alternatively, since the seminal paper [2], (generalized) Benders decomposition (BD) [6], [7] has been widely used to solve various CSCOPF problems, such as: optimization of operation cost, reactive power planning, computation of available transfer capability, etc. [2], [8]–[15]. The CSCOPF problem is considered either as such [2], [8]–[12], or is embedded in a more general formulation such as generators unit commitment [13]–[15]. In the context of the unit commitment problem most CSCOPF approaches use simplified linear (dc) formulation, in order to reduce problem complexity, except for [15]. An exhaustive list of generalized BD applications in power systems is provided in [16].

BD approach consists of decomposing the original CSCOPF problem into a master problem and several slave subproblems which interact iteratively. It is very appealing due to the possibility to keep the size of master and slave problems very tractable (almost the same as optimizing a system pre- or post-contingency state only) as well as to distribute computations among several processors, which can considerably speed-up computations [2]. On the other hand, BD requires (theoretically) the convexity of the feasible region which can not be guaranteed in the ac CSCOPF model, and consequently it is recommended to be used with care [2], [9].

To mitigate the drawbacks of these two approaches we propose instead a new Iterative CSCOPF approach (ICSCOPF), which comprises four modules: a standard CSCOPF module applied to a small subset of potentially binding contingencies, a (steady-state) security analysis (SSSA) module, a contingency filtering (CF) module, and an OPF module to check the "controllability" of post-contingency states. This approach essentially aims to efficiently identify an as small as possible superset of the binding contingencies at the CSCOPF optimum. We call *binding* a contingency which leads to active post-contingency constraints, different than in the base case, related to branch currents and/or voltage magnitudes. Assuming that the CSCOPF problem is feasible, the set of binding contingencies is the smallest subset of the full postulated contingency set which provides the same optimal objective value as the full set. We provide extensive simulations results on three test systems of up to 1203 buses, with three different filtering schemes, and comparisons with Benders decomposition.

The rest of the paper is organized as follows. Section II introduces the CSCOPF problem. Section III presents the ICSCOPF approach and its variants. Section IV provides numerical comparisons with competing approaches for the solution of two CSCOPF problems. Conclusions are drawn in Section V. Details concerning the application of Benders decomposition are collected in the Appendices.

II. CORRECTIVE SECURITY-CONSTRAINED OPTIMAL POWER FLOW PROBLEM

The CSCOPF problem can be formulated as follows [2]:

$$\min_{\mathbf{x}_0,\ldots,\mathbf{x}_c,\mathbf{u}_0,\ldots,\mathbf{u}_c} \quad f_0(\mathbf{x}_0,\mathbf{u}_0) \tag{1}$$

$$\text{s.t.} \quad \mathbf{g}_k(\mathbf{x}_k,\mathbf{u}_k) = \mathbf{0} \quad k = 0,\ldots,c \tag{2}$$

$$\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k) \leq \mathbf{0} \quad k = 0,\ldots,c \tag{3}$$

$$|\mathbf{u}_k - \mathbf{u}_0| \leq \overline{\Delta\mathbf{u}_k} \quad k = 1,\ldots,c \tag{4}$$

where f_0 is the objective function, and for the kth system configuration ($k = 0$ corresponds to the pre-contingency configuration, while $k = 1,\ldots,c$ correspond to the c post-contingency configurations), $\mathbf{x}_k$ is the vector of state variables (i.e., real and imaginary part of voltage at all buses), $\mathbf{u}_k$ is the vector of control variables (e.g., generators active power, generators voltage, controllable transformer ratios, shunt element reactances, phase shifters angle, etc.), $\overline{\Delta\mathbf{u}_k} = T_k d\mathbf{u}_k/dt$ is the vector of maximal allowed variation of control variables between the base case and kth post-contingency state, T_k is the assumed time horizon allowed for corrective actions to ensure post-contingency state feasibility and $d\mathbf{u}_k/dt$ is the rate of change of control variables in response to contingency.

Constraints (2) and (3) impose the feasibility of the pre-contingency and corrected post-contingency states. Equality constraints (2) are essentially the ac bus power balance equations, while the inequality constraints (3) include physical limits of equipments (e.g., bounds on: generators active/reactive powers, controllable transformers ratio, shunts reactance, etc.) and operational limits (e.g., branch currents and voltage magnitudes). Inequalities (4) are "coupling" constraints aimed to prevent unrealistic variations of control variables between the base case and post-contingency states.

III. ITERATIVE APPROACH TO THE CSCOPF SOLUTION

A. Rationale of the ICSCOPF Approach

Let $\mathbf{P}_0^\star$ be an optimal operating point (computed by an OPF or a CSCOPF considering only a subset of the c postulated contingencies), and $\mathbf{u}_0^\star$ be its corresponding vector of optimal controls settings. Clearly, $\mathbf{u}_0^\star$ is also the optimal solution of the CSCOPF problem (1)–(4) if for any contingency $k = 1,\ldots,c$ there exist corrective controls $\mathbf{u}_k$ satisfying (4) and ensuring the feasibility of post-contingency state $\mathbf{x}_k$. We may check this by solving for every contingency k the following post-contingency optimal power flow (PCOPF) problem:

$$\min_{\mathbf{x}_k,\mathbf{u}_k,\mathbf{z}_k} \quad \mathbf{e}^T\mathbf{z}_k = y_k^\star \tag{5}$$

$$\text{s.t.} \quad \mathbf{g}_k(\mathbf{x}_k,\mathbf{u}_k) = \mathbf{0} \tag{6}$$

$$\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k) \leq \mathbf{0} \tag{7}$$

$$|\mathbf{u}_k - \mathbf{u}_0^\star| \leq \overline{\Delta\mathbf{u}_k} + \mathbf{z}_k \tag{8}$$

$$\mathbf{z}_k \geq \mathbf{0} \tag{9}$$

where $\mathbf{e}$ is a vector of ones of appropriate dimension and $\mathbf{z}_k$ is a vector of (positive) slack variables aimed to relax the coupling constraints.

The objective of this PCOPF problem is to minimize the degree of post-contingency infeasibility, measured by the violations of the coupling constraints (8). The contingency k is called controllable [14] if $y_k^\star = 0$ (or equivalently, if $\mathbf{z}_k^\star = \mathbf{0}$), meaning that a feasible post-contingency state may be reached thanks to

available corrective actions. Conversely, we will call the kth contingency uncontrollable if $y_k^\star > 0$ (or equivalently, if at least one component of $\mathbf{z}_k^\star$ is strictly positive), which implies that with the given $\mathbf{u}_0^\star$ no feasible post-contingency state may be reached with available corrective controls.

The conjecture behind the ICSCOPF approach is that in order to identify the binding contingencies at the CSCOPF optimum, it is sufficient to include only uncontrollable contingencies in the CSCOPF problem. Furthermore, to speed-up computations, we propose to use a contingency filter to identify (ideally) all uncontrollable contingencies, while introducing as few as possible controllable contingencies.

B. Algorithm of the ICSCOPF Approach

The basic algorithm of the ICSCOPF approach is as follows.
1) Let $\mathbf{P}_0$ be the operating point to be optimized and $\mathcal{C} = \{1,\ldots,c\}$ the contingency set with respect to which the system must be secure when post-contingency corrective actions are taken into account. Set the potentially binding contingency subset $\mathcal{C}_b = \emptyset$.
2) Solve the CSCOPF by including, beside base case constraints, only the post-contingency constraints for the subset $\mathcal{C}_b$. Let $\mathbf{P}_{\mathcal{C}_b}^\star$ be the optimal operating point.
3) Simulate each contingency in $\mathcal{C} \setminus \mathcal{C}_b$ at $\mathbf{P}_{\mathcal{C}_b}^\star$ by a classical load flow program. If none of them leads to constraint violations, $\mathbf{P}_{\mathcal{C}_b}^\star$ is the locally secure optimal solution and the computation terminates. Otherwise, let $\mathcal{C}_c$ be the subset of critical contingencies (i.e., those leading to some constraint violations).
4) Filter the contingencies from the $\mathcal{C}_c$ subset at $\mathbf{P}_{\mathcal{C}_b}^\star$. Let $\mathcal{C}_s \subseteq \mathcal{C}_c$ be the subset of selected contingencies.
5) Check the post-contingency state feasibility of contingencies from $\mathcal{C}_s$ at $\mathbf{P}_{\mathcal{C}_b}^\star$ by PCOPF, described in Section III-A. Let $\mathcal{C}_u \subseteq \mathcal{C}_s$ be the subset of uncontrollable contingencies. If $\mathcal{C}_u \neq \emptyset$, set $\mathcal{C}_b \leftarrow \mathcal{C}_b \cup \mathcal{C}_u$ and go to step 2.
6) Check the post-contingency feasibility of contingencies from $\mathcal{C}_c \setminus \mathcal{C}_s$ at $\mathbf{P}_{\mathcal{C}_b}^\star$ by PCOPF. Let $\mathcal{C}_u$ be the subset of uncontrollable contingencies. If $\mathcal{C}_u = \emptyset$, then $\mathbf{P}_{\mathcal{C}_b}^\star$ is the locally secure optimal solution and the computation terminates. Otherwise, set $\mathcal{C}_b \leftarrow \mathcal{C}_b \cup \mathcal{C}_u$ and go to step 2.

Observe that, since initially the current subset of potentially binding contingencies is empty ($\mathcal{C}_b = \emptyset$), the first CSCOPF call is in fact an OPF computation, which contains only base case constraints. The main advantage of using the first OPF computation is that, if the resulting operating point turns out to be secure with respect to the contingency set $\mathcal{C}$, there is no need to iterate on CSCOPF computations. Additionally, binding contingencies can be sooner revealed by the subsequent filtering techniques, especially if the OPF outcome is close to the sought security-constrained optimum.

We solve the CSCOPF and PCOPF problems intervening at steps 2, 5 and 6 by the interior-point method [17].

Step 3 of the algorithm is a (reduced) SSSA. The simulation of the system response to contingencies at this step is performed by a classical full ac load flow computation software. For an optimal base case $\mathbf{P}_{\mathcal{C}_b}^\star$, at some stage, a contingency is called critical (it belongs to subset $\mathcal{C}_c$) if it leads to post-contingency constraint violations or post-contingency load flow divergence. Otherwise, it is called noncritical. Note that, since classical load flow software does not take into account some time-varying control actions (e.g., generation rescheduling, shunt reactance

change, etc.), the SSSA alone can not decide whether the current optimal base case $\mathbf{P}^\star_{\mathcal{C}_b}$ is the optimum of the CSCOPF if some contingencies from the subset $\mathcal{C} \setminus \mathcal{C}_b$ violate constraints. Indeed, if a such contingency leads to constraint violation it may be possible that the post-contingency feasibility is ensured without altering the optimal base case, thanks to the computation of post-contingency corrective actions by the CSCOPF. The aim of using the SSSA is thus two-fold: to filter out noncritical contingencies (subset $\mathcal{C} \setminus \mathcal{C}_b \setminus \mathcal{C}_c$) as well as to serve as a basis for the contingency filtering performed at step 4.

Step 4 performs a contingency filtering with the aim to further reduce the number of contingencies that are treated at the subsequent steps. Any efficient contingency filtering technique can be used to this purpose. In this work, unless otherwise specified, we use the nondominated contingency (NDC) approach motivated by the good results obtained with it in the context of PSCOPF [18]. Additional filtering results will also be provided by means of the classical severity-index-based contingency ranking technique [1], [3]. These two techniques are described in Section III-C. The *selected* (respectively *discarded*) contingencies by the CF technique at the current iteration form the subset $\mathcal{C}_s$ (respectively $\mathcal{C}_c \setminus \mathcal{C}_s$).

Steps 5 and 6 check, by means of PCOPF problems, whether the contingencies from the subsets $\mathcal{C}_s$ or $\mathcal{C}_c \setminus \mathcal{C}_s$ lead to constraints violation that can not be removed by post-contingency controls only, and filter out controllable contingencies.

In the context of this iterative algorithm, we can guarantee that the full CSCOPF optimal solution has been reached, when all contingencies that are not yet included in the CSCOPF are noncritical (see step 3), or when the PCOPF check shows that all critical contingencies turn out to be also controllable.

Observe that, in our approach, the set of contingencies $\mathcal{C}_b$ treated at step 2 can only grow from one iteration to the next. Steps 3, 4, 5 and 6 are designed so as to control the growth of $\mathcal{C}_b$, by efficiently identifying the uncontrollable contingencies at every iteration. Obviously, a contingency labeled as noncritical, discarded or controllable at an iteration may become critical, selected, uncontrollable at subsequent iterations, and possibly binding at the final optimal solution.

C. Contingency Filtering Techniques

1) Severity-Index-Based Contingency Ranking Approach: The classical severity-index (SI)-based contingency ranking technique, most often used in the context of any iterative SCOPF solution, is based on post-contingency violations, derived from SSSA applied to the SCOPF solution [1], [3]. Let us denote by $h_{kr}(\mathbf{P}^\star_{\mathcal{C}_b})$ the left-hand value of constraint r, $\forall r \in \{1, \ldots, q\}$, relative to any contingency $k \in \mathcal{C}_c$, where q is the size of any vector $\mathbf{h}_k$ in (3). This quantity is computed at step 3 of the ICSCOPF procedure after simulating, by a classical power flow program, contingency k at the current operating point $\mathbf{P}^\star_{\mathcal{C}_b}$. Let us further define by

$$h^+_{kr}\left(\mathbf{P}^\star_{\mathcal{C}_b}\right) = \max\left(0, h_{kr}\left(\mathbf{P}^\star_{\mathcal{C}_b}\right)\right) \quad (10)$$

the constraint violation (by extension, we set this quantity to $+\infty$, $\forall\, r \in \{1, \ldots, q\}$, for contingencies leading to load flow divergence).

The classical severity index used in the context of SCOPF contingency selection is defined for contingency k as [1], [3]

$$SI(k) = \left\| \sum_{r=1}^{q} w_r h^+_{kr}\left(\mathbf{P}^\star_{\mathcal{C}_b}\right) \right\|_2 \quad (11)$$

where w_r denotes the weight associated to constraint r, which unless explicitly specified will be taken equal to 1, and $\|\cdot\|_2$ denotes the euclidean norm. The value of this SI is refreshed at every iteration the ICSCOPF procedure.

2) Nondominated Contingency (NDC) Approach: The NDC technique [18] relies on the comparison (at every iteration of the ICSCOPF procedure) of the constraints violations among critical contingencies (the subset $\mathcal{C}_c$).

Intuitively, a critical contingency is *dominated* at a given step of the ICSCOPF procedure, if there exists another contingency which leads to larger violation for every constraint.

We say that contingency k *dominates* contingency j if

$$h^+_{kr}(\mathbf{P}^\star_{\mathcal{C}_b}) \geq h^+_{jr}(\mathbf{P}^\star_{\mathcal{C}_b}), \quad \forall r \in \{1, \ldots, q\}$$

and the inequality is *strict* for at least one $r \in \{1, \ldots, q\}$, h^+_{kr} and h^+_{jr} referring to the same type of constraint. We say that contingency k is *dominated* in $\mathcal{C}_c$ if there is a contingency $j \in \mathcal{C}_c \setminus \{k\}$ that dominates it. The set of these contingencies and its complement are respectively denoted by $\mathcal{C}_d$ and $\mathcal{C}_s$ ($\mathcal{C}_s = \mathcal{C}_c \setminus \mathcal{C}_d$). We say that contingency k is *nondominated* in $\mathcal{C}_c$ if no contingency $j \in \mathcal{C}_c \setminus \{k\}$ dominates it. Thus, for contingency k to be nondominated it is not mandatory that it dominates at least one contingency from $\mathcal{C}_c$.

When filtering contingencies according to this technique, we identify at step 4 of the ICSCOPF the members of $\mathcal{C}_s$ in the following way.
1) Initialization: let $\mathcal{C}_d = \emptyset$.
2) For each contingency $j \in \mathcal{C}_c$ do:
    a) For each contingency $k \in \mathcal{C}_c \setminus (\{j\} \cup \mathcal{C}_d)$ do:
      If k dominates j let $\mathcal{C}_d \leftarrow \mathcal{C}_d \cup \{j\}$.
3) Let $\mathcal{C}_s = \mathcal{C}_c \setminus \mathcal{C}_d$.

When using this CF technique we call alternatively a selected (respectively discarded) contingency as *nondominated* (respectively *dominated*).

D. ICSCOPF Algorithm Variants

Note that several shortcuts of the basic ICSCOPF algorithm are possible by disabling one or more steps or using alternative techniques at some steps, as described below.
1) *Without filtering (WF):* is obtained by disabling the steps 4, 5 and 6 of the ICSCOPF algorithm, and includes automatically in the CSCOPF all critical contingencies identified by SSSA at each iteration (the SSSA acts as the sole filter).
2) *Infeasible post-contingency optimal power flow (IP-COPF):* is obtained by disabling the contingency filtering step of the ICSCOPF algorithm. Thus one adds to the CSCOPF at every iteration uncontrollable contingencies only.
3) *Severity index-based (SI):* consists in using at step 4 of the ICSCOPF algorithm the classical SI of Section III-C-I to filter contingencies, instead of the NDC scheme.

TABLE I
RECAPITULATION OF THE TERMINOLOGY USED FOR CONTINGENCIES

Type	Explanation
uncontrollable	it does not lead to feasible post-contingency state when taking into account only corrective actions
controllable	it leads to feasible post-contingency state thanks to corrective actions only
critical	it leads to post-contingency constraints violations at the power flow solution or to power flow divergence
non-critical	it does not lead to post-contingency constraints violations at the power flow solution or to power flow divergence
selected	it is selected by a certain filter
discarded	it is discarded by a certain filter
non-dominated	there is no contingency that leads to larger violations of every constraint
dominated	there is a contingency that leads to larger violations of every constraint

TABLE II
TEST SYSTEMS SUMMARY

system	n	g	d	b	l	t	o	s	c
Nordic32	60	23	22	81	57	31	4	12	53
IEEE118	118	54	91	186	175	11	9	14	184
RTE	1203	177	767	1797	1394	403	203	11	1210

A comparison of performances of the basic ICSCOPF approach and its variants will be provided in Section IV. Besides, the direct approach (DA) and Benders decomposition (see Appendix A) are provided as baseline for comparison.

For convenience, we recapitulate in Table I our terminology concerning the various types of contingencies.

IV. NUMERICAL RESULTS

A. Description of the Test Systems Used

In this section we present detailed numerical results obtained with the presented CSCOPF approaches on three test systems: a 60-bus system, which is a modified variant of the Nordic32 system [19], the IEEE118 system [20], and a model of the RTE (the French transmission system operator) system. A summary of their characteristics is given in Table II, where: n, g, d, b, l, t, o, s, and c denote the number of: buses, generators, loads, branches, lines, transformers, transformers with controllable ratio, shunt elements, and contingencies considered in CSCOPF, respectively. All tests have been performed on a PC 1.7-GHz Pentium IV with 512-Mb RAM.

B. Results With the IEEE118 Test System

1) Problem Statement: We focus on the problem of minimizing the overall generation cost by means of a "decoupled" CSCOPF. Control variables are the generators active power. Equality constraints are the ac bus active/reactive power balance equations and imposed voltages of generators. Inequality constraints are bounds on generator active/reactive powers, limits on branch currents and coupling constraints. These constraints apply both in pre- and in post-contingency states. As regards the coupling constraints we assume that, following the loss of a transmission (respectively generation) equipment, every generator is able to reschedule up to 8% (respectively 10%) of its active power physical range.

2) Application of the ICSCOPF Approach: We solve the CSCOPF problem by the ICSCOPF algorithm of Section III-B,

TABLE III
BINDING CONSTRAINTS AT SUCCESSIVE CSCOPF SOLUTIONS

iter	P_g	Q_g	I	cpl	total
1	3	0	3	-	6
2	11	0	24	70	105
3	19	0	32	88	139

TABLE IV
SSSA AND CF REPORT AT SUCCESSIVE CSCOPF SOLUTIONS

	SSSA				CF	
iter	ncc	ncv	$\Delta I_m(\%)$	$\Delta I_a(\%)$	nsc	ndc
1	49	71	78.9	9.5	19	30
2	35	47	34.8	9.3	12	23
3	32	43	29.0	9.0	11	21

TABLE V
UNCONTROLLABLE AND BINDING CONTINGENCIES

iter	uncontrollable contingencies	binding contingencies
1	31, 40, 53, 84, 96, 175	-
2	32, 122	31, 40, 84, 96, 175
3	-	31, 40, 84, 96, 175, 32, 122

while filtering contingencies with the NDC approach of Section III-C-II. We first run the base case OPF (no contingency constraints) and observe that three branch current constraints are binding at the optimal solution. The SSSA at this point yields 49 critical contingencies (out of 184) leading to branch current violations. The CF reveals among them 19 nondominated contingencies which are further checked with the PCOPF. Only six of these contingencies are deemed uncontrollable and hence are included in the CSCOPF. The latter is run again, leading to 24 active branch current constraints at the optimum, corresponding to five different post-contingency states. The SSSA at this point yields 35 critical contingencies (out of 178), 12 being considered as nondominated and among these two are found uncontrollable by PCOPF. These are added to those already selected and CSCOPF is run again with the resulting set of eight contingencies, providing 32 active branch current constraints in seven post-contingency states. The SSSA performed at this point indicates 32 critical contingencies (out of 176). Since, after checking them with PCOPF, all turn out to be controllable the sought optimum is reached. As a matter of fact seven contingencies (out of 184) are actually binding at the final optimum.

The main results of the computational process are summarized in Tables III–V.

Table III provides the number and type of binding constraints at the optimum of the CSCOPF. The columns labeled with P_g, Q_g, I, and cpl refer to constraints relative to generator active power, generator reactive power, branch current, and generator active power coupling constraints, respectively.

Table IV displays the SSSA and CF report at the optimal solution provided by the CSCOPF, where ncc, ncv, ΔI_m (respectively ΔI_a), nsc, and ndc represent the number of critical contingencies, the total number constraints violated, the maximum (respectively the average) branch overload among all critical contingencies, the number of selected contingencies and the number of discarded contingencies, respectively.

Table V lists the uncontrollable and binding contingencies at successive CSCOPF solutions. Observe that, in this example,

almost all uncontrollable contingencies at various iterations are binding at the final optimum.

By comparing Table IV and V one can conclude that the efficiency of the NDC filtering technique is quite satisfactory. Thus, the CF is very efficient at the first iteration, allowing the identification of all six uncontrollable contingencies (five of them being binding at the next iteration) by keeping 19 nondominated (out of 49 critical) contingencies, thereby saving 30 runs of the PCOPF module. The CF is slightly less efficient at the second iteration, where it identifies the two uncontrollable contingencies (both being binding at the final optimum) by keeping 12 nondominated (out of 35 critical) contingencies and saving however 23 runs of the PCOPF.

Clearly, additional uncontrollable contingencies may appear during iterations of the sequential procedure, due to shifting the optimal base case to accommodate the uncontrollable contingencies identified in previous iterations. This is a reasonable risk taken by the iterative approach with respect to the direct approach, since in practice only a small number of contingencies from the whole set are binding at the final optimum. As is shown in Table XVI of Appendix D, this effect is even worse if one uses the Benders decomposition, since any uncontrollable contingency at an iteration is taken into account just by a linearized constraint, added to the base case OPF.

3) Comparison of Various CSCOPF Approaches: We now compare the efficiency in terms of overall CPU time of various CSCOPF approaches described in Section III, namely: the proposed ICSCOPF approach, its variants WF, IPCOPF and SI (see Section III-D), the direct approach (DA), and the BD. For completeness, details about the BD algorithm and its results are given in Appendices A and B.

Table VI reports, for these CSCOPF approaches, the overall CPU time (in seconds) of all tasks during the CSCOPF solution. In this Table $npbc = \mathrm{card}(\mathcal{C}_b)$ denotes the number of potentially binding contingencies handled in the CSCOPF iterative approach at every iteration, while nr is the number of contingencies for which the PCOPF variant is run. According to the sequential CSCOPF algorithm nr is equal to either nsc or $ncc = nsc + ndc$. As regards BD, the superscript * which appears in the column $npbc$ indicates that Benders cuts are accumulated during iterations to OPF base case constraints (see Appendix B). Finally, for the SI approach we provide the range of overall CPU time of CSCOPF solution, when the number of top ranked contingencies selected at every iteration, which we denote by M, is allowed to vary between 1 and 20. The CPU times of other tasks during SI approach correspond to its best performances, obtained for $M = 11$.

Note first that any iterative CSCOPF approach, with or without filtering, is (significantly) faster than the direct approach. In this example BD approach outperforms all other CSCOPF approaches in terms of overall CPU time, being followed by ICSCOPF. Observe also that even for the best parameter M setting, the SI technique leads to a much slower execution time than with the (NDC-based) ICSCOPF. More details about the comparison between NDC and SI approaches are reported in Appendix C.

Despite of the significant number of contingencies binding at the optimum, in this example the BD performs very well due to: the infeasibility degree of slave subproblems is rather small during iterations (see Table XIV), indicating that the base case OPF solution is quite close to that of CSCOPF, and the post-con-

TABLE VI
COMPARISON OF VARIOUS APPROACHES TO THE CSCOPF SOLUTION

approach	iter	CSCOPF ($npbc$)	SSSA	CF	PCOPF (nr)	time
ICSCOPF	1	0.5 (0)	2.7	0.0	9.2 (19)	
	2	7.9 (6)	2.7	0.0	5.8 (12)	60.1
	3	13.2 (8)	2.6	0.0	15.5 (32)	
DA	-	232.1 (184)	-	-	-	232.1
WF	1	0.5 (0)	2.7	-	-	
	2	68.7 (49)	1.9	-	-	152.1
	3	76.5 (54)	1.8	-	-	
IPCOPF	1	0.5 (0)	2.7	-	24.1 (49)	
	2	7.9 (6)	2.7	-	16.6 (35)	85.8
	3	13.2 (8)	2.6	-	15.5 (32)	
SI	1	0.5 (0)	2.7	0.0	5.3 (11)	
	2	5.1 (4)	2.7	0.0	5.3 (11)	92.2...
	3	6.4 (5)	2.7	0.0	15.5 (32)	...124.7
	4	9.8 (7)	2.7	0.0	5.4 (11)	
	5	13.2 (8)	2.6	0.0	15.5 (32)	
BD	1	0.5 (0)	2.7	0.0	9.2 (19)	
	2	0.6 (0*)	2.7	0.0	7.7 (15)	48.1
	3	0.7 (0*)	2.7	0.0	21.4 (42)	

tingency voltage magnitudes are hardly sensitive to generators active power rescheduling.

Let us stress the importance of CF within all CSCOPF techniques. For instance by disabling the CF step in the BD approach the execution time increases from 48.1 s to 75.8 s, i.e., more than the ICSCOPF technique. Also, both ICSCOPF and IPCOPF techniques are faster than WF scheme. Last but not least, the computing time of the CF task is negligible (order of milliseconds) when using the NDC or SI techniques.

Note that there are two factors which influence the time saved by contingency filtering. The first one is the number of calls to PCOPF variant and depends on the filter quality. The smaller the number of calls to PCOPF variant, the higher the overall time saving. For instance, although the contingencies included in CSCOPF are the same in both ICSCOPF and IPCOPF approaches, ICSCOPF approach calls 54 times less the PCOPF variant than IPCOPF technique. The second factor is the number of selected but uncontrollable contingencies, which is unpredictable beforehand and varies from one problem to another. Clearly, the lower the number of such contingencies, the higher the time saved by ICSCOPF approach.

C. Results With the Nordic32 Test System

1) Problem Statement: We concentrate on the generation cost minimization by means of a "full" CSCOPF. For this problem we consider the following control variables: generators active power, generators voltage, controllable transformer ratios and shunt reactances. Equality constraints are again the bus active/reactive power balance equations, while inequality constraints include bounds on all control variables as well as limits on generators reactive power, bus voltage magnitudes and branch currents. The bus voltage magnitudes are allowed to vary between 0.95 pu (respectively 0.92 pu) and 1.05 pu (respectively 1.08 pu) in pre-contingency (respectively post-contingency) state. In the coupling constraints we assume that, following the outage of a transmission (respectively generation) equipment, every generator is able to reschedule up to 10% (respectively 20%) of its active power physical range. The transformers ratio (respectively shunts reactance) are allowed to vary, following a contingency, up to 50% (respectively 20%) of their physical range. Generator terminal voltages are allowed

TABLE VII
BINDING CONSTRAINTS AT SUCCESSIVE CSCOPF SOLUTIONS

iter	P_g	Q_g	I	V	r	x	cpl	total
1	18	0	1	22	0	3	-	44
2	46	3	4	108	4	19	85	269

TABLE VIII
SSSA AND CF REPORT AT SUCCESSIVE CSCOPF SOLUTIONS

iter	SSSA							CF	
	ncc	$ncvI$	$ncvV$	ΔI_m (%)	ΔI_a (%)	ΔV_m (%)	ΔV_a (%)	nsc	ndc
1	23	40	9	77.3	20.4	5.0	2.6	13	10
2	4	0	7	-	-	3.2	1.9	4	0

TABLE IX
UNCONTROLLABLE AND BINDING CONTINGENCIES

iter	uncontrollable contingencies	binding contingencies
1	5, 12, 13, 15, 18, 19, 22, 23, 11	-
2	-	5, 12, 13, 18

to vary freely between 0.95 pu and 1.05 pu in both pre- and post-contingency states.

2) Application of the ICSCOPF Approach: The main results of the ICSCOPF approach are provided in Tables VII–IX.

Table VII yields the number and type of binding constraints at the CSCOPF optimum for successive iterations. The columns labeled with V, r, and x refer to constraints relative to bus voltage magnitudes, controllable transformer ratios, and shunt reactances, respectively.

Table VIII reports results of SSSA and NDC-based CF at each iteration, where *ncvI* (respectively *ncvV*) and ΔV_m (respectively ΔV_a) represent the number of branch current (respectively voltage magnitude) violations and the maximum (respectively average) voltage magnitude limit violation among all critical contingencies.

Table IX provides the uncontrollable and binding contingencies at successive CSCOPF solutions. Note that, two among the four binding contingencies (5 and 12) are both thermal and voltage limited, while the others only thermal constrained. Unlike the previous example, the binding contingencies are identified at iteration 2 since all of them are nondominated and uncontrollable at iteration 1; the other four critical and nondominated contingencies at iteration 2 turn out to be controllable by means of PCOPF variant (see Table VIII).

By comparing Table VIII and IX , one observes again that NDC technique performs well. Indeed, it classifies as nondominated 13 (out of 23 critical) contingencies and identifies correctly eight out of nine uncontrollable contingencies, except of contingency 11 which anyway is not binding at the optimum. Unlike the previous example, a less extent of uncontrollable contingencies are binding at the optimum (4 out of 9).

3) Comparison of Various CSCOPF Approaches: Table X reports the CPU times of all tasks during the CSCOPF solution of considered approaches. Detailed numerical results of CSCOPF solution by BD are provided in Appendix D.

As regards the SI approach we provide the range of overall CPU time of CSCOPF solution, when parameter M is allowed to vary between 1 and 20, while using the same weight, and

TABLE X
COMPARISON OF VARIOUS APPROACHES TO THE CSCOPF SOLUTION

approach	iter	CSCOPF ($npbc$)	SSSA	CF	PCOPF (nr)	time
ICSCOPF	1	0.5 (0)	0.5	0.0	7.1 (13)	19.7
	2	9.2 (8)	0.4	0.0	2.0 (4)	
DA	-	72.9 (53)	-	-	-	72.9
WF	1	0.5 (0)	0.5	-	-	28.1
	2	27.6 (23)	0.3	-	-	
IPCOPF	1	0.5 (0)	0.5	-	12.3 (23)	26.5
	2	10.8 (9)	0.4	-	2.0 (4)	
SI	1	0.5 (0)	0.5	0.0	3.2 (6)	15.2...
	2	7.9 (6)	0.5	0.0	2.6 (5)	...25.0
BD	1	0.5 (0)	0.5	0.0	7.1 (13)	58.0
	2	0.5 (0*)	0.5	0.0	6.8 (12)	
	3	0.5 (0*)	0.5	0.0	6.8 (12)	
	4	0.5 (0*)	0.5	0.0	5.4 (10)	
	5	0.6 (0*)	0.5	0.0	6.0 (11)	
	6	0.6 (0*)	0.5	0.0	6.1 (11)	
	7	0.6 (0*)	0.5	0.0	6.1 (11)	
	8	0.6 (0*)	0.5	0.0	5.3 (10)	

TABLE XI
SSSA AND CF REPORT AT SUCCESSIVE CSCOPF SOLUTIONS

iter	SSSA				CF	
	ncc	ncv	ΔI_m(%)	ΔI_a(%)	nsc	ndc
1	13	13	25.7	7.3	6	7
2	6	6	10.2	4.8	5	1

TABLE XII
INFEASIBILITY DEGREE (MW) OF UNCONTROLLABLE CONTINGENCIES

iter	uncontrollable contingencies	
	393	405
1	190.5	300.1
2	0.0	27.2
3	0.0	0.3

equal to 1 in (10), for both branch current and voltage magnitude post-contingency violations with respect to their limits. The CPU times of other tasks during SI approach correspond to its best performances, obtained for $M = 6$. Appendix E provides additional results about comparison of NDC and SI filtering techniques.

Since in SI approach the parameter M is set beforehand, we can conclude that in this case ICSCOPF and SI approaches lead to comparable performances, being faster than other competing techniques.

In contrast with the previous example BD approach converges very slowly, being faster only than the DA (see the Appendix D). We also notice that in this case the CF improves very little (7.2 s) the overall execution time of BD approach (see Appendix E for an explanation).

D. Results With the RTE System

We solve the same type of CSCOPF problem as in Section IV-BI for a 1203-bus model of the RTE system (see Table II).

The main results obtained with the ICSCOPF approach and with BD are reported in Tables XI–XIII, using the same format as in Section IV-B.

We notice that in this large scale example only one contingency (405) is binding at the optimum. On the other hand, when applying the ICSCOPF technique we find that 13 contingencies are critical at the first iteration and six among them are selected

TABLE XIII
COMPARISON OF VARIOUS APPROACHES TO THE CSCOPF SOLUTION

approach	iter	CSCOPF (npbc)	SSSA	CF	PCOPF (nr)	time
ICSCOPF	1	6.2 (0)	181.5	0.0	56.4 (6)	564.7
	2	84.3 (2)	181.2	0.0	55.1 (6)	
DA	-	- (1210)	-	-	-	-
WF	1	6.2 (0)	181.5	-	-	989.4
	2	622.3 (13)	179.4	-	-	
IPCOPF	1	6.2 (0)	181.5	-	122.2 (13)	630.5
	2	84.3 (2)	181.2	-	55.1 (6)	
SI	1	6.2 (0)	181.5	0.0	29.5 (3)	537.8...
	2	84.3 (2)	181.2	0.0	55.1 (6)	...989.4
BD	1	6.2 (0)	181.5	0.0	56.4 (6)	745.0
	2	7.0 (0*)	181.5	0.0	47.8 (5)	
	3	7.8 (0*)	181.5	0.0	75.3 (8)	

by the NDC filter. The latter performs quite well since the six selected contingencies indeed include the two uncontrollable contingencies (see Table XII) and the single one that is binding at the optimum.

Table XIII shows that the ICSCOPF approach is again faster than the other techniques (computer memory requirements prevented us from performing DA due to the large number of contingencies (1210) to be included with this method into the CSCOPF). The overall CPU time reported in the last column of this Table shows that the ICSCOPF approach also scales well to large systems and large contingency lists.

The conclusions drawn in Appendices C and E concerning the comparison of NDC and SI filtering techniques still hold in the case of the RTE system.

Looking more closely at the CPU times of the CSCOPF itself (third column) of Tables XIII, VI, and X we observe that it grows faster than linearly with the number of contingencies included in these computations. Thus, we expect that the ICSCOPF method could significantly outperform the BD approach when the number of uncontrollable contingencies at every iteration remains reasonably small and/or when the accuracy of the linear Benders cuts is questionable, such as in very stressed operating conditions, or when the CSCOPF optimum is far enough from the base case OPF solution.

V. CONCLUSIONS

This paper has presented and compared performances in terms of CPU time of several approaches aiming to solve the CSCOPF problem. They have been tested on three systems of 60, 118, and 1203 buses.

We have proposed a new iterative approach to the solution of CSCOPF problem which proves being overall faster than other competing techniques. The ICSCOPF approach exhibits additional advantages over the popular BD technique in terms of convergence robustness and theoretical soundness in the context of nonconvex nonlinear CSCOPF problems. The numerical results obtained with the ICSCOPF approach show that for solving the full problem, it is more effective to include progressively some potentially binding contingencies in the CSCOPF formulation, than to optimize the base case while keeping on shrinking its feasible region by accumulating Benders cuts. In conclusion the ICSCOPF approach appears to outperform the BD approach for CSCOPF computations.

This paper also emphasizes the importance of contingency filtering within sequential CSCOPF algorithms. Although the

NDC technique, which was adapted here from its original context of PSCOPF, appears to perform slightly less well in the context of CSCOPF, it compares quite favorably to the classical severity index-based contingency ranking technique. Moreover, it is worth to stress that the NDC technique is a parameter free technique as opposed to SI-based schemes, where parameters such as the number of top ranked contingencies selected and weights relative to different types of constraint violations must be chosen in an ad hoc way.

Finally, let us note that all these variants of the iterative CSCOPF approach as well as BD can benefit of a parallel computation framework to further speed-up computations.

APPENDIX

A) Solving the CSCOPF by Benders Decomposition: The (generalized) BD for CSCOPF solution consists of decomposing the original problem (1)–(4) into a master problem and several slave subproblems which interact iteratively [2].

The slave subproblem corresponding to contingency k can be formulated as the PCOPF (5)–(9).

In the BD approach for each uncontrollable contingency k a Benders cut (11) is generated and added to the master problem

$$b_k(\mathbf{u}_0) = y_k^\star + \boldsymbol{\pi}_k^{\star T}(\mathbf{u}_0 - \mathbf{u}_0^\star) \leq 0 \qquad (12)$$

where $\boldsymbol{\pi}_k^\star$ is the vector of dual variables corresponding to the constraint (8) at the optimum of the slave subproblem. This constraint conveys information about how the current optimum $\mathbf{u}_0^\star$ should be modified in order to reduce the problem infeasibility. Observe that the sole unknown in (11) is $\mathbf{u}_0$.

The master problem is formulated as

$$\min_{\mathbf{x}_0, \mathbf{u}_0} \quad f_0(\mathbf{x}_0, \mathbf{u}_0)$$
$$\text{s.t.} \quad \mathbf{g}_0(\mathbf{x}_0, \mathbf{u}_0) = \mathbf{0} \qquad (13)$$
$$\mathbf{h}_0(\mathbf{x}_0, \mathbf{u}_0) \leq \mathbf{0} \qquad (14)$$
$$\mathbf{b}(\mathbf{u}_0) \leq \mathbf{0} \qquad (15)$$

and contains base case constraints (13), (14) as well as all Benders cuts of type (11) provided by slave subproblems at every iteration, which we write compactly as (15).

The standard algorithm of CSCOPF solution by BD contains the following steps [2].

1) Solve a relaxation (12)–(14) of the master problem. Let $\mathbf{u}_0^\star$ be the optimal solution.
2) For each contingency $k = 1, \ldots, c$ solve the slave subproblem (5)–(9). If $y_k^\star > 0$, build the corresponding Benders cut (11).
3) If the infeasibility of each slave subproblem is below a predefined tolerance ε, i.e., $y_k^\star \leq \varepsilon$, $\forall\, k = 1, \ldots, c$ the convergence is achieved and $\mathbf{u}_0^\star$ is the optimal solution.
4) Solve the full master problem (12)–(15) by adding all Benders cuts generated at step 2. Let $\mathbf{u}_0^\star$ be the optimal solution. Go to step 2.

To further speed-up computations the BD standard algorithm is enhanced by using a SSSA and then the NDC filter at step 2, before solving the slave subproblems.

B) Application of Benders Decomposition Approach on the IEEE118 Test System: The main results of CSCOPF solution by

TABLE XIV
INFEASIBILITY DEGREE (MW) OF UNCONTROLLABLE CONTINGENCIES

iter	uncontrollable contingencies							
	31	32	40	53	84	96	122	175
1	27.9	0.0	111.3	11.0	41.8	45.8	0.0	12.7
2	0.9	16.1	0.0	0.0	0.5	4.2	5.6	2.5
3	0.0	0.1	0.0	0.0	0.1	0.2	0.0	1.0

TABLE XV
SSSA AND CF REPORT AT SUCCESSIVE MASTER PROBLEM SOLUTIONS

	SSSA				CF	
iter	ncc	ncv	$\Delta I_m(\%)$	$\Delta I_a(\%)$	nsc	ndc
1	49	71	78.9	9.5	19	30
2	43	58	64.8	10.7	15	28
3	42	55	64.8	10.7	16	26

TABLE XVI
INFEASIBILITY DEGREE (MW) OF UNCONTROLLABLE CONTINGENCIES

	SSSA							CF	
iter	ncc	$ncvI$	$ncvV$	ΔI_m (%)	ΔI_a (%)	ΔV_m (%)	ΔV_a (%)	nsc	ndc
1	23	40	9	77.3	20.4	5.0	2.6	13	10
2	12	11	7	30.6	12.2	3.6	2.0	12	0
3	12	12	7	36.9	15.7	3.7	1.9	12	0
4	13	18	5	54.1	18.2	4.0	2.3	10	3
5	11	11	7	27.0	11.2	4.0	2.0	11	0
6	11	10	7	22.8	11.6	3.9	2.2	11	0
7	11	11	7	24.2	11.7	3.3	2.0	11	0
8	10	10	5	23.9	11.5	3.3	2.1	10	0

TABLE XVII
SSSA AND CF REPORT AT SUCCESSIVE MASTER PROBLEM SOLUTIONS

	SSSA							CF	
iter	ncc	$ncvI$	$ncvV$	ΔI_m (%)	ΔI_a (%)	ΔV_m (%)	ΔV_a (%)	nsc	ndc
1	23	40	9	77.3	20.4	5.0	2.6	13	10
2	12	11	7	30.6	12.2	3.6	2.0	12	0
3	12	12	7	36.9	15.7	3.7	1.9	12	0
4	13	18	5	54.1	18.2	4.0	2.3	10	3
5	11	11	7	27.0	11.2	4.0	2.0	11	0
6	11	10	7	22.8	11.6	3.9	2.2	11	0
7	11	11	7	24.2	11.7	3.3	2.0	11	0
8	10	10	5	23.9	11.5	3.3	2.1	10	0

BD using the algorithm of Appendix A, while filtering contingencies with the NDC technique, are provided in Tables XV and XIV.

Table XIV displays the infeasibility degree (in MW) of uncontrollable contingencies at successive solutions of the master problem. We declare the convergence of BD as soon as the infeasibility of all slave subproblems become smaller than $\varepsilon = 5$ MW (see the algorithm of Appendix A), value kept for all subsequent examples of the paper. This table shows that 12 Benders cuts have been finally added to the master problem. Note that, likewise in the ICSCOPF approach, contingencies 32 and 122 become uncontrollable at the second iteration.

Table XV shows SSSA and CF results at successive master problem solutions. Except for the first iteration, these latter are different from the CSCOPF solutions of Table IV. We nevertheless observe that the NDC approach provides also here very satisfactory results, since the six uncontrollable contingencies at iterations 1 and 2 are identified by selecting 19 (out of 49) and 15 (out of 43) contingencies, respectively.

C) Comparison of Contingency Filtering Techniques on the IEEE118 Test System: We assess the NDC and SI filtering schemes based on their ability to identify all uncontrollable contingencies at an iteration and thereby to accelerate the CSCOPF solution.

The fastest SI-based CSCOPF solution (see Table VI) corresponds to $M = 11$, where M has been varied between 1 and 20. Therefore, even for the best parameter M setting, the SI-based filtering scheme leads to a much slower CSCOPF execution time than the NDC-based technique.

The NDC technique provides better results than SI scheme due to higher filtering accuracy. Thus, according to SI technique, the six uncontrollable contingencies at iteration 1 (see Table V) are ranked on positions: 1, 6, 9, 11, 14 and 37, respectively. Likewise, the two uncontrollable contingencies at iteration 2 are ranked on positions: 5 and 15, respectively. Thus, with SI scheme one needs to select at least the first 37 contingencies (out of 49) at iteration 1 and the first 15 contingencies (out of 35) at iteration 2. Since by using NDC technique one selects 19 contingencies at iteration 1, and 12 at iteration 2, respectively (see Table V), it results that, in this case, NDC technique clearly outperforms the SI scheme.

D) Application of Benders Decomposition Approach on the Nordic32 Test System: The main results of BD approach, using the NDC technique to filter contingencies, are gathered in Tables XVI and XVII.

Table XVI yields the infeasibility degree of uncontrollable contingencies at successive solutions of the master problems, a total number of 38 Benders cuts being accumulated into the master problem until convergence.

Note that, despite the very high initial infeasibility degree of most uncontrollable contingencies, its decrease is satisfactory at iteration 2. In contrast, the infeasibility degree of most uncontrollable contingencies significantly increases at next two iterations (3 and 4), while from iteration 5 on the convergence is slow. This behavior of BD implies that the new Benders cuts added to the master problem are unable to drive monotonically the successive optimal base cases to the final optimum.

From Tables XVI and XVII, one can observe that NDC filtering accuracy is rather poor at some iterations, e.g., at the second iteration it selects all 12 critical contingencies as dominated while only four of them are really uncontrollable. This behavior will be explained in Appendix E.

E) Comparison of Contingency Filtering Techniques on the Nordic32 Test System: The best performances of SI technique are obtained for $M = 6$, which allows to identify all four binding contingencies at the CSCOPF optimum. More precisely, the ICSCOPF approach is faster than SI scheme for $M \in [1, 5]$ and $M \geq 14$ while leading to exactly the same performances for $M = 13$.

Note that filtering further uncontrollable contingencies instead of selecting them all may be risky because uncontrollable but nonbinding contingencies at an iteration may become active at the final optimum. Recall that NDC approach selects 13 contingencies missing one uncontrollable nonbinding contingency (11). In order to select the same eight uncontrollable contingencies SI technique needs to keep at least the eight top ranked contingencies, being thus more (respectively less) accurate than NDC for $M \in [8, 12]$ (respectively $M \geq 14$).

To assess the impact of weighting factors on CPU times, we consider in the SI formula (10) constant weights (equal to 1)

for branch current limit violation, while varying weights related voltage limits violation between 0 and 100. Thus, weighting voltage limit violations five times higher than branch current violations leads to include three false alarms in the contingency ranking, which worsens the identification of binding contingencies. In these experiments the recorded CPU times of SI-based CSCOPF solution varies between 18.1 s and 25.0 s, which compares less favorably to NDC-based CSCOPF solution.

The good results obtained with SI technique in this case indicate that the overall amount of post-contingency violations is a reasonable indicator to assess whether a contingency is more prone to be uncontrollable than another. However, since this SI disregards the corrective controls sensitivity to remove violations, it may miss low ranked but uncontrollable contingencies. In contrast, NDC technique handles such situations by looking to both violation amount as well as whether violations are related to the same network elements or not. On the other hand, if critical contingencies lead to violations of different network elements, the NDC technique may select almost all critical contingencies as nondominated, while sometimes (see Appendix D) a good number of them are actually controllable by corrective actions only. Notice that this potential drawback of the NDC technique is mitigated in the context of iterative PSCOPF solution [18], where post-contingency violations corresponding to different network elements often require different preventive actions.

REFERENCES

[1] O. Alsac and B. Stott, "Optimal load flow with steady-state security," *IEEE Trans. Power App. Syst.*, vol. PAS-93, no. 3, pp. 745–751, May/Jun. 1974.

[2] A. J. Monticelli, M. V. P. Pereira, and S. Granville, "Security-constrained optimal power flow with post-contingency corrective rescheduling," *IEEE Trans. Power Syst.*, vol. 2, no. 1, pp. 175–182, Feb. 1987.

[3] B. Stott, O. Alsac, and A. J. Monticelli, "Security analysis and optimization (Invited Paper)," *Proc. IEEE*, vol. 75, no. 12, pp. 1623–1644, Dec. 1987.

[4] S. M. Shahidehpour *et al.*, "Nonlinear programming algorithms and decomposition strategies for optimal power flow," *IEEE Tutorial, Optimal Power Flow: Solution Techniques, Requirements and Challenges*, 1996.

[5] J. A. Momoh, R. J. Koessler, M. S. Bond, B. Stott, D. Sun, A. Papalexopoulos, and P. Ristanovic, "Challenges to optimal power flow," *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 444–455, Feb. 1997.

[6] J. F. Benders, "Partitioning procedure for solving mixed variables programming problems," *Numer. Math.*, vol. 4, pp. 238–252, 1962.

[7] A. M. Geoffrion, "Generalized Benders decomposition," *J. Optim. Theory Appl.*, vol. 10, no. 4, pp. 237–260, 1972.

[8] T. Gomez, I. J. Perez-Arriaga, J. Lumbreras, and V. M. Parra, "A security-constrained decomposition approach to optimal reactive power planning," *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 1069–1076, Aug. 1991.

[9] S. Granville and M. C. A. Lima, "Application of decomposition techniques to var planning: methodological and computational aspects," *IEEE Trans. Power Syst.*, vol. 9, no. 4, pp. 1780–1787, Nov. 1994.

[10] G. Strbac, S. Ahmed, D. Kirschen, and R. Allan, "A method for computing the value of corrective security," *IEEE Trans. Power Syst.*, vol. 13, no. 3, pp. 1096–1102, Aug. 1998.

[11] R. A. Schlueter, S. Liu, and N. Alemadi, "Preventive and corrective open access system dispatch based on the voltage stability security assessment and diagnosis," *Elect. Power Syst. Res.*, vol. 60, pp. 17–28, 2001.

[12] W. Li, M. Shaaban, Z. Yan, Y. Ni, and F. F. Wu, "Available transfer capability calculation with static security constraints," in *Proc. IEEE Power Eng. Soc. General Meeting*, Jul. 2003, pp. 306–310.

[13] F. Bouffard, F. D. Galiana, and A. J. Conejo, "Market-clearing with stochastic security (parts I and II)," *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 1818–1835, Nov. 2005.

[14] M. Shahidehpour and Z. Li, "Ac contingency dispatch based on security-constrained unit commitment," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 897–908, May 2006.

[15] J. Martinez-Crespo, J. Usaola, and J. L. Fernandez, "Security-constrained optimal generation scheduling in large-scale power systems," *IEEE Trans. Power Syst.*, vol. 21, no. 1, pp. 321–332, Feb. 2006.

[16] M. Shahidehopour and Y. Fu, "Benders decomposition: applying Benders decomposition to power systems," *IEEE Power Energy Mag.*, vol. 3, no. 2, pp. 20–21, Mar./Apr. 2005.

[17] F. Capitanescu, M. Glavic, D. Ernst, and L. Wehenkel, "Interior-point based algorithms for the solution of optimal power flow problems," *Elect. Power Syst. Res.*, vol. 77, no. 5-6, pp. 508–517, Apr. 2007.

[18] F. Capitanescu, M. Glavic, D. Ernst, and L. Wehenkel, "Contingency filtering techniques for preventive security-constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 1690–1697, Nov. 2007.

[19] Long-Term Dynamics, Phase II, CIGRE Task Force 38.02.08, 1995.

[20] IEEE-118 Bus System, 1996. [Online]. Available: http://www.ee.washington.edu.

Florin Capitanescu received the electrical power engineering degree from the University "Politehnica" of Bucharest, Bucharest, Romania, in 1997 and the Ph.D. degree from the University of Liège, Liège, Belgium, in 2003.

His main research interests include optimization methods and voltage stability.

Louis Wehenkel (M'93) received the Electrical Engineer degree (electronics) and the Ph.D. degree from the University of Liège, Liège, Belgium, in 1986 and 1990, respectively.

He is full Professor of Electrical Engineering and Computer Science at the University of Liège. His research is in stochastic methods for modeling, analysis, and optimization of complex systems, machine learning, and data mining, with applications in power systems planning/operation/control and bioinformatics/systems biology.