A Generalized Decomposition Framework for Large-Scale Transmission Expansion Planning

Mohammad Majidi-Qadikolai, *Student Member, IEEE*, and Ross Baldick, *Fellow, IEEE*

***Abstract*—In this paper, we propose a scalable and configurable decomposition framework for solving large-scale transmission capacity expansion planning with security constraints under uncertainties. This framework is capable of using both progressive hedging and Benders decomposition algorithms to decompose and parallelize a large-scale problem both vertically and horizontally. A scenario bundling method is also developed to create bundles through three steps, i.e., classification, clustering, and grouping with the objective of maximizing similarity between bundles. This bundling method can improve both quality of results (decreasing optimality gap) and performance (reducing computational time) of the proposed framework. To verify capabilities of the proposed method, it is applied to a reduced ERCOT system with 3179 buses, 4458 branches, and 10 scenarios. The numerical result for this case study shows that the proposed framework can make solving large-scale problems tractable, and provides high quality results (with less than 1% optimality gap) in a reasonable time (around 2.8 days).**

***Index Terms*—Benders decomposition, progressive hedging, security, stochastic programming, transmission expansion planning.**

NOMENCLATURE

Sets and Indices:

N_b: Set of buses; index *k, n*
N_g: Set of all generators; index g
N_{wg}: Set of all wind generators; index g
N_l: Set of all lines (existing and candidate); index *l, m*
N_o: Set of all existing lines; index *l, m*
N_n: Set of all candidate lines; index *l, m*
L_k: Set of lines connected to bus k
G_k: Set of all generators connected to bus k
Φ_l^ω: Set of lines with violated post-contingency flows under outage of line l in scenario ω
N_s^ω: Set of system operation states under scenario ω; index c ($c = 1$ represents the normal operation condition)
ICL^ω: Set of important lines for contingency analysis in scenario ω
υ: Superscript/index for iteration number
Ω: Set of scenarios; index ω
$\mathcal{I}$: Set of classes
$\mathcal{I}_i$: Set of scenarios in class i
$\mathcal{S}^i$: Set of clusters for class i
$\mathcal{S}_j^i$: Set of scenarios in cluster j for class i
$\mathcal{B}$: Set of bundles
$\mathcal{B}_i$: Set of scenarios in bundle i
$|\ |$: Size of a set

Parameters:

q_i: Per MWh load shedding penalty at bus i
γ_g: Per MWh wind curtailment penalty for wind farm g
Co_g: Per MWh generation cost for generator g
ζ_l: Annual cost of line l construction
d_k: Demand at bus k
B: Diagonal matrix of line suseptance
$P_g^{\max}/P_g^{\min}$: Maximum/Minimum capacity of generator g
$f_l^{\max}/f_l^{\min}$: Maximum/Minimum capacity of line l
C^ω: Matrix of contingencies (operation states) that specifies the status of lines under different contingencies (1 for in service and 0 for out of service lines) for scenario ω; index c
$\Gamma_{m,l}^\omega$: Magnitude of violation in flow of line m when line l is on outage in scenario ω
CII_l^ω: Contingency identification index for outage of line l in scenario ω
α: Line capacity modification factor for contingency conditions ($Emergency\ capacity\ Rating = (1+\alpha) \times Normal\ capacity\ Rating$)
ϑ: Variable freezing parameter
ρ_l: Penalty factor for line l in PH algorithm
κ: Size of each bundle
$\mathcal{A}$: Clustering attributes matrix
d: Size of a TEP optimization problem
SC: Number of structural constraints for a TEP problem
CV: Number of continues variables for a TEP problem
BV: Number of binary variables for a TEP problem

Random Variables:

$\tilde{\xi}^1$: load in MW
$\tilde{\xi}^2$: Available wind output in MW

Decision Variables:

$r_{k,c}$: Load curtailment at bus k under operating state c
CW_g: Wind curtailment for wind farm g

Manuscript received October 29, 2016; revised February 16, 2017 and May 26, 2017; accepted July 2, 2017. Date of publication July 11, 2017; date of current version February 16, 2018. This work was supported in part by the Defense Threat Reduction Agency and in part by the National Science Foundation. Paper no. TPWRS-01628-2016. *(Corresponding author: Mohammad Majidi-Qadikolai.)*

The authors are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78701 USA (e-mail: majidi.mohammad@gmail.com; baldick@ece.utexas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPWRS.2017.2724554

p_g: Output power of generator g
$f_{l,c}$: Power flow in line l under operation state c
$\theta_{i,c}$: Voltage angle at bus i under operating state c. $\Delta\theta_{l,c}$ is voltage angle difference across line l under operating state c. $\Delta\theta_{l,c} = \theta_{k,c} - \theta_{n,c}$ for line l from bus k to bus n.
x_l: Binary decision variable for line l
$\boldsymbol{x}^{\omega}$: Binary decision variables vector for scenario ω
$\boldsymbol{x}^{\mathcal{B}_i}$: Binary decision variables vector for bundle $\mathcal{B}_i$
$\boldsymbol{W}_{\mathcal{B}_i}$: Multiplier vector for bundle $\mathcal{B}_i$ in PH algorithm
$\boldsymbol{\mathcal{Z}}$: Binary variables matrix for clustering
$\boldsymbol{\mathcal{H}}$: Binary variables matrix for bundling

I. Introduction

With increasing interest in building large-scale solar parks and wind farms and the implementation of new environmental regulations such as the "clean power plan" that will result in retirement of some conventional power plants, the need for building new transmission network is inevitable even in places in which the demand growth is not significant [1].

A. Overview

The Transmission Expansion Planning (TEP) optimization problem has a long history that we briefly overview in this section. For a comprehensive overview of literature in this area, please read [2] and [3].

1) Solution Methods: Transmission planning methods can be divided into two main categories i.e., optimization-based, and heuristic models.

In optimization-based methods, which is the main focus of this paper, a mathematical formulation for TEP is developed and the problem is solved using classical optimization programming techniques. Several methods are proposed to formulate TEP problem. In [4] and [5], transmission planning is formulated as a linear optimization problem with continuous variables. Mixed-integer programming is another model that is widely used for TEP modeling ([6]–[9] for example). A nonlinear model for TEP is developed in [10]. A complex mathematical model for centralized transmission planning and decentralized generation expansion planning is developed in [11]. Decomposition techniques like Benders decomposition [12]–[17], cutting-plane method [18], and Progressive Hedging [19] are also used to solve the TEP optimization problem.

In heuristic models, the TEP problem is solved through several steps of generating, evaluating, and selecting expansion plans, with or without the user's help [2]. One of the common heuristic methods is to use sensitivity analysis to select additional circuits [20]–[23]. MISO [24], ERCOT [25], and CAISO [26] are three examples of independent system operators in the US that use different heuristic methods for TEP. As discussed in [2] and [27], existing optimization-based methods are computationally very expensive making them mainly impractical for large-scale TEP problems.

2) Power System Modeling: To model power flow analysis, either DC or AC models are used. Although AC models [28]–[30] are more accurate for power flow analysis, their nonlinear nature makes them less popular for long-term TEP problems compared to DC models [4], [6], [7], [31]. Linear approximation of network losses, reactive power and voltage magnitude are also integrated into the DC model to improve its performance for TEP analysis [32]–[34]. $N-1$ contingency analysis required by NERC for power system planning and operation [35] is integrated into TEP in [7], [9], [33], [36].

3) Uncertainties: Fast technology changes, new policies, increasing the penetration of mobile/flexible demand along with intermittent nature of renewable resources make it hard to accurately predict future generation mix/location and demand; therefore, these uncertainties should be explicitly modeled/evaluated in TEP process by system planners. Developing a single expansion plan that considers these uncertainties using methods that heavily depend on engineering judgment can be costly and inefficient. Authors in [8], [37] evaluated the impact of ignoring uncertainties on transmission planning. Authors in [38] used Information-Gap Decision Theory to model load uncertainties for a multi-stage transmission expansion planning problem.

The TEP optimization problem can be formulated as a two-stage stochastic resource allocation problem (a class of mixed-integer stochastic programming) to explicitly model uncertainties using a finite set of scenarios [39]. In this formulation, in the first stage, a decision about building a new transmission line is made, and the impact of this decision on power system operation under different scenarios is evaluated in the second stage. Although formulating TEP as a two-stage stochastic optimization problem provides a strong modeling capability [13], [14], [19], [36], solving the extensive form (EF) of this problem is not tractable even for medium size problems specially when $N-1$ contingency analysis is added to the problem. Therefore, decomposition and heuristic techniques should be used for solving TEP for medium to large-scale systems.

Robust optimization is another method to integrate uncertainties into the TEP formulation. In robust optimization, uncertainties are represented using a range for each uncertain parameter instead of developing scenarios (as used by stochastic optimization), and it finds a plan that is robust for the worst case scenario. In this case, the final result is usually too conservative motivates an Adaptive Robust Optimization [40] formulation with budget constraint limits to mitigate the level of robustness (conservativeness of results). Authors in [18], [41], [42] formulated the TEP as an adaptive robust optimization problem.

In this paper, we use stochastic optimization formulation to model uncertainties in TEP, corresponding to uncertainties with well-defined probability distributions.

B. Decomposition Techniques

Horizontal or *Vertical* decomposition techniques can be used to decompose a two-stage stochastic TEP optimization problem for large systems. Benders decomposition (BD) [43] is one of the widely used *horizontal* decomposition techniques for solving two-stage stochastic optimization problems. It divides the original problem into two parts i.e., master and subproblem and uses "cuts" from dual of the subproblem to model its constraints in the master problem [12]. References [12]–[17] applied BD to solve TEP optimization problem.

Although in several papers it is claimed that BD is easily scalable (for TEP) and can be used for real-size problems, authors in [44] showed that even for medium size networks when the number of scenarios is large (50 or more), an optimality gap between 3% to 6% would need to be accepted in the BD algorithm to get the result in a reasonable time. For large-scale problems, the subproblem itself will be hard to solve, and a large number of iterations between master and subproblem is required to meet optimality gap requirements. This problem worsens when reliability constraints are added to the TEP problem, in which subproblems should be solved for normal and under contingency operation states for all scenarios. The column-and-constraint generation method (also called cutting-plane method) is another *horizontal* decomposition technique that can be used to decompose a two-stage problem. In this method, primal "cuts" are used to represent the subproblem constraints in the master problem instead of dual cuts used by BD. Convergence guarantees and other properties of this method are explained in [45] and [46]. Authors in [18] deployed this decomposition technique for solving robust TEP.

Progressive Hedging (PH) [47] is aimed at decomposing a two-stage stochastic resource allocation problem *vertically* by solving the problem for each scenario separately, and adding *nonanticipativity* constraints to couple the first stage decision variables (standard PH). PH method for mixed-integer problems is a heuristic method that finds an upper bound answer for the optimization problem; however, authors in [48] developed a method to calculate a lower bound for results of the PH algorithm in order to quantify the quality of results. One drawback of standard PH algorithm is that for problems with a large number of scenarios and integer variables, it may need a large number of iterations to satisfy nonanticipativity constraints (and sometimes it may never converge if no heuristic action is taken inside the algorithm). Stochastic unit commitment [49], and transmission planning [19] are examples in power system in which standard PH is applied. Authors in [50] used PH for commodity network design, and in [51], PH algorithm is used for solving multi-stage stochastic mixed integer problems.

A decision regarding the type of decomposition technique i.e., *Horizontal* versus *Vertical* is usually made in advance (before problem formulation/modeling). However, depending on the size of the problem (either the network size or the number of scenarios) and the machine that is used to run the simulation, different decomposition techniques might be appropriate. For example, if the size of the network is large and a personal computer (PC) is used for simulation, probably using PH algorithm will not be a good choice because solving the extensive form of this problem for a single scenario by itself will be challenging, but moving from a PC to a workstation may change the situation. The same can be correct for BD when the model is developed for a problem with a small number of scenarios, and later a large number of scenarios is used to capture uncertainties. It can easily convert an efficient BD to an inefficient decomposition technique. Therefore a configurable framework is desirable.

In this paper, a generalized decomposition framework is developed that not only provides this opportunity to use either BD or HP but also makes it possible to use both decomposition techniques (hybrid), and takes advantages of both BD and PH for solving the same problem. Decomposing the problem by bundles of scenarios instead of each individual scenario will decrease the number of iterations in PH. But for large-scale problems, solving the extensive form (EF) of the bundled PH can be computationally expensive and even intractable. Instead of EF, we can use BD (as an efficient algorithm for problems with small/medium number of scenarios) to solve these bundled subproblems. In this way, a large-scale problem can be decomposed/parallelized both vertically and horizontally, and we can benefit from advantages of both decomposition techniques.

Our contributions in this paper are as follows:

1) Developing a scalable and configurable decomposition framework that makes it possible to use BD and PH techniques for solving large-scale problems
2) Proposing a bundling algorithm to effectively bundle scenarios for bundled PH

The proposed method is applied to two case studies i.e., a 13-bus system with 100 scenarios and a reduced ERCOT system with 3179 buses and 4458 branches with 10 scenarios. The results are also compared with standard PH, randomly bundled PH, and developed method in [36] to evaluate different aspects of the proposed framework. For the 13-bus case, the framework found the optimal answer (with 0.24% optimality gap) in 15 minutes, compared to the standard PH that found an answer (with 29.5% optimality gap) in more than 2 hours. For the ERCOT case study, the proposed framework found the expansion plan with 0.97% optimality gap after 2.78 days while the proposed method in [36] could not find a feasible solution after 15 days.

II. The Proposed Framework

A. Framework Overview

The proposed framework is designed to be flexible and configurable for different problem sizes on different machines. It can be configured to solve a problem in extensive form (EF), or using PH, BD, and Hybrid techniques that provides more flexibility from the modeling perspective. The proposed framework can be summarized as follows:

Phase 0: Data preparation

Step 1: Input data and set parameters

Input data includes base network, scenarios, and candidate lines list. In this step, the planner configures the framework by setting parameters; i.e., the number of scenarios in each bundle (κ) and the type of decomposition technique that should be used (PH, BD or Hybrid) for phases I and II. Settings for phase II can be modified later in step 4 if it is necessary.

Phase I: TEP without contingency analysis

Step 2: Scenario bundling

In this step, OPF for the base (existing) network is solved, and calculated load shedding and wind curtailment will be used to develop an attribute for scenario bundling. After developing appropriate criteria, scenarios are distributed between groups using the developed scenario bundling method (see Section II-B).

Step 3: Solving TEP
In this step, based on inputs from step 1 and bundles from step 2, TEP for normal operation states is solved. This step can be parallelized.

Phase II: TEP with contingency analysis
This phase is run if contingency analysis should be integrated in TEP process.

Step 4: Scenario Bundling
Based on parameter settings, the scenario bundling method (see Section II-B for more detail) is used to bundle scenarios. The VCL algorithm [9] is used to develop bundling criteria for this step.

Step 5: Solving TEP with contingency analysis
In this step, TEP with contingency analysis is solved. Based on framework's setting, either PH, BD, or hybrid may be used for solving this large-scale optimization problem. This step can be parallelized if PH and/or BD are selected as the solving algorithm.

Phase III: Quantifying the quality of results

Step 6: Calculating a lower bound answer
If PH or hybrid is selected for phase I and/or II, then finding a lower bound answer is necessary to quantify the quality of results. In this step, the proposed lower bound formulation for PH in [48] is used to calculate a lower bound.

Step 7: Calculate optimality gap
The optimality gap (ε) can be calculated using the upper bound from step 5 (or step 3 in case of TEP without contingency analysis) and the lower bound from step 6. The selected plan is $\varepsilon - suboptimal$.

The proposed framework is summarized in the flowchart in Fig. 1.

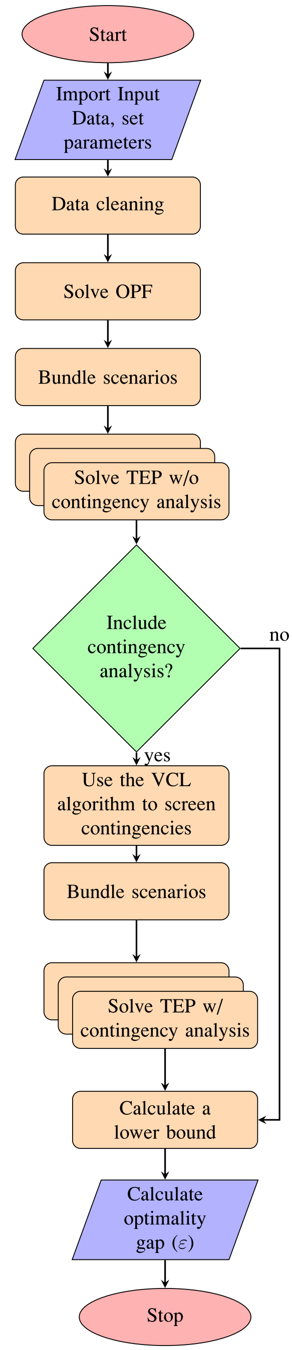

Fig. 1. Flowchart of the proposed framework.

B. Developed Scenario Bundling Method

In this section, a heuristic method is developed to bundle scenarios. The main purpose of this method is to create heterogeneous groups of scenarios with minimum dissimilarity *between* them (based on selected attributes/criteria) and with relatively the same computational burden. Having similar bundles will improve the performance of PH algorithm by facilitating convergence of nonanticipativity constraints, as for a set of identical groups of scenarios, PH only needs one iteration to converge (although this choice of bundling does not necessarily reduce computational time). In contrast with clustering in which the objective is to minimize dissimilarity *within* groups, scenario bundling tries to minimize dissimilarity *between* groups (mathematical formulation is provided in Section III-D). Developed groups partition the scenarios, and their size (κ) is constant for each phase. The proposed method bundles scenarios through three steps i.e., classification, clustering, and grouping. These steps are explained in the following sections. It should be noted that scenario bundling is required only if $1 < \kappa < |\Omega|$, where Ω is the set of all scenarios and $|\Omega|$ represents the size of this set.

1) Classification: In classification, a model or classifier is constructed to predict class labels such as "safe" or "risky" for bank loan application, or "light" and "heavy" loading conditions for electric networks. There are different classification methods such as Decision Tree Induction, Bayes Classification methods, and Rule-Based classification [52]. We use the Rule-Based method, because its structure allows us to easily integrate expert knowledge into the bundling process. It has the following structure:

$$\textbf{IF} \quad Condition \quad \textbf{THEN} \quad Conclusion \tag{1}$$

For our banking example, it can be written as

$$\textbf{IF} \quad age \leq 25 \; AND \; student \quad \textbf{THEN} \quad Safe$$

For electric network example, we can have

$$\textbf{IF}\ average\,line\,loading \geq 50\%\ \textbf{THEN}\ Heavily\,loaded$$

Rule-based classification will partition the original scenario set Ω into a finite number of non-empty classes $\mathcal{I} = \{\mathcal{I}_1, \ldots, \mathcal{I}_q\}$.

Different classification rules can be defined depending on the purpose of a study. For numerical analysis in Section V, we used the number of important lines for contingency analysis (ICLs) as a classifier in step 4. By using this classifier we may need to adjust the number of scenarios in classes (those that are close to boundaries) for feasibility of clustering in the next step. Classification is an optional step in the bundling process, and if there is no classifier, then there will be only one class that includes all scenarios ($\mathcal{I} = \{\mathcal{I}_1\}$).

2) Clustering: Clustering is the process of grouping a set of objects in a way that objects within a cluster have the highest similarity. In this step, similar scenarios in each class ($\mathcal{I}_i$) are clustered based on selected attribute/developed criteria, and form the set $\mathcal{S}^i = \{\mathcal{S}_1^i, \ldots, \mathcal{S}_c^i\}$. Without loss of generality, scenarios are clustered in groups with the same size in this paper, and the size of each cluster ($\mathcal{C}_s$) can be calculated from the following equation.

$$\mathcal{C}_s = \frac{|\Omega|}{\kappa} \tag{2}$$

where we assume that $|\Omega|$ is dividable by κ.

It is important to choose an attribute/criteria that is appropriate for the purpose of the study and provides insight for grouping in the next step. For example, bundles from step 2 of the proposed framework are used for solving TEP in step 3. Load shedding and wind curtailment (under normal operation condition) are highly penalized (compared to generation operation cost) in the TEP objective function (10); therefore, load (and wind) will be curtailed only if there is not enough transmission capacity to supply them (and transfer their output), which is a signal for a need for transmission expansion (resource adequacy is an implicit assumption for TEP studies). Therefore, we used these two components in the objective function to form a clustering attribute for phase I. A weighted sum of load shedding and wind curtailment (LW) is defined as a clustering attribute for this step. For step 4 of the framework, selected lines for contingency analysis for each scenario (ICL^ωs) is used as an attribute for scenario clustering because TEP with contingency analysis is solved in step 5, and ICL^ωs can significantly affect the selected expansion plan [9].

Partitioning method is used to create clusters by minimizing distance between different attributes of objects (scenarios here). For step 2, scenarios with closest LW values are clustered together, and for step 4, the objective of clustering optimization problem is to maximize similarity of ICL^ωs within each cluster. It creates a good "base" for grouping in the next step. An integer programming problem is solved to cluster scenarios in each step (see Section III-C for mathematical formulation).

3) Grouping into Bundles: In the last step of the proposed scenario bundling method, members of each cluster are distributed between groups (bundles) with the objective of minimizing dissimilarity *between* groups. For the scenario set Ω, a bundle set $\mathcal{B} = \{\mathcal{B}_1, \ldots, \mathcal{B}_b\}$ of non-empty and mutually exclusive subsets ($\forall i \neq j,\ \mathcal{B}_i \bigcap \mathcal{B}_j = \emptyset$ and $\bigcup_j \mathcal{B}_j = \Omega$) is formed. As scenarios in each cluster share some similar characteristics, one way is to distribute members of each cluster randomly between groups. It is also possible to define new criteria for grouping in this step. For developing new criteria, two main points should be noticed: first, the criteria should be at the group level rather than the scenario level because increasing similarity *between* groups is the purpose of this step. Second, the new criteria should not be significantly different compared to classification/clustering criteria, because the implicit assumption in this step is that scenarios in each cluster share similar attributes, and this assumption is mainly valid for the attributes used in previous steps. Ignoring these points may decrease similarity between formed groups.

For step 2, scenarios are distributed between groups so that groups have relatively the same aggregated LW value ($LW_{\mathcal{B}_i}$) because of its major contribution in the TEP objective function in step 3. For step 4, total number of ICLs in each group is used as a criteria for distributing scenarios between bundling groups. This attribute will result in forming groups with relatively the same number of operation states, which will have a huge impact on computational time. Combining this criteria with the one used for clustering will result in creating groups that have relatively the same impact on optimal result (because of similar lines for contingency analysis) and requires relatively the same computational burden (number of operation states).

As a separate stochastic TEP is solved for each bundle in PH algorithm, the probability of each scenario should be updated based on (3) and (4):

$$P_{\mathcal{B}_i} = \sum_{\omega \in \mathcal{B}_i} P^\omega \quad \forall \mathcal{B}_i \in \mathcal{B} \tag{3}$$

$$Pu^\omega = \frac{P^\omega}{P_{\mathcal{B}_i}} \quad \forall \omega \in \mathcal{B}_i, \forall \mathcal{B}_i \in \mathcal{B} \tag{4}$$

$$|\Omega| = \sum_{\mathcal{B}_i \in \mathcal{B}} |\mathcal{B}_i| \tag{5}$$

$$\sum_{\mathcal{B}_i \in \mathcal{B}} P_{\mathcal{B}_i} = 1 \tag{6}$$

where, P^ω is the original probability of scenario ω, $P_{\mathcal{B}_i}$ is probability of bundle $\mathcal{B}_i$ in set of bundles $\mathcal{B}$, and Pu^ω is updated probability of scenario ω as a member of bundle $\mathcal{B}_i$. Equations (5) and (6) enforce scenario bundling to be mutually exclusive.

Authors in [49] suggested that forming bundles with two scenarios may improve the performance of the PH algorithm for stochastic unit commitment problem, but they did not discuss how bundles should be formed. In [50], authors proposed a scenario grouping method for commodity transportation network planning, in which the objective of grouping is to maximize dissimilarity within groups (replacing minimization in (28) with maximization). Compared to [50], the proposed method in this paper minimizes dissimilarity between groups (using the objective function (37)), take into account the existing hardware infrastructure to control the size of each bundle, and forms bundles with relatively the same size to improve the performance of parallelizing (see Section IV-G for more details).

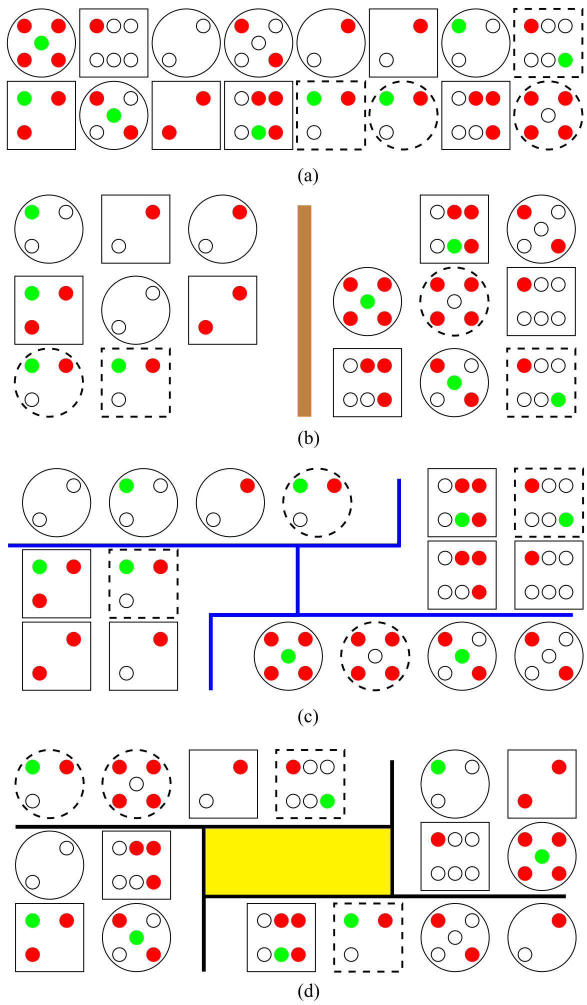

Fig. 2. An example for explaining different steps of the bundling method (a) A set of scenarios (b) Classification (c) Clustering (d) Grouping.

C. A Descriptive Example

In this section, a descriptive example is used to explain implementation of all steps of the developed scenario bundling method (from Section II-B). Fig. 2(a) shows a set of 16 scenarios ($|\Omega| = 16$) that are created to capture uncertainties in wind, load, and future market regulation on CO_2 emission for transmission planning purposes. The target is to bundle scenarios into groups of 4 ($\kappa = 4$) with the objective of minimizing the dissimilarity between groups (37). Shape (rectangular for high load and circle for high wind uncertainties), design (dashed lines represent the future market with CO_2 penalty, and solid lines for a future without CO_2 penalty), number of dots (shows the number of overloaded lines in the base case), and colors (the level of overload in lines) are used to visualize different attributes of scenarios. In the first step, scenarios are classified based on the number overloaded lines using the following rule:

$$\textbf{IF}\ number\,of\,overloaded\,lines \geq 5$$
$$\textbf{THEN}\ Heavly\,loaded\,network$$

In Fig. 2(b), the vertical brown line separates scenarios into two classes ($\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2\}$) based on their impact on network loading.

In the next step, scenarios in each class $\mathcal{I}_1$ and $\mathcal{I}_2$ are clustered based on similarity in uncertainties that they represent (their shapes). Based on (2), the size of clusters is equal to 4 ($\mathcal{C}_s = 4$). As the number of scenarios in each class is 8 ($|\mathcal{I}_1| = |\mathcal{I}_2| = 8$) and clustering with $\mathcal{C}_s = 4$ is feasible for each class, we do not need to modify the size of classes for this case. In Fig. 2(c), clusters are separated with blue lines ($\mathcal{S}^1 = \{\mathcal{S}_1^1, \mathcal{S}_2^1\}, \mathcal{S}^2 = \{\mathcal{S}_1^2, \mathcal{S}_2^2\}$). In the last step, scenarios in clusters are distributed between our 4 target groups ($|\mathcal{B}| = 4$) with the criteria that the number of overloaded lines and their level of overload (color here) have the most similarity (shown in Fig. 2(d)), and forming $\mathcal{B} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4\}$. Now we have 4 groups of scenarios, each including 2 high load scenarios and 2 high wind scenarios with 16 overloaded lines in each (7 at white level, 7 at red level, and 2 at green level).

It should be noted that similarity between these groups is only valid for attributes used in the bundling process. For example, the impact of having or not having CO_2 penalty (dashed versus solid lines) is considered as an attribute in neither of three bundling steps, and results (Fig. 2(d)) show that there is no similarity *between* groups for this attribute. These bundles of scenarios may not improve the performance of bundled PH algorithm if CO_2 penalty significantly affects the selected lines for transmission expansion (for example in systems with high penetration of cheap coal power plants).

III. Problem Formulation

A. Two-Stage Stochastic TEP Formulation

Stochastic programming is one of the widely used methods to model uncertainties (by developing different scenarios) in decision making process for resource allocation problems. Uncertainties in long-term transmission expansion can be categorized as macro uncertainties such as changes in market rules, environmental constraints or new technologies, and micro uncertainties such as hourly wind/solar and load variations [36]. To capture these uncertainties, different scenario generation/reduction methods might be used. The quality of scenarios is critical and can significantly affect the selected expansion plan. For example, in ERCOT, historical data along with workshops with stakeholders are used to develop scenarios for long-term TEP [53]. For a given scenario set, we develop a framework to efficiently solve this optimization problem; therefore, the nature of uncertainty and the origin of scenarios is not our concern in this work. It should be mentioned that minimizing the expected value is a better criterion for micro uncertainties. The two-stage stochastic TEP is formulated as follows:

$$Z^* = \min \zeta^\intercal \boldsymbol{x} + \mathbb{E}[Q(\boldsymbol{x}, \tilde{\xi})] \quad (7)$$

$$\text{st.} \quad \boldsymbol{x} \in \{0,1\}^{|N_l|} \quad (8)$$

$\mathbb{E}[Q(\boldsymbol{x}, \tilde{\xi})]$ represents the expected value of operation costs including load shedding and wind curtailment penalty and generation costs for TEP problem formulation. This expected value is approximated with a weighted sum of a limited number of scenarios as follows [54]:

$$\mathbb{E}[Q(\boldsymbol{x},\tilde{\xi})] \approx \sum_{\Omega} P^{\omega} Q(\boldsymbol{x},\xi^{\omega}) \tag{9}$$

where $Q(\boldsymbol{x},\xi)$ is the optimal value of power system operation for a given scenario ω [36].

$$Q(\boldsymbol{x},\xi) = \min \sum_{N_s}\left(\sum_{N_b} q_k r_{k,c}\right) + \sum_{N_{wg}} \gamma_g CW_g + \sum_{Ng} Co_g p_g \tag{10}$$

$$\text{st.} \quad -\sum_{L_k} f_{l,c} + \sum_{G_k} p_g + r_{k,c} = d_k \tag{11}$$

$$-\mathcal{M}_l(1 - C_{l,c}x_l) \leq f_{l,c} - B_{l,l}\Delta\theta_{l,c} \tag{12}$$

$$\mathcal{M}_l(1 - C_{l,c}x_l) \geq f_{l,c} - B_{l,l}\Delta\theta_{l,c} \tag{13}$$

$$CW_g \geq (P_g^{\max} - p_g) \tag{14}$$

$$(C_{l,c}x_l)f_l^{\min} \leq f_{l,c} \leq f_l^{\max}(C_{l,c}x_l) \tag{15}$$

$$P_g^{\min} \leq p_g \leq P_g^{\max} \tag{16}$$

$$0 \leq r_{k,c} \leq d_k \tag{17}$$

$$-\frac{\pi}{2} \leq \theta_{k,c} \leq \frac{\pi}{2} \tag{18}$$

$$CW_g \geq 0 \tag{19}$$

$$x_l = 1, \quad \forall l \in N_o \tag{20}$$

$$x_l \in \{0,1\}, \quad \forall l \in N_l \tag{21}$$

In (10), load shedding is penalized over all operating states (N_s) to satisfy the $N-1$ criterion (no load shedding is accepted during both normal and under single contingency states). Equation (11) enforces power balance at each bus. Equations (12) and (13) show DC representation of flow in transmission lines with big-$\mathcal{M}$ technique. Equation (14) measures wind curtailment at each bus. Equation (15) shows flow in all lines should always be between their maximum and minimum capacity limits. These limits will be modified based on the given value for α for emergency conditions (contingency in the network). Equations (16)–(18) enforce power plants' dispatch, load shedding and voltage angles to be between their minimum and maximum limits. Equation (19) enforces non-negativity of wind curtailment. Equation (20) sets decision variables for existing lines to 1. Equation (21) enforces that x_l is a binary decision variable for transmission lines ($x_l = 1$ when line l is built and $x_l = 0$ when line l is not built).

Constraints (11)–(19) represents lossless DC power flow model. Authors in [5] and [55] showed that DC model is accurate enough for long-term planning purpose because of the large level of simplifications in other aspects, but network losses, reactive power and voltage magnitude might be critical in some networks. Authors in [32]–[34], [56], [57] provide models to improve the performance of DC model by adding linear approximation of reactive power, network losses, and voltage magnitudes. As these models all preserve linearity of power flow model, they can be added to the model in this paper.

Depending on the size of the network and the number of scenarios, solving the extensive form of problem (7) can be extremely computationally expensive. Therefore, decomposition techniques are used to find a near-optimal answer for large-scale problems. In the next section, PH with bundled scenarios is explained as the base for steps 3 and 5 in our proposed framework. Details on BD technique is not in the scope of this paper and can be found in [58].

B. Progressive Hedging Algorithm With Bundled Scenarios

Progressive Hedging [47] is one of the decomposition techniques that can be used for solving two-stage (or multi-stage) stochastic mixed-integer optimization problems. The standard PH algorithm separates the problem vertically for each scenario. The TEP problem (7) can be rewritten as the following so-called *scenario* formulation:

$$Z^* = \min \sum_{\Omega} P^{\omega}[\zeta^{\mathsf{T}}\boldsymbol{x}^{\omega} + Q(\boldsymbol{x}^{\omega},\xi^{\omega})] \tag{22}$$

$$\text{st.} \quad \boldsymbol{x} \in \{0,1\}^{|N_l|} \tag{23}$$

$$\boldsymbol{x}^1 = \cdots = \boldsymbol{x}^s \tag{24}$$

A copy of decision variable vector $\boldsymbol{x}^{\omega}$ is created for each scenario ω in Ω that allows solution of the TEP problem for each scenario independently, and nonanticipativity constraints (24) are added to couple first stage solutions and guarantee that the final expansion plan does not depend on scenarios.

Instead of decomposing the problem for each individual scenario, it is possible to use bundles of scenarios ($\mathcal{B} = \{\mathcal{B}_1,\ldots,\mathcal{B}_b\}$) for decomposition. Equations (22)–(24) can be rewritten for bundled PH as follows:

$$Z^* = \min \sum_{\mathcal{B}} \left[P_{\mathcal{B}_i}(\zeta^{\mathsf{T}}\boldsymbol{x}^{\mathcal{B}_i}) + \sum_{\mathcal{B}_i} Pu^{\omega} Q(\boldsymbol{x}^{\mathcal{B}_i},\xi^{\omega})\right] \tag{25}$$

$$\text{st.} \quad \boldsymbol{x} \in \{0,1\}^{|N_l|} \tag{26}$$

$$\boldsymbol{x}^{\mathcal{B}_1} = \cdots = \boldsymbol{x}^{\mathcal{B}_b} \tag{27}$$

In this case, a copy of decision variable vector $\boldsymbol{x}^{\mathcal{B}_i}$ is created for all $\mathcal{B}_i$s in $\mathcal{B}$. Nonanticipativity constraints (27) are explicitly modeled for scenario bundles, and they are implicitly modeled for scenarios within each bundle (κ scenarios in bundle $\mathcal{B}_i$ already have the same first stage decision variable $\boldsymbol{x}^{\mathcal{B}_i}$) that usually reduces the number of iterations for convergence compared to standard PH.

Through an iterative process, PH will converge to a unique answer for the first stage decision variables by penalizing deviations of nonanticipative variables from their `mean` values. The PH algorithm with bundled scenarios is shown in Fig. 3. In the first line, the initial value of the iteration counter (υ), and multiplier vector ($\boldsymbol{W}_{\mathcal{B}_i}^{\upsilon}$) is set. From line 2–4, the TEP optimization problem for each bundle is solved separately (that can be parallelized). In line 5, the weighted sum of individual expansion plans ($\boldsymbol{x}^{\mathcal{B}_i,\upsilon}$s) is calculated. Line 6 calculates the deviation (Err) from averaged expansion plan ($\hat{\boldsymbol{x}}^{\upsilon}$). Lines 7–15 cover the main iterative part of the bundled PH algorithm. In line

1: Initialization: $\upsilon \leftarrow 1, \boldsymbol{W}^{\upsilon}_{\mathcal{B}_i} \leftarrow \mathbf{0}\ \forall \mathcal{B}_i \in \mathcal{B}$
2: **for** $\forall \mathcal{B}_i \in \mathcal{B}$ **do**
3: $\quad \boldsymbol{x}^{\mathcal{B}_i,\upsilon} \leftarrow argmin\, \boldsymbol{\zeta}^{\intercal}\boldsymbol{x}^{\mathcal{B}_i} + \sum_{\omega \in \mathcal{B}_i} Pu^{\omega} Q(\boldsymbol{x}^{\mathcal{B}_i}, \boldsymbol{\xi}^{\omega})$
4: **end for**
5: Aggregation: $\hat{\boldsymbol{x}}^{\upsilon} \leftarrow \sum_{\mathcal{B}} P_{\mathcal{B}_i}\boldsymbol{x}^{\mathcal{B}_i,\upsilon}$
6: $Err \leftarrow \sum P_{\mathcal{B}_i}\|\boldsymbol{x}^{\mathcal{B}_i,\upsilon} - \hat{\boldsymbol{x}}^{\upsilon}\|$
7: **while** $Err \geq \epsilon$ **do**
8: $\quad \upsilon \leftarrow \upsilon + 1$
9: $\quad \boldsymbol{W}^{\upsilon}_{\mathcal{B}_i} \leftarrow \boldsymbol{W}^{\upsilon-1}_{\mathcal{B}_i} + \boldsymbol{\rho}^{\intercal}(\boldsymbol{x}^{\mathcal{B}_i,\upsilon-1} - \hat{\boldsymbol{x}}^{\upsilon-1})$
10: $\quad$ **for** $\forall \mathcal{B}_i \in \mathcal{B}$ **do**
11: $\quad\quad \boldsymbol{x}^{\mathcal{B}_i,\upsilon} \leftarrow argmin\, \boldsymbol{\zeta}^{\intercal}\boldsymbol{x}^{\mathcal{B}_i} + \sum_{\omega \in \mathcal{B}_i} Pu^{\omega} Q(\boldsymbol{x}^{\mathcal{B}_i}, \boldsymbol{\xi}^{\omega}) + \boldsymbol{W}^{\upsilon\,\intercal}_{\mathcal{B}_i}\boldsymbol{x}^{\mathcal{B}_i} + \frac{\boldsymbol{\rho}^{\intercal}}{2}(\boldsymbol{x}^{\mathcal{B}_i} - \hat{\boldsymbol{x}}^{\upsilon-1})^2$
12: $\quad$ **end for**
13: $\quad$ Aggregation: $\hat{\boldsymbol{x}}^{\upsilon} \leftarrow \sum_{\mathcal{B}} P_{\mathcal{B}_i}\boldsymbol{x}^{\mathcal{B}_i,\upsilon}$
14: $\quad Err \leftarrow \sum P_{\mathcal{B}_i}\|\boldsymbol{x}^{\mathcal{B}_i,\upsilon} - \hat{\boldsymbol{x}}^{\upsilon}\|$
15: **end while**

Fig. 3. Progressive hedging algorithm with bundled scenarios.

8, the value of counter is updated. Line 9 updates the value of multiplier vector by using penalty vector $\boldsymbol{\rho}$. Lines 10–12 solve an updated TEP formulation with multiplier and penalizing deviation from average value of first stage decision variables. This optimization problem is solved for each bundle independently, so they can be solved in parallel. Lines 13 and 14 update the calculated average value for $\boldsymbol{x}$ and Err, respectively.

C. Clustering Algorithm

As defined in [52], "cluster analysis or clustering is the process of partitioning a set of data objects (or observations) into subsets. Each subset is a cluster such that objects in a cluster are similar to one another, yet dissimilar to objects in other clusters." Major fundamental clustering methods can be classified into four categories i.e., Partitioning methods, Hierarchical methods, Density-based methods, and Grid-based methods. A detailed discussion on each category can be found in [52].

A partitioning method can be used to find mutually exclusive clusters based on distances between their elements. For a finite set of scenarios (for example, $\mathcal{I}_1 = \Omega$), a set $\mathcal{S}^1 = \{\mathcal{S}^1_1, \ldots, \mathcal{S}^1_c\}$ of non-empty subsets of $\mathcal{S}^1_i$ is a partition if $\forall k \neq j,\ \mathcal{S}^1_k \bigcap \mathcal{S}^1_j = \emptyset$ and $\bigcup_j \mathcal{S}^1_j = \mathcal{I}_1$. Partitioning can be formulated as an integer programming problem in which the objective is to minimize the distance (Euclidean distance here) between members of each cluster based on selected attribute(s).

$$\min \sum_{m=1}^{n_a}\sum_{k=1}^{n_c}\sum_{i=1}^{n_s}\sum_{j=1}^{n_s} \|\mathcal{A}_{i,m}\mathcal{Z}_{i,k} - \mathcal{A}_{j,m}\mathcal{Z}_{j,k}\|^2 \tag{28}$$

$$\text{st.} \sum_{k=1}^{n_c} \mathcal{Z}_{i,k} = 1, \quad \forall i \in \mathcal{I}_1 \tag{29}$$

$$\sum_{i=1}^{n_s} \mathcal{Z}_{i,k} = \mathcal{C}_s \quad \forall k \in \mathcal{S}^1 \tag{30}$$

where $\mathcal{A}$ is attribute matrix ($[n_s \times n_a]$) for set $\mathcal{I}_1$, n_c is the number of clusters ($n_c = |\mathcal{S}^1|$), n_s is the number of scenarios in set $\mathcal{I}_1$ ($n_s = |\mathcal{I}_1|$), n_a is the number of attributes, $\mathcal{C}_s$ is the number of scenarios in each cluster ($\mathcal{C}_s = \frac{n_s}{n_c}$, equivalent to (2)), and $\mathcal{Z}$ is the binary decision variables matrix ($[n_s \times n_c]$) that assigns scenarios to clusters.

$$\mathcal{Z} = \begin{bmatrix} \mathcal{Z}_{1,1} & \cdots & \mathcal{Z}_{1,n_c} \\ \vdots & \ddots & \vdots \\ \mathcal{Z}_{n_s,1} & \cdots & \mathcal{Z}_{n_s,n_c} \end{bmatrix}$$

Equation (29) enforces that each scenario can only be a member of one cluster. Equation (30) enforces that all scenarios should be assigned to clusters and the size of all clusters is equal to $\mathcal{C}_s$. This is designed based on the assumption that we made in this paper. However, to have a flexible cluster size, (30) can be replaced with (31) and (32):

$$\sum_{k=1}^{n_c}\sum_{i=1}^{n_s} \mathcal{Z}_{i,k} = n_s \tag{31}$$

$$\sum_{i=1}^{n_s} \mathcal{Z}_{i,k} \geq 1 \quad \forall k \in \mathcal{S}^1 \tag{32}$$

Equation (32) guarantees that there will be no empty cluster.

The objective function (28) is nonlinear. As $\mathcal{Z}$ is a matrix of binary decision variables, for all $i = j$ the nonlinear term $\mathcal{Z}_{i,k}\mathcal{Z}_{j,k}$ can be replaced with $\mathcal{Z}_{i,k}$. For $i \neq j$, the nonlinear term $\mathcal{Z}_{i,k}\mathcal{Z}_{j,k}$ can be replaced with a new binary variable $\mathcal{Y}_r$, and constraints (33)–(35) should be added to the IP problem:

$$\mathcal{Y}_r \leq \mathcal{Z}_{i,k} \tag{33}$$

$$\mathcal{Y}_r \leq \mathcal{Z}_{j,k} \tag{34}$$

$$\mathcal{Y}_r \geq \mathcal{Z}_{i,k} + \mathcal{Z}_{j,k} - 1 \tag{35}$$

The size of matrix $\boldsymbol{\mathcal{Y}}$ ($[n_r \times n_c]$) for a scenario set of n_s scenarios is equal to:

$$n_r = \frac{n_s \times (n_s - 1)}{2} \tag{36}$$

It should be mentioned that for cases in which a very large number of scenarios should be clustered, solving IP can be computationally expensive. There are heuristics such as k-means methods that can be used for partitioning. Details of these methods are not in the scope of this paper and can be found in [52].

D. Scenario Bundling Algorithm

As stated in Section II-B, the main goal of scenario bundling is to maximize similarity (minimizing dissimilarity) *between* bundles to improve the performance of bundled PH algorithm. This problem can be formulated as an integer programming problem. The mathematical formulation for scenario bundling is as follows.

$$\min \sum_{m=1}^{n_a} \sum_{b,b'=1}^{n_b} \|\mathcal{Q}_{b,m} - \mathcal{Q}_{b',m}\|^2 \quad (37)$$

$$\text{st. } \mathcal{Q}_{b,m} = \texttt{mean}\left(\sum_{i=1}^{n_s} \mathcal{A}_{i,m}\mathcal{H}_{i,b}\right), \ \forall b \in \mathcal{B}, \forall m \in \mathcal{A} \quad (38)$$

$$\sum_{b=1}^{n_b} \mathcal{H}_{i,b} = 1, \quad \forall i \in \Omega \quad (39)$$

$$\sum_{i=1}^{n_s} \mathcal{H}_{i,b} = \kappa \quad \forall b \in \mathcal{B} \quad (40)$$

Where $\mathcal{Q}_{b,m}$ is the average value of attribute m in bundle b that can be calculated from (38), and $\boldsymbol{\mathcal{H}}$ is the binary decision variable matrix for bundling. The objective function (37) maximizes the similarity between bundles by minimizing the distance between mean value of attributes of bundles. Equation (39) enforces each scenario should be assigned to a bundle, and (40) enforces the size of each bundle.

Solving this problem for a large set of scenarios can be computationally expensive; therefore, a heuristic method is developed in Section II-B to solve this problem faster.

E. Variable Contingency List (VCL) Algorithm

Modified Line Outage Distribution Factors (LODFs) are used to estimate post-contingency flow in transmission lines when one line is on outage. The following equations (based on [9]) are used to create important contingency lists for different scenarios:

$$\Gamma^{\omega}_{m,l} = \frac{f^{\omega}_{m,l} - f^{\max}_{m}}{f^{\max}_{m}} \qquad ,\forall m, l \in N_o, \forall \omega \in \Omega \quad (41)$$

$$\Phi^{\omega}_{l} = \{m \in N_o \,|\Gamma^{\omega}_{m,l} \geq \alpha\} \quad ,\forall l \in N_o, \forall \omega \in \Omega \quad (42)$$

$$CII^{\omega}_{l} = \begin{cases} \frac{\sum_{\Phi^{\omega}_{l}} \Gamma^{\omega}_{m,l}}{|\Phi^{\omega}_{l}|}, & \text{if} \quad |\Phi^{\omega}_{l}| \neq 0 \\ 0, & \text{if} \quad |\Phi^{\omega}_{l}| = 0 \end{cases} \quad (43)$$

$$ICL^{\omega} = \{l \in N_o \,|CII^{\omega}_{l} \geq \alpha\} \quad ,\forall \omega \in \Omega, \quad (44)$$

where (41) calculates over/under loading on line m when line l is out. In this equation, $f^{\omega}_{m,l}$ represents the magnitude of post-contingency flow in line m when line l is on outage. Equations (42)–(43) are used to calculate Contingency Identification Index (CII) for each scenario with α as the line capacity modification factor during contingencies that accounts for emergency or short-term rating of lines. Equation (44) creates important contingency list (ICL) based on CII (see [9], [59] for more details).

IV. Model Performance Discussion

A. Parameter Settings for the Framework

The size of each bundle (κ) and the choice of a decomposition method are set in step 1 in the framework (see Section II-A). Table I shows different possible combinations for setting these two parameters. For the PH algorithm, by setting $\kappa = 1$ a standard PH is solved, $1 < \kappa < |\Omega|$ will result in a bundled PH, and $\kappa = |\Omega|$ is equivalent to solving the extensive form (EF) of the optimization problem. If BD is selected as the solving method, then for $1 \leq \kappa < |\Omega|$, the problem is solved separately for each bundle, and a heuristic method should be used to select a unique first stage answer. For $\kappa = |\Omega|$, a standard BD is solved. When Hybrid method is selected, for $1 \leq \kappa < |\Omega|$, both PH and BD are used for solving the problem in steps 3 and/or 5 in the framework. This is discussed more in Section IV-C. For $\kappa = |\Omega|$, hybrid method will be the same as BD method. It should be mentioned that these parameters can be set independently for phases I and II providing more flexibility, potentially improving the effectiveness of the proposed framework.

TABLE I
Framework Performance Under Different Settings

	PH	BD	Hybrid
$\kappa = 1$	PH	Heuristic	Hybrid
$1 < \kappa < \lvert\Omega\rvert$	PH	Heuristic	Hybrid
$\kappa = \lvert\Omega\rvert$	EF	BD	BD

B. Factors Affecting the Choice of Parameters

The size of the problem, the design of decomposition algorithms, existing hardware infrastructure, and solvers are critical for making a decision about setting parameters for the proposed framework. We briefly overview these factors in the following.

1) The Size of the Problem (d): The number of structural constraints (SC), (11)–(14), continuous (CV) and binary (BV) decision variables are main factors for the size of the TEP optimization problem. For the extensive form of this TEP formulation from Section III-A (depending on the choice and design of decomposition algorithms, new variables and constraints may be added), these values can be calculated from the following equations:

$$d = \{SC, CV, BV\} \quad (45)$$

$$SC = (2 \times (|N_b| + |N_l|) \times |N^{\omega}_s| + |N_{wg}|) \times |\Omega| \quad (46)$$

$$CV = ((2 \times |N_b| + |N_l|) \times |N^{\omega}_s| + |N_g| + |N_{wg}|) \times |\Omega| \quad (47)$$

$$BV = |N_n| \quad (48)$$

If no contingency reduction technique is used, then $|N^{\omega}_s| = |N_l| + 1$ to model outage of each line. If the VCL algorithm is used for contingency reduction, then $|N^{\omega}_s| = |ICL^{\omega}| + 1$.

2) Design of Decomposition Algorithms: PH and BD are not black-box software packages with input and output vectors. These algorithms are designed based on specific needs and conditions. For BD, there are several different designs such as standard BD [43], multi-cuts BD [60], and nested BD [61], and each design can be configured differently. For PH, either the standard form [47] or the bundled form [62] might be used. Similar to BD, there are several internal settings for PH that can affect the performance of this algorithm.

3) Existing Hardware Infrastructure: The machine that is used to solve the TEP problem has an undeniable impact on the choice of a decomposition algorithm and the size of each bundle (κ). Machines with high computing power are usually capable of solving larger problems that make it possible to choose bundled PH with a large bundle size (κ). In the case of using multiple machines (or virtual machines for Cloud based workstations), implemented parallel computation structure will be another key factor.

4) Solvers: The main feature of a solver that affects the choice of parameters for the framework is its capability to distribute computation over multiple cores of a CPU and use all computing power of the machine. GUROBI and CPLEX are examples of commercial solvers with this capability.

As discussed above, there are several factors that can affect hardware and software design of this framework. For a designed framework, running a few individual simulations can provide a relatively good understanding about the performance of each module, and help on setting parameters for the framework.

C. Linking PH and BD

Usually steps 3 and 5 are the most time consuming steps of the proposed framework in Section II-A for large-scale problems. These steps can be solved by either HP, BD or both (Hybrid). The algorithm explained in Fig. 3 is used as the main structure for solving TEP in steps 3 and 5. In the following, it is explained how this algorithm is tailored for all combinations in Table I. For PH (the second column of the table), the whole algorithm is run and the extensive form of stochastic TEP is solved in lines 3 and 11 in Fig. 3. For BD (the third column of the table), BD is used to solve TEP in line 3, and the algorithm is terminated in line 4. For the hybrid case (the fourth column of the table), the whole algorithm in Fig. 3 is run, and BD can be used to solve TEP in lines 3 and/or 11. If the BD is not used, the EF of TEP is solved. For $\kappa = |\Omega|$, Err in line 6 will be zero and the algorithm will be terminated in line 6.

D. PH Performance Improvement

Several heuristics such as finding appropriate values for $\boldsymbol{\rho}$, variable freezing, cyclic behavior detection, and terminating PH when the number of remaining unconverged variables is small can be used to improve the performance of the PH algorithm [63]. In the following we will discuss some of these heuristic methods that are used in this paper.

1) Choice of $\boldsymbol{\rho}$: A good approximation for $\boldsymbol{\rho}$ is important for the PH algorithm to perform well. As shown in Fig. 3, the value of multiplier vector ($\boldsymbol{W}^{\upsilon}_{\mathcal{B}_i}$) is updated using the penalty vector $\boldsymbol{\rho}$. An appropriate multiplier vector can affect the number of required iterations for PH convergence, and the quality of the lower bound answer [48]. In [63], different heuristic methods for calculating effective values for $\boldsymbol{\rho}$ are proposed. Our experience with those methods shows that for the TEP problem using the following equation from [63] results in a better convergence rate.

$$\rho_l = \frac{\zeta_l}{x_l^{\max} - x_l^{\min} + 1} \tag{49}$$

where ρ_l is the l^{th} element of vector $\boldsymbol{\rho}$, and

$$x_l^{\max} = \max_{\mathcal{B}_i \in \mathcal{B}} x_l^{\mathcal{B}_i} \tag{50}$$

$$x_l^{\min} = \min_{\mathcal{B}_i \in \mathcal{B}} x_l^{\mathcal{B}_i} \tag{51}$$

For values of $\boldsymbol{\rho}$ close to the unit cost of its associated variable, PH algorithm should have a better performance both from convergence speed and quality of results. Selecting higher values for $\boldsymbol{\rho}$ will improve the convergence rate but may negatively affect the quality of results. On the other hand, very small values for $\boldsymbol{\rho}$ can improve the quality of results (by decreasing optimality gap), but can significantly increase the number of iterations.

2) Variable Freezing: To improve the convergence of PH algorithm, the *variable freezing* technique can be used. Based on this technique, first stage decision variables with values that did not change over the past ϑ iterations are frozen for future iterations. For example, for a case with 5 bundles and $\vartheta = 4$, the value of the decision variable x_l is frozen if for all 5 bundles during 4 successive iterations, its value did not change ($x_l^{1,\upsilon+1} = \cdots = x_l^{5,\upsilon+4}$).

The impact of freezing variables can be investigated from two perspectives; i.e., its impact on simulation time and its impact on the selected plan.

1) Impact on simulation time
   By freezing binary variables, the total number of binary variables is decreased as frozen variables will have fixed values and no decision about them will be made in subsequent iterations. It improves the performance of the algorithm by decreasing computational time for each iteration (as a TEP optimization problem with fewer binary variables will typically solve faster) and reducing the number of iterations (as a PH problem with fewer nonanticipativity constraints will typically converge faster).
2) Impact on the selected plan
   When a decision variable is frozen, the implicit assumption is that its value will not change during next iterations, but this assumption may not always be valid. Therefore, the selected plan might be negatively affected when variable freezing technique is used, especially for small values of ϑ like 1 or 2. By using more conservative values for ϑ, this effect can be mitigated.

The selected plan will be more sensitive to a small value for ϑ when there are several relatively similar candidate lines (in terms of cost and/or electric parameters) in a geographically limited area. For a large-scale network in which candidate lines are widely spread, a smaller value for ϑ can be selected.

Using the variable freezing technique may result in situations with only a very few unfrozen decision variables. Then PH can be terminated (to decrease the number of iterations), and the TEP with remaining binary variables solved in extensive form or using the BD algorithm.

3) Identical Parallel Candidate Lines: We have also noticed that having two (or more) identical parallel candidate lines can result in an unnecessary non-zero values of Err on lines 6 and/or 14 in PH algorithm (Fig. 3) when only one of those lines is selected as a part of expansion plan. We recommend to slightly modify the investment cost for otherwise identical lines to break the symmetry.

The above mentioned heuristic techniques can be used to improve convergence of PH algorithm, but it may result in a higher optimality gap in step 7. In many practical cases, it is critical to get the result in a reasonable time; therefore a faster answer

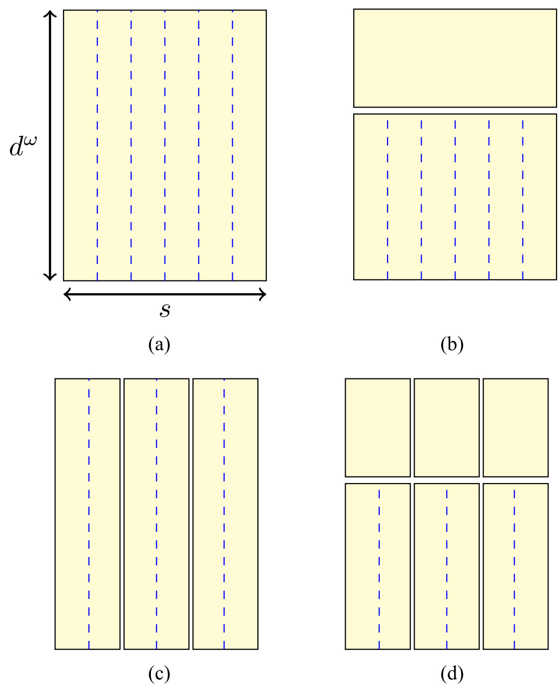

Fig. 4. The impact of different decomposition techniques, d^ω: size of the problem for scenario ω, s: the number of scenarios (6 for this example) (a) Extensive Form (b) BD (c) Bundled PH (d) Hybrid.

with a slightly higher optimality gap is usually acceptable (see Table III in Section V for numerical results).

E. Optimality Gap

The optimality gap is used as a measure for quantifying the quality of results in an optimization-based TEP. Based on Table I, the TEP problem is solved using one of these five methods i.e., heuristic, Extensive Form (EF), PH, BD, and hybrid. For parameter settings that will result in a heuristic method, we cannot calculate the optimality gap to quantify the quality of results. For the EF method, the optimality gap of the final result will be less than or equal to the solver's setting for maximum optimality gap. For BD, achieving the optimality gap is set as the stopping criterion; therefore, for EF and BD methods, it is possible to guarantee a pre-defined optimality gap (assuming that the algorithm successfully terminates). On the other hand, for PH and hybrid methods, the optimality gap is calculated after the algorithm is terminated to quantify the quality of final results, and there is no guarantee that the final optimality gap will be less than or equal to a pre-defined threshold. As discussed in Section IV-D, using appropriate values for $\boldsymbol{\rho}$ and setting a conservative value for ϑ can improve the optimality gap of the PH algorithm.

F. Scalability and Maintainability

Scalability is one of the main features of the proposed framework. We use Fig. 4 to discuss different aspects of this feature. Fig. 4(a) shows the size of the EF of a stochastic TEP problem with security constraints. In this Fig., d^ω represents the size of the TEP problem for scenario ω ($d^\omega = \{SC^\omega, CV^\omega, BV^\omega\}$), and s is the number of scenarios ($s = |\Omega|$).

$$SC^\omega = 2 \times (|N_b| + |N_l|) \times |N_s^\omega| + |N_{wg}| \quad (52)$$

$$CV^\omega = (2 \times |N_b| + |N_l|) \times |N_s^\omega| + |N_g| + |N_{wg}| \quad (53)$$

$$BV^\omega = |N_n| \quad (54)$$

For a case system with 6000 buses, 8000 existing lines and transformers, 500 conventional power plants, 100 wind farms, 100 candidate lines and 10 scenarios, $d^\omega = \{228.5M, 162.8M, 100\}$ when $|N_s^\omega| = 8101$ and $s = 10$. Total size of the problem in Fig. 4(a) will be $d = \{2285M, 1628M, 100\}$. This problem is practically impossible to solve in the EF. There are constraint reduction techniques [36], [64], [65] that can be used to decrease the size of this problem. Let's assume the VCL algorithm (see Section III-E) is used, and the size of N_s^ω is decreased form 8101 to 50. The size of the EF of this problem will be $d = \{14M, 10M, 100\}$. Even after a massive problem size reduction, solving the EF of the problem still remains computationally extremely expensive.

The BD algorithm (shown in Fig. 4(b)) moves binary decision variables to the master problem, and keeps all continuous variables in the subproblem. As the subproblem is a linear program, it is expected to be solved very fast; however, for the network in this example, the size of the subproblem will be $\{14M, 10M, 0\}$ which is not easy to solve especially if it should be solved in every iteration. Fig. 4(c) shows how bundled PH algorithm will decompose the problem. By creating bundles of two scenarios, the size of each subproblem for bundled PH will be $\{2.8M, 2.0M, 100\}$ (or $\{1.4M, 1.0M, 100\}$ for standard PH). Solving the extensive form of these subproblems might still be hard because of the large number of binary variables. In Fig. 4(d), the hybrid methods is used to decompose the problem both vertically and horizontally. By using this method, the size of each problem that needs to be solved in EF can be decreased up to $\{1.4M, 1.0M, 0\}$, which is a significant size reduction compared to $\{14M, 10M, 100\}$ for Fig. 4(a).

The size of this base case study can be increased either by increasing the number of candidate lines or the number of scenarios. The BD feature of the hybrid method will keep us away from exponentially increasing computational time as a result of adding new binary variables, and the bundled PH feature will keep the size of each subproblem relatively unchanged even if the total number of scenarios is increased significantly (by increasing the number of bundles instead of increasing the size of each bundle). Therefore the problem remains tractable, demonstrating the scalability of the proposed framework.

Another important feature of this framework is its maintainability. Because it is module based (BD algorithm, PH algorithm, bundling algorithm), and each module can be easily and (relatively) independently upgraded as technology improves.

G. Parallelizing

With proper hardware, parallelizing decreases computational time for solving a series of independent simulations, and it improves scalability of the framework. Simulations in steps 3 and 5 in the proposed framework can be parallelized, if PH, BD (with special configurations), or hybrid is selected to reduce elapsed time for solving TEP optimization problem by starting all independent simulations at the same time.

1) PH Algorithm: Based on PH algorithm for bundled scenarios shown in Fig. 3, lines 3 and 11 are run for each bundle (or each scenario in case of standard PH) independently. Therefore, we can parallelize both `for loops` (lines 2–4 and 10–12) in this algorithm, and start all simulations in each `loop` at the same time to decrease computational time. It should be noted that lines 10–12 should be solved for each iteration of the PH algorithm, and decreasing computational time here can be significantly rewarding from performance improvement perspective. As shown in lines 5 and 13 in Fig. 3, the algorithm can proceed to the next step when all parallelized simulations are completed. In the bundling process, it should be considered to develop bundles that need relatively similar computational time (see Section II-B3); so that the framework can benefit the most from parallelizing.

2) BD Algorithm: For standard BD, in which one cut is sent to master problem in each iteration, the subproblem should be solved in extensive form. For multi-cuts BD [60] and nested BD, [61] and [16], it is possible to solve subproblems in parallel that will decrease computational time.

3) Hybrid Method: As hybrid algorithm uses both PH and BD to solve a problem, it can benefit from both vertical and horizontal decomposition techniques and parallelize the problem solving with both algorithms (if applicable). For example, by using bundled PH, the problem will be vertically parallelized for each bundle $\mathcal{B}_i$. A nested BD can be used to solve each bundle, in which feasibility cuts for contingency operation states can be created in parallel.

V. Case Study and Numerical Results

In this section, we run numerical analysis for two case studies i.e., a 13-bus system with 100 scenarios and a reduced ERCOT system with 10 scenarios. All simulations are done with a personal computer with 2.0-GHz CPU and 32 GB of RAM. The proposed method is implemented in MATLAB R2014a [66] by using YALMIP R20150626 package [67] as a modeling software and GUROBI 5.6 [68] as a solver. To calculate the elapsed "Simulation Time," MATLAB built-in function `tic toc` is used. Steps 3 and 5 are parallelized using MATLAB built-in function `parfor`.

A. 13-Bus Test System

This case study contains 13 buses, 33 existing lines, 16 power plants, 9 load centers, and 36 candidate lines with 100 scenarios to capture uncertainties in wind and load [36]. This small case study with a large number of scenarios is used to demonstrate different steps of the proposed framework. The proposed method in [36] is used for solving TEP subproblems in lines 3 and 11 of the PH algorithm. To evaluate the performance of the proposed method, this test system is solved with four different methods that are explained in the following:

1) Case A: In case A, a standard PH (without bundling) is used to solve TEP problem. This method is used by [19] to solve TEP without contingency analysis. As stated before, MATLAB built-in function `parfor` is used to parallelize solving TEP for each scenario.

2) Case B: For case B, scenarios are bundled randomly using `randperm` function in MATLAB (instead of using the proposed method in Section II-B) to show the impact of bundling on performance of PH algorithm for TEP problem. The size of bundles is selected based on the problem size and machine's configuration ($\kappa = 20$).

3) Case C: This case solves the problem using the proposed framework in this paper. To show the implementation of the proposed framework, all steps are explained in detail.

Step 1: for phase I, κ is set to 50 as TEP without contingency analysis is solved, so a larger number of scenarios can be bundled compared to TEP with contingency analysis. For phase II, κ is set to 20 to fairly compare the result of cases B and C. Step 2: An OPF is solved for the base case to calculate LWs for bundling. Load shedding and wind curtailment penalties are set to \$9000/MWh and \$500/MWh respectively. It will result in the weight factor of 18 for load shedding (and 1 for wind curtailment), and LW for each scenario is calculated as the weighed sum of normalized wind and load curtailment in that scenario. Based on (2), $\mathcal{C}_s = 2$ and scenarios are clustered with the objective of minimizing the distance between LW^ω values in each cluster. In the last step, members of each cluster are distributed between bundling groups to minimize the distance between aggregated LW values ($LW_{\mathcal{B}_i}$s). Step 3: bundled PH is used to solve TEP without contingency analysis in this step. The final target is to solve TEP with contingency analysis, and results of this step are used as inputs for step 4 (to calculate a bundling attribute); therefore this step does not need to be solved until optimality. If TEP without contingency analysis is the final target, this step should be solved iteratively until the stopping criteria is met.

Phase II, step 4: the VCL algorithm [9] is used to find important lines for contingency analysis (ICL^ωs) using results from step 3. Scenarios are classified into 4 classes based on the size of ICL^ωs ($|ICL^\omega|$). Then, scenarios in each class are clustered based on similarity/dissimilarity of their ICL lists. It will result in clusters having members with relatively similar ICLs. In the last step of bundling, scenarios in each cluster are distributed between target bundles to create groups with relatively the same number of ICLs. This criterion tries to balance computational burden between groups. The size of ICLs in each group affects the number of operation states and consequently computational time. In step 5, bundled PH is solved iteratively until stopping criteria is met.

Phase III, Step 6: A lower bound is calculated (based on the proposed method by [48]) to quantify the quality of the result from step 5. In step 7, optimality gap is calculated based on upper and lower bounds from steps 5 and 6 respectively.

4) Case D: For this case, the proposed method in [36] is used to solve the TEP problem.

5) PH Algorithm Settings: The values of $\boldsymbol{\rho}$ are calculated based on (49). Freezing variables is one of the techniques that is used to improve convergence of PH algorithm. Variables that do not change over the most recent 4 iterations will be frozen at their values ($\vartheta = 4$). Moreover, if the number of remaining binary variables is less than or equal to 3, the PH algorithm is

TABLE II
SUMMARY OF RESULTS FOR 13-BUS SYSTEM

	Case A	Case B	Case C	Case D
No. of added lines	21	17	16	16
Objective Function ($b)	5.58	4.94	4.89	4.89
Simulation Time (hrs)	2.05	1.28	0.25	0.42
Optimality Gap	29.5%	1.65%	0.24%	2.7%

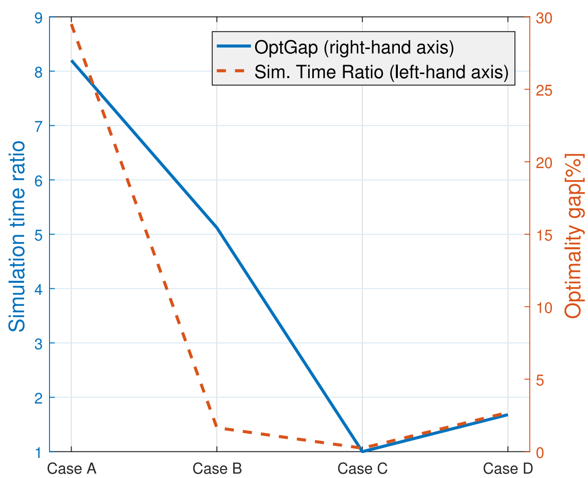

Fig. 5. Optimality gap and the ratio of simulation time.

terminated, and the extensive form of the problem is solved for remaining decision variables. These settings are applied to three cases A-C.

6) Model Performance Discussion: The simulation result for these four cases is summarized in Table II. Standard PH in case A needs more than 2 hours to solve this problem and the final result is 29.5%-suboptimal. It shows that standard PH will not have a good performance when the number of scenarios is large. For Case B, bundling reduced computational time by 50% and optimality gap is dropped to 1.65%. For case D, the TEP optimization problem is solved in 25 minutes with 2.7% optimality gap. The proposed method in case C reduced computational time to 15 minutes, and significantly improved the quality of results by decreasing optimality gap to 0.24%. Fig. 5 shows how computational time (left axis-solid blue line) and optimality gap (right axis-dashed orange line) are changed from case A–D. Computational time is normalized based on total time for case C. The proposed framework solved this problem more than 8 times faster than standard PH and 5 times faster than randomly bundled PH. It also found results with higher quality (0.24% compared to 1.65% and 29.4% for randomly bundled PH and standard PH respectively). From computational time perspective, cases C and D are relatively similar, but the quantified quality of results is significantly different, and case C provides a better optimality gap in somewhat less time.

To investigate the impact of parallelizing and variable freezing on computational time, we compared the performance of the framework under the following three alternatives:

1) *Alter. 1*: With variable freezing and without parallelizing

TABLE III
IMPACT OF PARALLELIZING AND VARIABLE FREEZING ON PERFORMANCE

		Alter. 1	Alter. 2	Alter. 3
Optimality Gap	Case A	29.5%	0.85%	29.5%
	Case B	1.65%	0.13%	1.65%
	Case C	0.24%	0.12%	0.24%
Simulation Time (hrs)	Case A	93.92	185.23	2.05
	Case B	7.38	132.97	1.28
	Case C	7.16	82.7	0.25

2) *Alter. 2*: Without variable freezing and with parallelizing
3) *Alter. 3*: With variable freezing and with parallelizing

Table III summarizes the impact of these two factors on optimality gap and computational time for cases A-C under these three alternatives.

The result from the second row shows that variable freezing may negatively affect the quality of results and increases the optimality gap (*Alter. 2*, in which variable freezing is ignored, has the lowest optimality gap). As expected, parallelizing will not affect the quality of results (similar optimality gaps for *Alter. 1* and *Alter. 3*) The third row in Table III shows the computational time for three alternatives. For *Alter. 1*, standard PH (Case A) is affected the most (compared to cases B and C) when parallelizing is not used because each iteration includes running TEP for all individual scenarios (simulation time increased from 2.05 to 92.38 hours). For bundled PH, both cases B and C could solve the problem in relatively the same time showing that when simulations are run sequentially (instead of in parallel), the impact of balancing computational burden between bundles (that will result in an earlier termination for a parallelized `for loop`) will be less effective. Variable freezing has a significant impact on computational time as it will decrease both the number of iterations and computational time for each iteration. Comparing the computational time and optimality gap for *Alter. 2* and *Alter. 3* shows the trade-off between quality of results and computational time. For example, for case C, the optimality gap is slightly increased from 0.12% to 0.24%; however the computational time is decreased from 82.7 hours to 0.25 hours that demonstrates the effectiveness of the proposed framework.

B. ERCOT Case Study

A reduced ERCOT network is developed with 3179 buses, 474 generation units, 3598 load centers, 123 wind farms and 4458 branches. All non-radial 138 kV and 345 kV lines in the ERCOT network are explicitly modeled. Generators and loads that were connected to lower voltage levels or radial network are moved to nearby modeled buses. Ten different scenarios are developed to model load and wind uncertainties (using historical data) with 46 new lines as candidates for transmission expansion. Similar to the 13-bus system, four cases A–D are simulated to compare the results. For phase I in case C, $\kappa = 5$ and for case B and phase II in case C, $\kappa = 2$. The proposed method in [36] is used to solve TEP in lines 3 and 11 of the bundled PH algorithm (Fig. 3). The parameter ϑ is set to 3. All other parameters are set the same as the 13-bus system. Numer-

TABLE IV
SUMMARY OF RESULTS FOR ERCOT SYSTEM

	Case A	Case B	Case C	Case D
No. of added lines	6	9	4	–
Objective Function ($b)	8.102	8.230	8.007	–
Simulation Time (days)	9.2	14.9	2.78	15
Optimality Gap	3.1%	6.24%	0.97%	–

ical result is given in Table IV. As the number of scenarios is not large for this system, standard PH (case A) has a reasonable performance; however, the elapsed time of over a week may not be acceptable. For case B (randomly bundled scenarios), simulation is terminated manually after 14.9 days and a lower bound is calculated. The fourth column (case C) demonstrates the impact of the proposed framework on improving quality of results (decreasing optimality gap from 6.24% to 0.97%) and reducing computational time (by more than 5.3 times) for solving this large-scale problem. We could not get a feasible solution for case D after 15 days, demonstrating the need for decomposition-based methods for large-scale problems.

Results for this case demonstrates that bundling by itself may not necessarily improve the performance of PH without careful consideration of the choice of bundles. Because each iteration for the PH algorithm is finished when TEP for all bundles are completely solved (lines 5 and 13 in Fig. 3). This comparison also highlights the importance of the grouping step in the proposed bundling method.

VI. CONCLUSION

In this paper, a generalized decomposition framework is developed for solving large-scale TEP problems. This framework is easily scalable, and its flexible structure makes it possible to configure it for problems with different sizes. It allows decomposition of a problem both *vertically* and *horizontally*, using bundled PH and BD algorithms respectively. The designed framework makes it possible to parallelize simulation and tract solving TEP for large-scale systems. A heuristic method is also developed to effectively bundle scenarios for PH algorithm. Its objective is to maximize similarity *between* bundles to improve the performance of the PH algorithm by speeding convergence of nonanticipativity constraints. Using this bundling heuristic decreased computational time by a factor of more than 8 and improved quality of results by reducing optimality gap from 29.5% to 0.24% for a 13-bus system with 100 scenarios. For a reduced ERCOT case study with 3179 buses and 10 scenarios, it provided a high quality result (0.97% optimality gap) in a reasonable time (2.8 days). The proposed framework makes solving TEP optimization problem for real-size networks tractable. This framework can be used by ISOs and transmission system owners for TEP studies. Multi-stage TEP is part of our future work.

ACKNOWLEDGMENT

We would like to gratefully thank our reviewers for their constructive comments that helped us to improve the quality of the paper. The authors would also like to thank Dr. David Morton for his suggestions.

REFERENCES

[1] J. Pfeifenberger and J. Chang, *Well-Planned Electric Transmission Saves Customer Costs: Improved Transmission Planning is Key to the Transmission to a Carbon-Constrained Future*. Cambridge, MA, USA: Brattle Group, Jun. 2016.

[2] G. Latorre, R. Cruz, J. Areiza, and A. Villegas, "Classification of publications and models on transmission expansion planning," *IEEE Trans. Power Syst.*, vol. 18, no. 2, pp. 938–946, May 2003.

[3] R. Hemmati, R.-A. Hooshmand, and A. Khodabakhshian, "Comprehensive review of generation and transmission expansion planning," *IET Gener., Transmiss. Distrib.*, vol. 7, no. 9, pp. 955–964, Sep. 2013.

[4] R. Villasana, L. Garver, and S. Salon, "Transmission network planning using linear programming," *IEEE Trans. Power App. Syst.*, vol. PAS-104, no. 2, pp. 349–356, Feb. 1985.

[5] L. Garver, "Transmission network estimation using linear programming," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 7, pp. 1688–1697, Sep. 1970.

[6] L. Bahiense, G. Oliveira, M. Pereira, and S. Granville, "A mixed integer disjunctive model for transmission network expansion," *IEEE Trans. Power Syst.*, vol. 16, no. 3, pp. 560–565, Aug. 2001.

[7] A. Rudkevich, "A nodal capacity market for co-optimization of generation and transmission expansion," in *Proc. 50th Annu. Allerton Conf. Commun. Control Comput.*, Oct. 2012, pp. 1080–1088.

[8] F. D. Munoz, B. F. Hobbs, J. L. Ho, and S. Kasina, "An engineering-economic approach to transmission planning under market and regulatory uncertainties: WECC case study," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 307–317, Jan. 2014.

[9] M. Majidi-Qadikolai and R. Baldick, "Integration of N-1 contingency analysis with systematic transmission capacity expansion planning: Ercot case study," *IEEE Trans. Power Syst.*, vol. 31, no. 3, pp. 2234–2245, May 2016.

[10] H. K. Youssef and R. Hackam, "New transmission planning model," *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 9–18, Feb. 1989.

[11] S. Jin and S. M. Ryan, "A tri-level model of centralized transmission and decentralized generation expansion planning for an electricity market: Part I," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 132–141, Jan. 2014.

[12] S. Granville *et al.*, "Mathematical Decomposition techniques for expansion planning vol 2: Analysis of the linearised power flow model using the bender decomposition technique," EPRI, Palo Alto, CA, USA, Tech. Rep. EL-5299, 1988.

[13] H. Park and R. Baldick, "Transmission planning under uncertainties of wind and load: Sequential approximation approach," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2395–2402, Aug. 2013.

[14] G. Chen, Z. Dong, and D. Hill, "Transmission network expansion planning with wind energy integration: A stochastic programming model," in *Proc. IEEE Power Energy Soc. General Meeting*, Jul. 2012, pp. 1–10.

[15] H. Zhang, V. Vittal, and G. Heydt, "An aggregated multi-cut decomposition algorithm for two-stage transmission expansion planning problems," in *Proc. IEEE Power Energy Soc. General Meeting*, Jul. 2015, pp. 1–5.

[16] T. Akbari, A. Rahimikian, and A. Kazemi, "A multi-stage stochastic transmission expansion planning method," *Energy Convers. Manage.*, vol. 52, no. 89, pp. 2844–2853, 2011.

[17] O. Alizadeh-Mousavi and M. Zima-Bokarjova, "Efficient benders cuts for transmission expansion planning," *Elect. Power Syst. Res.*, vol. 131, pp. 275–284, 2016.

[18] C. Ruiz and A. Conejo, "Robust transmission expansion planning," *Eur. J. Oper. Res.*, vol. 242, no. 2, pp. 390–401, 2015.

[19] F. D. Munoz and J.-P. Watson, "A scalable solution framework for stochastic transmission and generation planning problems," *Comput. Manage. Sci.*, vol. 12, no. 4, pp. 491–518, 2015.

[20] M. V. F. Pereira and L. M. V. G. Pinto, "Application of sensitivity analysis of load supplying capability to interactive transmission expansion planning," *IEEE Trans. Power App. Syst.*, vol. PAS-104, no. 2, pp. 381–389, Feb. 1985.

[21] G. Latorre-Bayona and I. J. Perez-Arriaga, "Chopin, a heuristic model for long term transmission expansion planning," *IEEE Trans. Power Syst.*, vol. 9, no. 4, pp. 1886–1894, Nov. 1994.

[22] M. Majidi-Qadikolai and R. Baldick, "Reducing the number of candidate lines for high level transmission capacity expansion planning under uncertainties," in *Proc. North Amer. Power Symp.*, Oct. 2015, pp. 1–6.

[23] A. Monticelli, A. Santos, M. V. F. Pereira, S. H. Cunha, B. J. Parker, and J. C. G. Praca, "Interactive transmission network planning using a least-effort criterion," *IEEE Power Eng. Rev.*, vol. PER-2, no. 10, pp. 46–47, Oct. 1982.

[24] Midcontinent ISO, "MTEP16 MISO Transmission Expansion Plan," 2016.

[25] ERCOT System Planning, "2016 Long-Term System Assessment for the ERCOT Region," 2016.
[26] Market & Infrastructure Development, CAISO, "California ISO 2016–2017 Regional Transmission Planning Process," 2016.
[27] F. D. Munoz, J.-P. Watson, and B. F. Hobbs, "Optimizing your options: Extracting the full economic value of transmission when planning under uncertainty," *Elect. J.*, vol. 28, no. 5, pp. 26–38, 2015.
[28] P. V. Gomes and J. T. Saraiva, "Hybrid discrete evolutionary pso for ac dynamic transmission expansion planning," in *Proc. IEEE Int. Energy Conf.*, Apr. 2016, pp. 1–6.
[29] I. Alhamrouni, A. B. Khairuddin, A. K. Ferdavani, M. Salem, and A. Alnajjar, "Differential evolution algorithm for multistage transmission expansion planning based on ac load flow model," in *Proc. 3rd IET Int. Conf. Clean Energy Technol.*, Nov. 2014, pp. 1–4.
[30] M. J. Rider, A. V. Garcia, and R. Romero, "Power system transmission network expansion planning using ac model," *IET Gener. Transmiss. Distrib.*, vol. 1, no. 5, pp. 731–742, Sep. 2007.
[31] M. Majidi-Qadikolai and R. Baldick, "Stochastic transmission expansion planning (TEP) with N-1 contingency analysis for large-scale network," in *Proc. INFORMS Int. Conf.*, Jun. 2016.
[32] N. Alguacil, A. Motto, and A. Conejo, "Transmission expansion planning: A mixed-integer LP approach," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1070–1077, Aug. 2003.
[33] H. Zhang, G. Heydt, V. Vittal, and J. Quintero, "An improved network model for transmission expansion planning considering reactive power and network losses," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3471–3479, Aug. 2013.
[34] R. Bent, C. Coffrin, R. R. E. Gumucio, and P. V. Hentenryck, "Transmission network expansion planning: Bridging the gap between ac heuristics and dc approximations," in *Proc. Power Syst. Comput. Conf.*, Aug. 2014, pp. 1–8.
[35] *Transmission System Planning Performance Requirements, TPL-001–4*, 2015. [Online]. Available: http://www.nerc.com
[36] M. Majidi-Qadikolai and R. Baldick, "Stochastic transmission capacity expansion planning with special scenario selection for integrating n-1 contingency analysis," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4901–4912, Nov. 2016.
[37] E. B. Cedeo and S. Arora, "Performance comparison of transmission network expansion planning under deterministic and uncertain conditions," *Int. J. Elect. Power Energy Syst.*, vol. 33, no. 7, pp. 1288–1295, 2011.
[38] S. Sarhadi and T. Amraee, "Robust dynamic network expansion planning considering load uncertainty," *Int. J. Elect. Power Energy Syst.*, vol. 71, pp. 140–150, 2015.
[39] P. Kall and S. Woodruff, *Stochastic Programming*. Chichester, U.K.: Wiley, 1994.
[40] D. Bertsimas, D. B. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Rev.*, vol. 53, no. 3, pp. 464–501, 2011.
[41] R. Garcia-Bertrand and R. Minguez, "Dynamic robust transmission network expansion planning," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 2618–2628, Jul. 2017.
[42] R. Minguez and R. Garcia-Bertrand, "Robust transmission network expansion planning in energy systems: Improving computational performance," *Eur. J. Oper. Res.*, vol. 248, no. 1, pp. 21–32, 2016.
[43] J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik*, vol. 3, pp. 238–252, 1962.
[44] F. D. Munoz, J.-P. Watson, and B. F. Hobbs, "New bounding and decomposition approaches for MILP investment problems: Multi-area transmission and generation planning under policy constraints," Sandia Natl. Lab., Albuquerque, NM, USA, SAND2014–4398J, 2014.
[45] R. Jiang, J. Wang, M. Zhang, and Y. Guan, "Two-stage minimax regret robust unit commitment," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2271–2282, Aug. 2013.
[46] Z. B. and L. Zhao, "Solving two-stage robust optimization problems using a column-and-constraint generation method," *Oper. Res. Lett.*, vol. 41, pp. 457–461, 2013.
[47] R. Rockafellar and R. J.-B. Wets, "Scenario and policy aggregation in optimization under uncertainty," *Math. Oper. Res.*, vol. 16, pp. 119–147, 1991.
[48] D. Gade, G. Hackebeil, S. M. Ryan, J.-P. Watson, R. J.-B. Wets, and D. L. Woodruff, "Obtaining lower bounds from the progressive hedging algorithm for stochastic mixed-integer programs," *Math. Program.*, vol. 157, no. 1, pp. 47–67, 2016.
[49] S. M. Ryan, R. J. B. Wets, D. L. Woodruff, C. Silva-Monroy, and J. P. Watson, "Toward scalable, parallel progressive hedging for stochastic unit commitment," in *Proc. IEEE Power Energy Soc. General Meeting*, Jul. 2013, pp. 1–5.
[50] T. G. Crainic, M. Hewitt, and W. Rei, "Scenario grouping in a progressive hedging-based meta-heuristic for stochastic network design," *Comput. Oper. Res.*, vol. 43, pp. 90–99, 2014.
[51] L. F. Escudero, M. A. Garn, M. Merino, and G. Prez, "An algorithmic framework for solving large-scale multistage stochastic mixed 01 problems with nonsymmetric scenario trees," *Comput. Oper. Res.*, vol. 39, no. 5, pp. 1133–1144, 2012.
[52] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Amsterdam, The Netherlands: Elsevier, 2011.
[53] ERCOT System Planning, "2014 Long-Term System Assessment for the ERCOT Region," 2014.
[54] Y. Ermoliev and R. J.-B. Wets, *Numerical Techniques for Stochastic Optimization* (Springer Series in Computational Mathematics). New York, NY, USA: Springer-Verlag, 1988.
[55] R. Baldick, K. Dixit, and T. Overbye, "Empirical analysis of the variation of distribution factors with loading," in *Proc. IEEE Power Eng. Soc. General Meeting*, Jun. 2005, vol. 1, pp. 221–229.
[56] C. Coffrin, P. Van Hentenryck, and R. Bent, "Approximating line losses and apparent power in ac power flow linearizations," in *Proc. IEEE Power Energy Soc. General Meeting*, Jul. 2012, pp. 1–8.
[57] C. Coffrin and P. V. Hentenryck, "A linear-programming approximation of ac power flows," *INFORMS J. Comput.*, vol. 26, no. 4, pp. 718–734, 2014.
[58] A. Conejo, E. Castillo, R. Minguez, and R. Garcia-Bertrand, *Decomposition Techniques in Mathematical Programming: Engineering and Science Applications*. New York, NY, USA: Springer, 2006.
[59] M. Majidi-Qadikolai and R. Baldick, "Large-scale transmission capacity expansion planning under n-1 contingency analysis," in *Proc. INFORMS Annu. Meeting*, Philadelphia, PA, USA, Nov. 2015.
[60] J. R. Birge and F. V. Louveaux, "A multicut algorithm for two-stage stochastic linear programs," *Eur. J. Oper. Res.*, vol. 34, pp. 384–392, 1988.
[61] C. R. Glassey, "Nested decomposition and multi-stage linear programs," *Manage. Sci.*, vol. 20, no. 3, pp. 282–292, 1973.
[62] R. J. Wets, "The aggregation principle in scenario analysis and stochastic optimization," in *Algorithms and Model Formulations in Mathematical*, S. Wallace, Ed. New York, NY, USA: Springer-Verlag, 1989, pp. 91–113.
[63] J.-P. Watson and D. L. Woodruff, "Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems," *Comput. Manage. Sci.*, vol. 8, no. 4, pp. 355–370, 2011.
[64] R. Madani, J. Lavaei, and R. Baldick, "Constraint screening for security analysis of power networks," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 1828–1838, May 2017.
[65] A. Ardakani and F. Bouffard, "Identification of umbrella constraints in dc-based security-constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 3924–3934, Nov. 2013.
[66] MATLAB, *Version 8.3.0.532 (R2014a)*, MathWorks, Natick, MA, USA, 2014.
[67] J. Lofberg, "Yalmip: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robot. Autom.*, Taipei, Taiwan, 2004, pp. 284–289. [Online]. Available: http://users.isy.liu.se/johanl/yalmip
[68] *Gurobi Optimizer Reference Manual*, Gurobi Optimization, Houston, TX, USA, 2014. [Online]. Available: http://www.gurobi.com

Mohammad Majidi-Qadikolai (S'14) received the B.Sc. degree from Power and Water University of Technology, Tehran, Iran, and the M.Sc. degree from the University of Tehran, Tehran, both in electrical engineering. He is currently working toward the Ph.D. degree in electrical engineering at The University of Texas at Austin, Austin, TX, USA. His research interests include power system planning and operation under uncertainties, stochastic programming, renewable energy integration, and electricity markets.

Ross Baldick (F'07) received the B.Sc. degree in mathematics and physics and the B.E. degree in electrical engineering from the University of Sydney, Australia, and the M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, Berkeley, CA, USA, in 1988 and 1990, respectively. From 1991 to 1992, he was a Postdoctoral Fellow at the Lawrence Berkeley Laboratory. In 1992 and 1993, he was an Assistant Professor at Worcester Polytechnic Institute, Worcester, MA, USA. He is currently a Professor in the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA.