Observing the State of a Smart Grid
Using Bilevel Programming

Sonia Toubaline[(✉)], Pierre-Louis Poirion, Claudia D'Ambrosio, and Leo Liberti

CNRS LIX, Ecole Polytechnique, 91128 Palaiseau, France
{toubaline,poirion,dambrosio,liberti}@lix.polytechnique.fr

Abstract. Monitoring an electrical network is an important and challenging task. Phasor measurement units are measurement devices that can be used for a state estimation of this network. In this paper we consider a PMU placement problem without conventional measurements and with zero injection nodes for a full observability of the network. We propose two new approaches to model this problem, which take into account a propagation rule based on Ohm's and Kirchoff's law. The natural binary linear programming description models an iterative observability process. We remove the iteration by reformulating its fixed point conditions to a bilevel program, which we then further reformulate to a single-level mixed-integer linear program. We also present a bilevel algorithm to solve directly the proposed bilevel model. We implemented and tested our models and algorithm: the results show that the bilevel algorithm is better in terms of running time and size of instances which can be solved.

Keywords: Bilevel program · Mixed integer linear program · Monitoring electrical network · PMU placement problem

1 Introduction

One of the properties making a grid *smart* is that its state is continuously monitored. The term *state* is an abstract concept which may be represented in many ways. We consider that a state of a grid is the set of values of all the branch currents and node voltages. Monitoring the state of a grid can be achieved using tools of measurement and control. A piece of equipment which can be used is Phasor Measurement Unit (PMU). PMUs are monitoring devices that provide time synchronized phasor measurement (a phasor is a complex number that represents both the magnitude and phase angle of the sine waves found in electricity). A PMU placed at a (sub)station measures the voltage and phase angle of this (sub)station and branch current phasor of all transmission line emerging from it [11]. PMUs are synchronized via global positioning systems (GPS) and send large bursts of data to a system monitoring centre. Due to the relatively high cost of PMUs, their optimal placement constitutes an important challenge.

Modelling the network by a graph where nodes correspond to (sub)stations and edges to transmission lines, the optimal location problem of PMUs, called

PMU PLACEMENT PROBLEM, consists of determining the minimum number of PMUs to place on the nodes, in order to ensure a full observability of the graph. A graph is said fully observed if the voltage is known at each node, and the current known on each edge. In [2], the observability of a graph is defined by two rules: (i) if a node has a PMU then this node and all its neighbours are observed; (ii) if an observed node has all its neighbours observed, except one, then this latter is observed. The PMU PLACEMENT PROBLEM is also known as POWER DOMINATING SET (PDS) problem [8]. This problem has been largely studied in literature. The PDS is shown to be *NP*-complete even for bipartite, chordal graphs [8] and planar bipartite graphs [2], and polynomial for trees and grids [4]. Some approximation results are presented in [1]. Different solution methods have been proposed to solve the PMU PLACEMENT PROBLEM [11,12]. In all these approaches, the propagation rule has been considered for the zero injection nodes with a limited depth.

The PMU PLACEMENT PROBLEM has also been studied for PMUs with limited channels ℓ where only a limited number of incident transmission lines can be observed [9,10]. Korkali and Abur propose a binary linear program considering for each node of the graph all the possible combinations of ℓ edges among incident edges of that node [9]. The number of combinations can be exponential. Kumar and Rao propose a new method to solve PMU PLACEMENT PROBLEM based on node connectivity and edge selectivity matrices where the number of channels are less than the minimum degree of the graph [10]. For PMUs with one channel, only one incident line can be observed. The placement of PMUs is then no more on nodes but on edges. The first rule of observability is then: if an edge has a PMU then its extremity nodes are observed. Emami et al. propose a binary linear program for this variant of the problem [5,6], which turns out to be equivalent to the minimum edge cover problem. They consider the second rule of observability defined in [2] for zero injection nodes with only one depth.

We propose in this paper a new approach to model the optimal location of PMUs with one channel. We place PMUs on edges (next to one of the adjacent nodes) and take into account both rules of observability. We call this particular variant the POWER EDGE SET (PES) problem. More specifically, we consider the case without conventional measurements (measures provided by non synchronized sensors) and all nodes are zero injection nodes (no current is injected in the network at those nodes). We present two mathematical formulations for this problem: iterative and bilevel models; the latter can be formally inferred from the former by means of a fixed-point argument. The former is a Binary Linear Program. We show that the latter can be reformulated exactly to a Mixed-Integer Linear Program (MILP) with binary variables, and also propose a cutting-plane algorithm to solve it in its native bilevel formulation. We benchmarked our solution methods on standard IEEE bus-systems [14] and randomly generated graphs.

2   Problem Statement

Let $G = (V, E)$ be a graph modelling the electrical network where $V = \{1, \ldots, n\}$ is the set of nodes representing the (sub)stations and E the set of edges corresponding to transmission lines. For $i \in V$, $\Gamma(i) = \{j : \{i, j\} \in E\}$ is the set of neighbours (adjacent nodes) of i. For graph-theoretical notions, see [7,13].

In this paper, we are interested in the optimal placement of PMUs with one channel, so as to ensure a full observability of G. We consider that no conventional measures exist, which would reduce the number of PMUs to install.

A PMU is placed on an edge $\{i, j\}$ close to node i, for $i \in V$ and $\{i, j\} \in E$.

We remark that the fact that PMU placement occurs closer to an adjacent node than the other is relevant for physical reasons, but irrelevant for our abstract modelling purposes. Henceforth, we shall simply assume that placement occurs on an edge $\{i, j\} \in E$. A graph is said to be observable if all node voltages and current edges are known either measured by a PMU or estimated using electrical laws. The problem is defined as follows:

POWER EDGE SET (PES) problem
Input: A graph $G = (V, E)$.
Output: A PMU placement $\Pi \subset E$ of minimum cardinality such that G is fully observable.

2.1   Observability of a Graph

Let Π be a given PMU placement on G and Ω the set of observed nodes. The observability of G is defined by the two following rules based on electrical laws explained below.

$R1$: If a PMU is placed on an edge $\{i, j\}$, then nodes i and j are observed

$$\{i, j\} \in \Pi \Rightarrow i, j \in \Omega$$

$R2$: If an observed node i has all its neighbors observed, save one, then this node is observed

$$i \in \Omega \text{ and } |\Gamma(i) \setminus \Omega| \leq 1 \Rightarrow \Gamma(i) \subseteq \Omega$$

By rule $R1$, the PMU placed at $\{i, j\}$ measures the voltage at i and the current on $\{i, j\}$. Using Ohm's law, we can deduce the voltage on j. Then i and j are observed. By rule $R2$, if a node i and all its neighbors $k \in \Gamma(i)$ are observed, except a single node j, then using Ohm's law we can determine the current on $\{i, k\}$ for $k \in \Gamma(i) \setminus \{j\}$; knowing the currents on all $\{i, k\}$ (for $k \neq j$) we can deduce the current on $\{i, j\}$ using Kirchoff's law. Then, knowing the voltage at i and the current on $\{i, j\}$, we determine the voltage on j using Ohm's law. Hence, j is observed.

3   Mathematical Modelling

We present two Mathematical Programming (MP) models for PES problem: an iterative based model, and a bilevel one.

3.1   Iterative Model

The model is based on the iterative process of observability given by rules $R1$ and $R2$. Assuming the problem instance to be a feasible one, the set of observed nodes can be found in at most $n-2$ steps: this is because, in the worst case, only one PMU is placed on an edge (this observes the adjacent nodes), and one more node is observed at each iteration. Let $T = n-2$ be the maximum number of steps. The iterative model (P_{IT}) is as follows.

Variables. We define the following set of variables:

$$\forall i \in V, j \in \Gamma(i), \quad s_{ij} = \begin{cases} 1 \text{ if a PMU is installed on } \{i,j\} \\ 0 \text{ otherwise} \end{cases}$$

$$\forall i \in V, t = 0, \ldots, T, \quad \omega_{it} = \begin{cases} 1 \text{ if the node } i \text{ is observed at step } t \\ 0 \text{ otherwise} \end{cases}$$

and $\forall i \in V, j \in \Gamma(i), t = 0, \ldots, T-1$,

$$y_{ijt} = \begin{cases} 1 \text{ if R2 is used to observe } j \text{ using the observed node } i \text{ at step } t \\ 0 \text{ otherwise.} \end{cases}$$

Constraints. The set of constraints is the following:

– All nodes must be observed at step T

$$\forall i \in V, \quad \omega_{iT} = 1$$

– If a node i is observed at step 0 then at least one PMU is placed at $\{i,j\}$ or $\{j,i\}$ for a given neighbour j of i

$$\forall i \in V, \quad \omega_{i0} \leq \sum_{j \in \Gamma(i)} (s_{ij} + s_{ji})$$

– The set of constraints corresponding to rule $R2$ is the following:
  • If i not observed at step t is observed at step $t+1$ then at least one neighbour observed node has been used to observe i

$$\forall i \in V, t = 0, \ldots, T-1, \quad \omega_{i(t+1)} \leq \omega_{it} + \sum_{\ell:i\in\Gamma(\ell)} y_{\ell it}$$

  • If an observed node i is used at step t to observe a neighbour node j and j is observed at step $t+1$ then j is not observed at step t

$$\forall i \in V, j \in \Gamma(i), t = 0, \ldots, T-1, \quad \omega_{j(t+1)} + y_{ijt} \leq \omega_{jt} + \omega_{it} + 1$$

- If an observed node i is used at step t to observe a neighbour node j and j is observed at step $t+1$ then j is not observed at step t and all the other neighbour nodes k of i are observed at step t

$$\forall i \in V, j, k \in \Gamma(i), k \neq j, t = 0, \ldots, T-1, \quad \omega_{j(t+1)} + y_{ijt} \leq \omega_{jt} + \omega_{kt} + 1$$

– If a node i is observed at step t then i is observed from step t to step T

$$\forall i \in V, t = 0, \ldots, T-1, \quad \omega_{it} \leq \omega_{i(t+1)}.$$

Objective Function. The aim of PES problem is to minimize the number of PMUs to install and that allow a full observability of G. Hence the objective function is given by

$$\min \sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij}.$$

3.2   From Iterative to Bilevel Model

We show in this subsection how to deduce a bilevel model from the iterative one using a fixed-point method.

Let $\omega^t = (\omega_{it} \mid i \in V)$ be the characteristic vector describing the observability of nodes at step t. The iterative model computes the vector values for $t \in \{1, \ldots, T\}$. Let $\omega = (\omega_i \mid i \in V)$ be the characteristic vector of Ω:

$$\forall i \in V, \quad \omega_i = \begin{cases} 1 \text{ if node } i \text{ is observed} \\ 0 \text{ otherwise.} \end{cases}$$

We have that $\omega = \omega^T$. We show now how to obtain a non iterative model where ω are the only variables that model the observability of the graph.

Let $t \leq T$ and i a node in V. The recursive relation that allows to express $\omega_{i(t+1)}$ in function of ω_t is:

$$\omega_{i(t+1)} = \max \left(\omega_{it}, \max_{j \in \Gamma(i)} \left(1 - |\Gamma(j)| + \omega_{jt} + \sum_{k \in \Gamma(j), k \neq i} \omega_{kt} \right) \right)$$

meaning that a node i is observed at step $t+1$ if it was already observed at step t or if there exists a neighbour j of i such that all the other neighbours $k \neq i$ of j are observed.

Let $\theta : \{0, 1\}^n \mapsto \{0, 1\}^n$ be a function where

$$\forall i \in V, \theta_i(x) = \max \left(x_i, \max_{j \in \Gamma(i)} \left(1 - |\Gamma(j)| + x_j + \sum_{k \in \Gamma(j), k \neq i} x_k \right) \right).$$

with $x = (x_i \mid i \in V)$ and $(\theta(x))_i = \theta_i(x)$.

By definition we have that:

$$\forall t \in \{1, \ldots, T-1\} \quad \omega^{(t+1)} = \theta(\omega^t).$$

Recursive Computation of ω: The vector ω is determined recursively as follows:

– Based on $R1$, for $t = 0$ we have:

$$\forall i \in V, \; \omega_{i0} = \max(\max_{j \in \Gamma(i)} s_{ij}, \; \max_{j \in \Gamma(i)} s_{ji})$$

– Knowing ω^t, we can compute $\omega^{(t+1)}$ by looking for an optimal solution of the linear program:

$$(*) \quad \begin{cases} \displaystyle\min_{\omega^{t+1} \in \{0,1\}^n} \sum_{i=1}^{n} \omega_{i(t+1)} \\ \quad \forall i \in V \qquad \omega_{i(t+1)} \geq \omega_{it} \\ \forall i \in V, \; j \in \Gamma(i) \qquad \omega_{i(t+1)} \geq 1 - |\Gamma(j)| + \omega_{jt} + \displaystyle\sum_{\substack{k \in \Gamma(j) \\ k \neq i}} \omega_{kt}. \end{cases}$$

Theorem 1. *We have that ω is the smallest fixed point of θ.*

Proof. By the definition of the function θ, we have that $\theta_i(\omega) \geq \omega_i, \forall i \in V$. Suppose that $\exists i \in V, \theta_i(\omega) > \omega_i$ i.e. $\omega_i = 0$ and $\theta_i(\omega) = 1$. Then $i \in \Omega$ by an application of $R2$ which implies $\omega_i = 1 > 0 = \omega_i$, contradiction. Hence $\theta(\omega) = \omega$.

Assume now that $\exists \omega' < \omega : \omega' = \theta(\omega')$. This means that $R2$ cannot be used to observed more nodes. Hence the number of nodes observed in ω' is less then the one in ω, i.e. $\sum_{i \in V} \omega'_i < \sum_{i \in V} \omega_i$ which contradict the optimality of $(*)$.

Therefore, ω is the smallest fixed point of θ and correspond to the optimal solution of the following linear program:

$$\begin{cases} \displaystyle\min_{\omega \in \{0,1\}^n} \sum_{i=1}^{n} \omega_i \\ \quad \omega_i \geq s_{ij} + s_{ji} & \forall i \in V, \; j \in \Gamma(i) \\ \quad \omega_i - \omega_j - \displaystyle\sum_{k \in \Gamma(j), k \neq i} \omega_k \geq 1 - |\Gamma(j)| & \forall i \in V, \; j \in \Gamma(i) \end{cases}$$

$\square$

3.3 Bilevel Model

We describe in this subsection the bilevel program proposed to model the PES problem. We also show how it can be reformulated to a MILP.

The formulation

$$
(\dagger)\;\begin{cases}
\min\limits_{s}\; \sum\limits_{i\in V}\sum\limits_{j\in \Gamma(i)} s_{ij} \\
\quad s_{ij} \in \{0,1\} \qquad \forall i \in V, j \in \Gamma(i) \\
\quad f(s) \geq n \\
f(s) = \begin{cases}
\min\limits_{\omega}\; \sum\limits_{i\in V} \omega_i \\
\omega_i \geq s_{ij} + s_{ji} & \forall i \in V, j \in \Gamma(i) \\
\omega_i - \omega_j - \sum\limits_{k\in \Gamma(j), k\neq i} \omega_k \geq 1 - |\Gamma(j)| & \forall i \in V, j \in \Gamma(i) \\
\omega_i \in \{0,1\} & \forall i \in V
\end{cases}
\end{cases}
$$

In the upper level problem, the objective is to minimize the number of PMUs to install such that the number of observed nodes given fy the function $f(s)$ is at least n. The function f corresponds to the optimal value of the lower level problem described below and s is the vector representing s_{ij}.

In the lower level problem, the objective is to minimize the number of nodes observed. The first set of constraints says that if a PMU is placed at $\{i,j\}$ or $\{j,i\}$ then i and j are observed. The second expresses the propagation rule $R2$: if a non observed node i has an observed neighbour j that has all its others node neighbours $k(k \neq i)$ observed then i is observed.

MILP Reformulation. The integrality of variables ω_i can be relaxed in the lower level problem.

Lemma 1. *For each $i \in V$, the constraint $\omega_i \in \{0,1\}$ can be replaced by $\omega_i \geq 0$.*

Proof. Let $\bar{\omega}$ be an optimal solution of the slave problem and consider a certain configuration of installed PMUs in the graph.

By the first constraint

$$\omega_i \geq s_{ij} + s_{ji}, \; \forall i \in V, j \in \Gamma(i) \tag{1}$$

we have that $\exists S \subseteq V, \forall i \in S, \bar{\omega}_i = 1$. If we rewrite the second constraint of the slave problem as

$$\forall i \in V, j \in \Gamma(i), \quad \omega_i \geq \omega_j + \sum_{k\in \Gamma(j), k\neq i} \omega_k - |\Gamma(j)| + 1$$

we have that the right hand side $r(\bar{\omega}) \in [1 - |\Gamma(j)|, 1]$.

If $r(\bar{\omega}) \in]0,1[$ then $\exists z \in \Gamma(j) \cup \{j\} : \bar{\omega}_z \in]0,1[$. Hence, $\exists Z \subseteq V, \forall z \in Z : \bar{\omega}_z \in]0,1[$. Also, $\forall z \in Z$, z is not constrained by (1) otherwise $\bar{\omega}_z = 1$. By the objective function direction, $\forall z \in Z$ we can set $\bar{\omega}_z = 0$ and still be feasible, which contradict the optimality of $\bar{\omega}$. Therefore, we can relax the integrity of variables ω to $[0,1]^n$.

Similarly, we can prove that $\forall i \in V, \omega_i \geq 0$. $\qquad\square$

Hence, by replacing the lower-level problem by its KarushKuhnTucker (KKT) conditions [3], we obtain the following MILP:

$$
(P) \begin{cases}
\min_{s} \sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij} & \\
\\
s_{ij} \in \{0, 1\} & \forall i \in V, j \in \Gamma(i) \\
\\
\sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij}\mu_{ij} + s_{ji}\mu_{ij} + (1 - |\Gamma(j)|)\lambda_{ij} \geq n & \\
\\
\sum_{j \in \Gamma(i)} \left(\mu_{ij} + \lambda_{ij} - \lambda_{ji} - \sum_{k \in \Gamma(j), k \neq i} \lambda_{kj} \right) \leq 1 & \forall i \in V \\
\\
\lambda_{ij}, \mu_{ij} \geq 0 & \forall i \in V, j \in \Gamma(i)
\end{cases}
$$

We now prove that the dual variables μ_{ij} are bounded, $\forall i \in V, j \in \Gamma(i)$.

Proposition 1. $\forall i \in V,\ j \in \Gamma(i),\ \exists M > 0 :\ \mu_{ij} \leq M.$

Proof. Let (s^*, μ^*, λ^*) be an optimal solution of (P) and (s^*, ω^*) be the corresponding optimal solution of the bilevel formulation. In particular, we consider (s^*, μ^*, λ^*) such that (μ^*, λ^*) is a basis solution of the dual program of the linear program that defines f:

$$
\begin{cases}
\max_{s} \sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij}\mu_{ij} + s_{ji}\mu_{ij} + (1 - |\Gamma(j)|)\lambda_{ij} & \\
\sum_{j \in \Gamma(i)} \left(\mu_{ij} + \lambda_{ij} - \lambda_{ji} - \sum_{k \in \Gamma(j), k \neq i} \lambda_{kj} \right) \leq 1\ \forall i \in V & \\
\lambda_{ij}, \mu_{ij} \geq 0 & \forall i \in V, j \in \Gamma(i)
\end{cases}
$$

Necessarily at most n dual variables are non-zero. Let $I = \{(i, j) \mid \mu_{ij} \neq 0\}$ and $J = \{(i, j) \mid \lambda_{ij} \neq 0\}$. We have $|I| + |J| \leq n$.

Let $i \in \{1, ..., n\}$ such that $\omega_i^* = 1$. By complementary slackness conditions we have

$$
\sum_{j \in \Gamma(i)} \left(\mu_{ij}^* + \lambda_{ij}^* - \lambda_{ji}^* - \sum_{k \in \Gamma(j), k \neq i} \lambda_{kj}^* \right) = 1 \tag{2}
$$

Let $A_B \in \mathbb{R}^{n \times n}$ be the basis matrix corresponding to the optimal solution (μ^*, λ^*). By Eq. (2), $v = (\mu^*, \lambda^*, \beta^*)$ is a solution of the system $A_B\, v = e$, where β^* denotes the slack variables used to write the above dual program in standard form, e is a vector in $\mathbb{R}^n$ where each component is one, and all elements of A_B are in $\{-1, 0, 1\}$.

Since $A_B^{-1} = \frac{\text{adj}(A_B)}{\det(A_B)}$, where $\text{adj}(A_B)$ is the adjugate matrix of A_B and $det(A_B)$ is the determinant of A_B, using Hadamard inequality for determinant, we obtain that the dual variables μ_{ij} are all bounded by $M = n^{\frac{n}{2}}$, $\forall i \in V, j \in \Gamma(i)$. $\qquad \square$

By Proposition 1, we can linearize the program (P) by replacing the variable products by $\forall i \in V, j \in \Gamma(i)$, $p_{ij} = s_{ij}\mu_{ij}$ and $q_{ij} = s_{ji}\mu_{ij}$. Therefore, we obtain the MILP (P_{MILP}).

$$(P_{\mathsf{MILP}}) \begin{cases} \min \sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij} \\ s_{i,j} \in \{0, 1\} & \forall i \in V, j \in \Gamma(i) \\ \sum_{i \in V} \sum_{j \in \Gamma(i)} p_{ij} + q_{ij} + (1 - |\Gamma(j)|)\lambda_{ij} \geq n \\ \sum_{j \in \Gamma(i)} (\mu_{ij} + \lambda_{ij} - \lambda_{ji} - \sum_{k \in \Gamma(j), k \neq i} \lambda_{kj}) \leq 1 \; \forall i \in V \\ p_{ij} \leq M \, s_{ij} & \forall i \in V, j \in \Gamma(i) \\ p_{ij} \leq \mu_{ij} & \forall i \in V, j \in \Gamma(i) \\ p_{ij} \geq \mu_{ij} - M(1 - s_{ij}) & \forall i \in V, j \in \Gamma(i) \\ q_{ij} \leq M \, s_{ji} & \forall i \in V, j \in \Gamma(i) \\ q_{ij} \leq \mu_{ij} & \forall i \in V, j \in \Gamma(i) \\ q_{ij} \geq \mu_{ij} - M(1 - s_{ji}) & \forall i \in V, j \in \Gamma(i) \\ \lambda_{ij}, \mu_{ij} \geq 0 & \forall i \in V, j \in \Gamma(i) \end{cases}$$

4   An Algorithm for the Bilevel Problem

We propose a cutting plane algorithm BiLevelSolve to solve the bilevel program (†) directly. BiLevelSolve iteratively solves a modified version of the upper level problem as a *master* MILP, adding a new cut at each iteration. The cuts are generated by means of the combinatorial procedure GenerateCut on the lower level *slave* problem.

Consider the following MILP P^k:

$$[P^k] \quad \begin{cases} \min_{s \in \{0,1\}^{|E|}} \sum_{i \in V} \sum_{j \in \Gamma(i)} s_{ij} \\ \forall h \leq k \qquad\qquad \alpha^h s \geq 1, \end{cases} \tag{3}$$

where $\alpha^h \in \{0, 1\}^{|E|}$ for each $h \leq k$, and k is the main algorithm iteration counter: at iteration k, P^k has k linear covering constraint, starting with $\alpha^1 = (1, \ldots, 1)$.

Although BiLevelSolve needs exponentially many cuts in the worst case, we found it to perform very well empirically.

5   Computational Results

All the experimentations presented here were performed on a 2.70 GHz computer with 8.0 GB RAM. The models (P_{IT}), (P_{MILP}) and the bilevel algorithm were implemented using IBM ILOG CPLEX 12.6. We considered as instances a 5-bus system and standard IEEE n-bus systems, with $n \in \{7, 14, 30, 57, 118\}$ [14]. We also generate randomly graphs with n nodes and $m = 1.4 \times n$ for $n = \{5 \times i, i = 1, \ldots, 10\}$ where 1.4 is the average rate of edges over nodes in standard IEEE

Algorithm 1. BiLevelSolve

1: $k = 1$
2: termination $\leftarrow 0$
3: **while** termination $= 0$ **do**
4:     $s \leftarrow$ MILPSolve(P^k)
5:     $k \leftarrow k + 1$
6:     $\alpha^k =$ GenerateCut$(s, \text{termination})$
7:     $P^k \leftarrow [P^{k-1} \text{ s.t. } \alpha^k s \geq 1]$
8: **end while**

Algorithm 2. GenerateCut$(s, \text{termination})$

1: termination $\leftarrow 0$
2: *// observe nodes according to PMUs in s*
3: place PMUs in G in all edges in the support of s
4: apply rules R1 and R2 to G, to obtain $\Omega \subseteq V$ (observed nodes)
5: **if** $\Omega = V$ **then**
6:     *// if PMUs in s suffice to observe all nodes, terminate*
7:     termination $\leftarrow 1$
8:     $\alpha \leftarrow (0, \ldots, 0)$
9: **else**
10:     *// otherwise, apply more PMUs and aim to observe all nodes*
11:     $\Theta \leftarrow \Omega$
12:     **while** $\Omega \subsetneq V$ **do**
13:         choose $v \in V \, setminus \, \Omega$ and $\{u, v\} \in E$
14:         place PMU in $\{u, v\}$ and apply R1, R2 to update Ω
15:         **if** $\Omega \neq V$ **then**
16:             $\Theta \leftarrow \Omega$
17:         **end if**
18:     **end while**
19:     *// generate cut on edges not induced by nodes observed*
20:     *// at R2 application step before full observability*
21:     let F be the set of edges induced by Θ
22:     let α be the support of $E \setminus F$
23: **end if**
24: **return** $(\alpha, \text{termination})$

---

bus systems. The instances can be forests and no node is isolated. For each value of n, 10 different instances were generated and tested. The results obtained are reported in Table 1 where each given value for the randomly generated graphs is the average over 10 instances. We limited the running time to 2 h. For any instance which is not solved optimally within the time limit, the running time is set to this limit. We reported: (i) the *Gap*, expressed as a percentage, that is the average over ratios $\frac{UB - BS}{UB}$ computed on all instances returning at least one feasible solution, where UB is the final best upper bound and BS is the best

solution value found; and (ii) the number of instances #opt solved optimally, and the number of instances that run out of memory (mof, for memory overflow).

Table 1. Computational results

Networks/graphs	n	m	Iterative			MILP			Bilevel algo.		
			Time (s)	Gap (%)	#opt (mof)	Time (s)	Gap (%)	#opt (mof)	Time (s)	Gap (%)	#opt (mof)
IEEE bus system	5	6	1.30	0	1	**1.29**	0	1	2.01	0	1
	7	8	6.22	0	1	**1.33**	0	1	2.14	0	1
	14	20	19.30	0	1	**1.38**	0	1	2.17	0	1
	30	41	7200	100	0	40.93	0	1	**2.37**	0	1
	57	80	7200	100	0	7200	63.51	0	**4.26**	0	1
	118	176	7200	100	0	7200	100	0	**247.41**	0	1
Rand. gen. graphs	5	7	**1.30**	0	10	6.42	0	10	2.09	0	10
	10	14	2.20	0	10	**1.37**	0	10	2.16	0	10
	15	21	297.60	0	10	**1.48**	0	10	2.18	0	10
	20	28	7200	69.89	0	2.34	0	10	**2.26**	0	10
	25	35	7200	96.33	0	29.16	0	10	**2.64**	0	10
	30	42	7200	95.71	0	1820.58	5.31	8	**4.86**	0	10
	35	49	*7200*	*98.14*	0(1)	3789.77	16.81	6	**15.26**	0	10
	40	56	7200	93.11	0	*6316.36*	*33.56*	1(5)	**24.34**	0	10
	45	62	7200	98.57	0	7200	47.14	0	**148.58**	0	9
	50	70	7200	93.16	0	*7200*	*50.36*	0(4)	**414.24**	0	7

italics: average over instances that did not run out of memory

We note that the iterative model cannot be used to solve medium and larger size instances. The MILP model can solve instances with more larger size than the iterative one but cannot solve large size instances. The bilevel algorithm can solve almost all the instances considered in few seconds. It did not solve only 4 instances of the random generated graphs considered within the time limit. For small instances, MILP performs a little better than the bilevel. This is due to the choice of the solution selected at each iteration to generate the cutting plane in the bilevel algorithm. Hence some iterations may be needed to converge to the optimal solution in the bilevel algorithm while in the MILP model, having a small number of variables and constraints for those instances, the model converges in few seconds.

Therefore the bilevel algorithm is better in terms of running time and size of instances that can be solved.

Remark 1. We assumed here that the installation cost is the same for every PMU location at a node along an edge. If not, the problem consists then in finding the placement of PMUs that ensures a full observability of the graph and minimize the total installation cost. Let, $\forall i \in V, j \in \Gamma(i)$, c_{ij} be the cost of installing a PMU on $\{i, j\}$ at i. The new objective function is then given by:

$$\min \sum_{i \in V} \sum_{j \in \Gamma(i)} c_{ij}\, s_{ij}$$

Remark 2. The models proposed for PMU PLACEMENT PROBLEM, where PMUs are with unlimited number of channels and hence the placement is done on nodes, do not consider the propagation rule too. Our proposed models can easily be adapted to this node version.

6 Conclusions

We presented a new approach to model PES problem using a propagation rule based on Ohm's and Kirchoff's laws to reduce the number of PMUs to place. We proposed two mathematical models: an iterative model and a bilevel one. The iterative model is based on the observability propagation process and is given by a binary linear program. The bilevel model is deduced from the iterative one using fixed point method. We showed that we can transform the bilevel model to a MILP. We proposed also an algorithm to solve the bilevel model. We implemented these models and algorithm for the bilevel program and we performed tests on different IEEE bus systems and randomly generated graphs. The results showed that: the iterative model cannot be used for medium and large instances; the MILP model can solve instances with more large size than the iterative one but cannot solve large size instances; and the bilevel algorithm can solve instances with large sizes. Therefore, the bilevel algorithm is better in terms of running time and size of instances that can be solved. Further future work could be to model the case of conventional measures. We can also consider the case of line outage and single contingency of PMUs. Another further future work would be to generalize our models for the case of PMUs with limited channels ℓ. Also, due to maintenance or repairing works the electrical network topology is not fixed. Hence, another interesting perspective is to study the PMU PLACEMENT PROBLEM under these conditions by proposing a robust model and a solution method to solve it.

References

1. Aazami, A., Stilp, M.D.: Approximation algorithms and hardness for domination with propagation. In: Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P. (eds.) RANDOM 2007 and APPROX 2007. LNCS, vol. 4627, pp. 1–15. Springer, Heidelberg (2007)
2. Brueni, D.J., Heath, L.: The PMU placement problem. SIAM J. Discrete Math. **19**(3), 744–761 (2005)
3. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
4. Dorfling, M., Henning, M.A.: A note on power domination in grid graphs. Discrete Appl. Math. **154**, 1023–1027 (2006)

5. Emami, R., Abur, A.: Robust measurement design by placing synchronized phasor measurements on network branches. IEEE Trans. Power Syst. **25**(1), 38–43 (2010)
6. Emami, R., Abur, A., Galvan, F.: Optimal placement of phasor measurements for enhanced state estimate: a case study. In: Proceedings of the 16th Power Systems Computation Conference, Glasgow, Scotland, pp. 1–6, 14–18 July 2008
7. Gross, J.L., Yellen, J.: Handbook of Graph Theory. Discrete Mathematics and Its Applications, vol. 25. CRC Press, Boca Raton (2003)
8. Haynes, T.W., Hedetniemi, S.M., Hedetniemi, S.T., Henning, M.A.: Domination in graphs applied to electric power networks. SIAM J. Discrete Math. **15**(4), 519–529 (2002)
9. Korkali, M., Abur, A.: Placement of PMUs with channel limits. In: IEEE Power and Energy Society General Meeting (2009)
10. Kumar, R., Rao, V.S.: Optimal placement of PMUs with limited number of channels. In: Proceedings of North American Power Symposium (NAPS), Boston, MA, pp. 1–7, 4–6 August 2011
11. Manousakis, N.M., Korres, G.N., Georgilakis, P.S.: Optimal placement of phasor measurement units: a literature review. In: Proceedings of the 16th International Conference on Intelligent System Application to Power Systems, ISAP 2011, Hersonissos, Greece, pp. 1–6, 25–28 September 2011
12. Manousakis, N.M., Korres, G.N., Georgilakis, P.S.: Taxonomy of PMU placement methodologies. IEEE Trans. Power Syst. **27**(2), 1070–1077 (2012)
13. West, D.B.: Introduction to Graph Theory, 2nd edn. Prentice Hall, Englewood Cliffs (2000)
14. Zimmerman, R.D., Murillo-Sánchez, C.E.: MATPOWER. http://www.pserc.cornell.edu/matpower/