A scalable mixed-integer decomposition approach for optimal power system restoration

Ignacio Aravena, *Member, IEEE,* Deepak Rajan, Georgios Patsakis, *Student, IEEE,* Shmuel S. Oren, *Life Fellow, IEEE,* and Jennifer Rios

***Abstract*—The optimal restoration problem lies at the foundation of the evaluation and improvement of resilience in power systems. In this paper we present a scalable decomposition algorithm, based on the integer L-shaped method, for solving this problem for realistic power systems. The algorithm works by partitioning the problem into a master problem and a slave problem. The master problem optimizes over energization sequences of generators, buses and branches, for the entire restoration horizon, while respecting stylized energization requirements. The slave problem verifies that there exist power flow solutions supporting the energization sequences proposed by the master problem, adding cuts to the master if an infeasible sequence is detected. The power flow verification is carried out using a piece-wise linear approximation of the power flow equations, capable of capturing overvoltage situations common in restoration procedures. We develop specialized binary cuts and hybrid Benders-binary cuts for separating infeasible islands, which are stronger than regular no-good cuts, as well as heuristics for obtaining feasible solutions. We present numerical results for the IEEE 39- and 118-bus systems, and for the Chilean 1500-bus grid, where our algorithm finds near-optimal restoration sequences orders of magnitude faster than the state-of-the-art, demonstrating the effectiveness of the proposed approach.**

***Index Terms*—Power system restoration, power system resilience, integer L-shaped method, power flow approximations**

I. Introduction

POWER system operation practices are undergoing a major transformation following the integration of renewable energy resources and information technologies into the grid. This transformation, alongside with the advent of increasingly frequent extreme weather events, has made power system operations vulnerable to large forecast errors, cyberattacks and natural disasters. These events can result in large-scale outages, which can lead to billions of dollars in damages and pose serious threats to health and public safety, among other consequences [1]. In response, system planers and operators are currently reviewing their practices for ensuring power system resilience [1], [2], that is, the ability to withstand and recover from disruptive events [3].

Understanding optimal power system restoration (OPSR) is crucial to improving resilience. At an operational level, OPSR can provide significant improvements in restoration speed and robustness over fixed restoration priorities [4]. At an assessment and planning level, OPSR provides a natural definition and calculation method for most resilience metrics [1, Table 2.2] using simulation, as well as an objective tool for estimating improvements in resilience following changes in equipment and technologies.

In short, OPSR consist on deciding the sequence of energization steps that would bring the grid back to normal operation, as fast as possible, after a blackout. From a computational perspective, it corresponds to a mixed-integer nonlinear program (MINLP), where binary variables represent energization decisions (on/off switches of branches, buses, generators, and other grid components) and nonlinear constraints model power flows throughout the network, at several snapshots of a given horizon. OPSR has been continuously addressed by the technical literature over, at least, the last 30 years [5], [6], using mostly heuristics and meta-heuristics to obtain feasible solutions to the problem [7]. These approaches, in general, do not provide guarantees on solution quality (i.e. optimality gaps), however they allow to consider very detailed models of the power grid such as stability and transient dynamics [8].

Mathematical programming methods for OPSR, providing both feasible solutions and quality guarantees, have become increasingly popular in recent years and are the focus of this paper. These methods are commonly based on mixed-integer linear programming and approximations or relaxations of the power flow equations. In particular, Sun *et al.* [9] formulated and solved a mixed-integer linear program (MILP) to optimize the startup sequence of generators assuming a copper plate model for the grid. Van Hentenryck and Coffrin [4] solve the OPSR problem with a linear programming approximation of the AC power flow equations [10] on a network with 266 components (73 buses, 106 branches, 33 generators and 54 loads) with different partial outage scenarios with up to 61 de-energized components. Arab *et al.* [11] use the same approximation of the power flow equations to solve OPSR along with the reparation scheduling problem for the IEEE118 test system (471 components: 118 buses, 186 branches, 14 shunt compensators, 54 generators and 99 loads) with 12 de-energized components. Patsakis *et al.* [12] present a model decides where to allocate blackstart units (i.e. generators with the capabilities to startup without connecting to a previously energized grid) which includes OPSR. The authors solve the blackstart allocation model using a linear approximation of the power flow equations and considering a full blackout scenario for the IEEE39 test system (116 components: 39 buses, 46 branches, 10 generators and 21 loads), the IEEE118 test system and a simplified WECC system (871 components:

I. Aravena and D. Rajan are with the Lawrence Livermore National Laboratory. Email: aravenasolis1@llnl.gov, rajan3@llnl.gov

G. Patsakis and S.S. Oren are with the University of California at Berkeley. Email: gpatsakis@berkeley.edu, oren@ieor.berkeley.edu.

J. Rios is with PG&E. Email: LJDb@pge.com.

225 buses, 371 branches, 151 generators and 124 loads).

Several authors have also proposed decomposition schemes for the OPSR problem, all of them using linear programming approximations of the power flow equations. Arab *et al.* [13] propose the use of Benders decomposition for OPSR and use their algorithm to solve an instance of the IEEE118 system with 16 de-energized components. Golshani *et al.* [14] consider the stochastic OPSR problem with energization of renewable resources during restoration. The authors propose a variant of the L-shaped method with an approximate (invalid) optimality cut and use it to solve OPSR from a full blackout on the IEEE57 test system (186 components: 57 buses, 80 branches, 7 generators and 42 loads).

The state-of-the-art in OPSR, in summary, is capable of solving only small stylized instances. The present paper seeks to extend the state-of-the-art in OPSR by presenting an algorithm that can handle realistic instances, which are more complex and larger than previously studied instances [4], [9], [11], [13], [14], [12], while still providing optimality guarantees. In particular, we present a specialized decomposition algorithm based on the integer L-shaped method [15], [16] that splits the problem into energization decisions and power flow feasibility. We develop specialized binary cuts and hybrid Benders-binary cuts for separating infeasible islands, which are stronger than regular L-shaped infeasibility cuts. We also propose heuristics for computing an initial solution to OPSR and for obtaining candidate solutions by rounding nodes of the branch-and-cut tree. In comparison, the state-of-the-art [13], [14] only uses plain versions of Benders and the L-shaped algorithms, or only computes initial solutions with the help of heuristics [12]. We present numerical experiments for OPSR from a full blackout on the IEEE39 and IEEE118 test systems, and on the Chilean SIC system (3696 components: 1548 buses, 1114 lines, 433 2-winding transformers, 131 3-winding transformers, 14 series compensators, 159 shunt compensators, 297 generators and 605 loads), which is 5 times larger and requires a much more detailed power flow feasibility model than the state-of-the-art.

The rest of the paper is organized as follows. Section II describes the current practices followed by operators during power system restoration and assumptions that can be derived from them. Section III presents our formulation for the OPSR problem. Section IV presents the proposed algorithm for OPSR, including separation oracles, infeasibility cuts and heuristics. Section V presents details of our implementation and numerical results for the 3 aforementioned systems. Section VI summarizes the main findings of the paper and presents directions for future research.

II. Current practices in power system restoration

Restoration plans of different power systems differ significantly from one another (see, for example, [17], [18], [19], [20]), which can be attributed to (*i*) technical characteristics of the grid and generating units, (*ii*) the most likely causes of blackouts on each system and (*iii*) particular governance structures.

Despite these differences, most restoration plans share an underlying divide-and-conquer strategy [1]. After a blackout, the system is partitioned into several pre-determined areas, each of which is under the control of a single Control Center (CC). Each CC assesses the damage at generating units, substations and, when possible, transmission lines within its area (the state of certain transmission lines, e.g. long lines, can only be known after an attempt of energization). The CC then selects how to restore service from a list of alternative restoration plans, constructed and validated offline for each area, or it devises a new plan based on the damage evaluation. These restoration plans might be adapted during implementation and their primary goal is to build a stable island on each area. A plan indicates at least (*i*) which blackstart units to use, (*ii*) switching sequences for energizing cranking paths towards larger units and (*iii*) when to start serving critical loads, such as government facilities, hospitals, law enforcement and nuclear power plants [17], [18]. Other consumers, at this stage, are only served to help the CC in maintaining the stability of the island or maintaining voltage within technical limits.

Depending on the operation rules of each system, utility-scale renewable resources may either only be used once an island is deemed stable by the corresponding CC [18] or they may not be considered at any stage of the restoration process [2]. These policies, however, would require revision as the energy transition to renewable energy forces conventional power plants to retire [19].

Neighboring CCs that have completed their restoration plans can then proceed to interconnect. The interconnection process is commonly coordinated by or monitored by a third-party, commonly an Independent System Operator (ISO) or an alike organization, in matters including synchronization, frequency and voltage regulation, and additional load pickup. Interconnected areas are progressively connected among them, until the systems forms a single stable island. Finally, the rest of the consumers are incrementally connected to the grid, restoring normal operations except for zones where major damages need to be repaired.

III. Problem formulation

We model OPSR from the perspective of an optimistic planner, making the following assumptions, which are based both on the current literature and industry practices. First, we assume that a centralized organization has full observability and control of the grid. This assumption, customary in the academic literature [6], [7], [12], [14], implies (*i*) that the network is in a known state, ignoring that the plan might need to be adapted during implementation due to unforeseen line unavailabilities, and (*ii*) that switching actions can be commanded in a centralized fashion, ignoring that plans that cut across multiple CC areas might be difficult to implement in practice. Second, following the current practice in several systems, we assume that renewable generating units (wind and solar) are not allowed to inject power into the grid during restoration, making the operation of the grid a deterministic process. Third, following the academic literature, we assume that dynamic phenomena and limits below the temporal resolution of our model (15 minutes to 1 hour), including ramp rate limits, electro-mechanic and electromagnetic transients, have a neglectable effect in constraining the restoration plan.

The first two assumptions allow us to formulate OPSR as a deterministic MINLP, whereas the third assumption forces additional structure into OPSR, which we later exploit in our algorithm. In what follows we present our OPSR model by parts: subsection III-A presents the constraints that govern the energization of transmission system components and the startup of generators, subsection III-B presents the constraints that must be respected in order to maintain steady-state feasibility and subsection III-C explains our objective function. Although present in our implementation and experiments, for brevity, we omit the variables and constraints modeling power tranformers. The nomenclature used in this section and the rest of the paper is detailed in appendix A.

A. Constraint set I: energization

We represent the energization status of any component in the system using the binary variable u, with appropriate subindexes, which takes the value 1 if the component is energized and 0 otherwise. Additionally, we use binary variables u^{CR} to indicate whether generators are cranking or not. Our formulation includes initialization parameters $u_{\cdot,t}$ for $t < \underline{T}$ to model the initial conditions of the restoration process.

Energization requirements for generators are modeled as constraints (1). Constraints (1a) enforce that generators are either offline ($u_{g,t} = 0, u^{CR}_{g,t} = 0$), cranking ($u_{g,t} = 1, u^{CR}_{g,t} = 1$) or online ($u_{g,t} = 1, u^{CR}_{g,t} = 0$). Constraints (1b) ensure that an offline generator must be cranked across at least CT_g snapshots before being online, while constraints (1c) require that non-blackstart generators are connected to an energized bus to be cranked. We implement the positive part operator $(\cdot)^+$ using its standard linear programming reformulation, which we omit here for brevity. Constraint (1d) captures the fact that any online generator energizes its connection bus.

$$u^{CR}_{g,t} \leq u_{g,t} \quad \forall g \in G, t \in T \tag{1a}$$

$$\sum_{\tau=t-CT_g+1}^{t} (u_{g,\tau} - u_{g,\tau-1})^+ \leq u^{CR}_{g,t} \quad \forall g \in G, t \in T \tag{1b}$$

$$u^{CR}_{g,t} \leq u_{n(g),t} \quad \forall g \in G \setminus G^{BS}, t \in T \tag{1c}$$

$$u_{g,t} - u^{CR}_{g,t} \leq u_{n(g),t} \quad \forall g \in G, t \in T \tag{1d}$$

A necessary requirement for a bus to be energized is that it is connected through an energized path to a generator. Following [21], we formulate the bus energization requirement as constraints (2). These constraints demand that for each subset of buses M, a bus $n \in M$ can only be energized if there is an online generator connected to M or if a branch (i.e. a line, transformer or series compensator) in the boundary ∂M of M (i.e. branches with one terminal in M and one terminal outside M) is energized.

$$\begin{gathered} u_{n,t} \leq \sum_{g \in G(M)} (u_{g,t} - u^{CR}_{g,t}) + \sum_{l \in L(\partial M)} u_{l,t} \\ \forall M \subseteq N : SC(\partial M) = \emptyset, n \in M, t \in T \end{gathered} \tag{2}$$

There are exponentially many constraints (2) in terms of $|N|$ but, luckily, they can be separated in polynomial time [21].

A line can only be energized if one of its terminal buses was energized in the previous snapshot. Once a line is energized, all of its terminal buses must be energized. These restrictions can be formulated as constraints (3), where constraints (3a) ensure the sequential progression of the energization process.

$$u_{l,t} \leq \sum_{i \in I(l)} u_{n(i),t-1} \quad \forall l \in L, t \in T \tag{3a}$$

$$u_{l,t} \leq u_{n(i),t} \quad \forall l \in L, i \in I(l), t \in T \tag{3b}$$

Series compensators differ from lines and transformers in that they are not open when de-energized, instead they are short-circuited. This implies that energization status of buses at terminals of series compensators must be equal at all times. We model this requirement as constraints (4).

$$u_{n(i),t} = u_{n(j),t} \quad \forall sc \in SC, \{i, j\} = I(sc), t \in T \tag{4}$$

In practice, operators must pause between switching maneuvers in order to verify that the new stationary operating point is stable, leading to constraint (5), which limits the number of branches that can be energized in between snapshots.

$$\sum_{l \in L} (u_{l,t} - u_{l,t-1})^+ + \sum_{sc \in SC} (u_{sc,t} - u_{sc,t-1})^+ \leq K \quad \forall t \in T \tag{5}$$

We set the parameter K as the resolution of the OPSR model, assuming at most one branch energization per minute.

B. Constraint set II: power flow feasibility

In what follows, we use domains S and power flow functions P and Q to describe our formulation of the AC power flow equations concisely while capturing the level of detail of our implementation, necessary to model realistic networks. The description of these domains and functions corresponds to mixed-integer linear expressions that approximate the actual equations of the grid. We refer the reader to [22] for a detailed explanation of this approximation.

We model generators using constraints (6), which enforce that active and reactive power output belong in the operation domain of each generator, at its corresponding status.

$$(p_{g,t}, q_{g,t}) \in S_g(u_{g,t}, u^{CR}_{g,t}) \quad \forall g \in G, \forall t \in T \tag{6}$$

Constraints (7) ensure that energized buses have voltages between $\underline{V}$ and $\overline{V}$, commonly 0.9pu and 1.1pu, while de-energized buses have no voltage.

$$\underline{V} u_{n,t} \leq v_{n,t} \leq \overline{V} u_{n,t} \quad \forall n \in N, t \in T \tag{7}$$

Constraints (8a) impose apparent power limits on active and reactive power flows of transmission lines. Constraints (8b) model the AC power flow equations for energized lines, and sets active and reactive power flows to zero for de-energized (open) lines.

$$(p_{l,i,t}, q_{l,i,t}) \in S_l \quad \forall l \in L, i \in I(l), t \in T \tag{8a}$$

$$\begin{aligned} p_{l,i,t} &= u_{l,t} \cdot P_l(v_{n(i),t}, v_{n(j),t}, \theta_{n(i),t} - \theta_{n(j),t}), \\ q_{l,i,t} &= u_{l,t} \cdot Q_l(v_{n(i),t}, v_{n(j),t}, \theta_{n(i),t} - \theta_{n(j),t}) \\ &\forall l \in L, i, j \in I(l), i \neq j, t \in T \end{aligned} \tag{8b}$$

Series compensators are modeled in a similar fashion to lines. Constraints (9a) impose security limits on power entering the component, while constraints (9b) enforce the AC power flow equations, which become trivially satisfied when the series compensator is short-circuited. When this happens, constraints (9c) enforce the voltage on the buses of both terminals of the series compensator to be identical.

$$(p_{sc,i,t}, q_{sc,i,t}) \in S_{sc} \quad \forall sc \in SC, i \in I(sc), t \in T \tag{9a}$$

$$\begin{aligned} &u_{sc,t} \cdot p_{sc,i,t} = u_{sc,t} \cdot P_{sc}(v_{n(i),t}, v_{n(j),t}, \theta_{n(i),t} - \theta_{n(j),t}),\\ &\quad u_{sc,t} \cdot q_{sc,i,t} = u_{sc,t} \cdot Q_{sc}(v_{n(i),t}, v_{n(j),t}, \theta_{n(i),t} - \theta_{n(j),t})\\ &\quad \forall sc \in SC, i,j \in I(sc), i \neq j, t \in T \end{aligned} \tag{9b}$$

$$\begin{aligned} &-(\overline{V} - \underline{V})u_{sc,t} \leq v_{n(i),t} - v_{n(j),t} \leq (\overline{V} - \underline{V})u_{sc,t},\\ &\quad (1 - u_{sc,t}) \cdot (\theta_{n(i),t} - \theta_{n(j),t}) = 0\\ &\quad \forall sc \in SC, i,j \in I(sc), i \neq j, t \in T \end{aligned} \tag{9c}$$

In addition to series compensators, systems usually have shunt compensators, modeled according to (10). For variable shunts, controlled with power electronics, these constraints accurately represent the limits of the device, with $\underline{Q}_{sh}$ and $\overline{Q}_{sh}$ usually becoming constant on $v_{n(sh),t}$. For banks of reactors and banks of capacitors, on the other hand, these constraints ignore the discrete nature of the number of sections connected.

$$\underline{Q}_{sh}(v_{n(sh),t}) \leq q_{sh,t} \leq \overline{Q}_{sh}(v_{n(sh),t}) \quad \forall sh \in SH, t \in T \tag{10}$$

The power withdrawn by consumers must belong in a set S_c that depends both on the energization status and voltage magnitude of their connection bus, as indicated in (11). The set S_c allows consumers to be partially served when the bus $n(c)$ is energized, while it maintains a constant power factor for the load at constant voltage, thereby assuming that each consumer is homogeneous.

$$(p_{c,t}, q_{c,t}) \in S_c(u_{n(c),t}, v_{n(c),t}) \quad \forall c \in C, t \in T \tag{11}$$

Active and reactive power balance equations (12a) and (12b) complete the set of AC power flow restrictions.

$$\begin{aligned} &\sum_{g \in G(n)} p_{g,t} = \sum_{\substack{l \in L(n), i \in I(l)\\ n(i)=n}} p_{l,i,t} + \sum_{\substack{sc \in SC(n), i \in I(sc)\\ n(i)=n}} p_{sc,i,t} + \\ &\quad \sum_{c \in C(n)} p_{c,t} \quad \forall n \in N, t \in T \end{aligned} \tag{12a}$$

$$\begin{aligned} &\sum_{g \in G(n)} q_{g,t} = \sum_{\substack{l \in L(n), i \in I(l)\\ n(i)=n}} q_{l,i,t} + \sum_{\substack{sc \in SC(n), i \in I(sc)\\ n(i)=n}} q_{sc,i,t} + \\ &\quad \sum_{sh \in SH(n)} q_{sh,t} + \sum_{c \in C(n)} q_{c,t} \quad \forall n \in N, t \in T \end{aligned} \tag{12b}$$

C. Objective function

The goal of the restoration process is to return the power grid to a normal operation state, or as close as possible to it, in terms of stability and coverage. Loads, except critical loads, are a tool to achieve such goal. With this in mind, we use (13) as objective function.

$$\begin{aligned} \max_{\substack{u \in \{0,1\}\\ v,\theta,p,q}} \quad & \sum_{t \in T} \Bigg(\alpha \cdot \Xi\Bigg(\sum_{g \in G} J_g \cdot \left(u_{g,t} - u^{CR}_{g,t}\right) \Bigg) + \\ & \sum_{l \in L} \beta_l u_{l,t} + \sum_{n \in N} \beta_n u_{n,t} \Bigg) \end{aligned} \tag{13}$$

Here the term in the first line uses system inertia as a surrogate for stability and the terms in the second line use the energization status of branches and buses as a surrogate for coverage. Ξ is a piecewise concave increasing function that cuts off contributions above normal-operation inertia to the objective function. Parameters α and β are set following priorities for restoration of the system under consideration.

Assembling objective and constraints, the OPSR problem corresponds to the mathematical program (1) – (13). Products between variables (or functions of variables) in constraints (8b) and (9b) can be linearized exactly using McCormick envelopes [23]. This reformulation of (1) – (13) corresponds to a large-scale MILP (millions of variables and constraints for a realistic system), with a very poor relaxation of the power flow equations because of the large coefficients required by the McCormick envelopes. The size and complexity of the OPSR model cannot be handled by current commercial MILP solvers, motivating the development of the algorithm presented in the following section.

IV. DECOMPOSITION ALGORITHM

We decompose OPSR into a *Master* problem and a *Slave* subproblem, placed inside a *Feasibility Oracle*. The *Master* problem maximizes (13), considering only energization variables $\boldsymbol{u}$ and constraints (1), (2) with $|M| = 1$, (3) – (5). The *Slave* subproblem finds a feasible power flow solution, i.e. a solution to (2), (6) – (12), supporting a fixed energization plan $\bar{\boldsymbol{u}}$. The *Feasibility Oracle* receives $\bar{\boldsymbol{u}}$ from the *Master* problem and it determines whether a solution for the *Slave* subproblem exists. If no solution exists, the *Feasibility Oracle* provides cuts (separating inequalities) that exclude $\bar{\boldsymbol{u}}$ from the *Master* problem. The general outline of our decomposition algorithm is presented in Fig. 1. We make the algorithm practically efficient by expoiting two ideas in our implementation of the *Feasibility Oracle*.

First, we note that an energization plan $\bar{\boldsymbol{u}}$ corresponds to a sequence of electrical island configurations. Hence, $\bar{\boldsymbol{u}}$ will be feasible if and only if there exist a power flow solution for each island at every time step. This structure provides the following advantages: (*i*) we can check the feasibily of $\bar{\boldsymbol{u}}$ by evaluating its constituent islands indepentently, solving smaller subproblems and exploiting parallelism, (*ii*) we can generate island-based cuts that are a stronger than those used by the standard integer L-shaped method [15], (*iii*) once an island is marked as infeasible, we can use these cuts to exclude it from multiple time snapshots and not only from the ones where it appears in $\bar{\boldsymbol{u}}$, and (*iv*) we can maintain a hash table of all islands for which we have determined feasibility across oracle

Use *Initialization* (Sec. IV-B1) to construct a feasible point $\overline{\boldsymbol{u}}^0$
Let $\overline{\boldsymbol{u}}^{\text{last}} := \overline{\boldsymbol{u}}^0$
begin *Branch-and-Bound* on *Master*
    Start with $\overline{\boldsymbol{u}}^0$ as incumbent
    At every binary candidate $\overline{\boldsymbol{u}}$:
        Call *Feasibility Oracle* (Sec. IV-A) on $\overline{\boldsymbol{u}}$
        if $\overline{\boldsymbol{u}}$ is feasible **then**
            Update incumbent
        end if
    At every fractional candidate $\overline{\boldsymbol{u}}$:
        if $\|\overline{\boldsymbol{u}} - \overline{\boldsymbol{u}}^{\text{last}}\| \geq \delta$ **then**
            Let $\overline{\boldsymbol{u}}^{\text{last}} := \overline{\boldsymbol{u}}$ and call *Rounding* (Sec. IV-B2) on $\overline{\boldsymbol{u}}$
        end if
end

Fig. 1. Outer program of specialized integer L-shaped method. The *Branch-and-Bound* method traverses the binary tree of energization decisions, finding nodes with binary candidates, which are tested for feasibility, and nodes with fractional candidates, which are rounded to binary candidates (if sufficiently different from the last rounded fractional candidate $\bar{\boldsymbol{u}}^{\text{last}}$).

1: Let `energizable` := true
2: Detect unique islands implied by $\bar{\boldsymbol{u}}$ in all time periods, $\mathscr{I}(\bar{\boldsymbol{u}})$
3: **for** $\mathcal{I} \in \mathscr{I}(\bar{\boldsymbol{u}})$ **do**
4:     **if** $\sum_{g \in G(\mathcal{I})} (\bar{u}_g^{\mathcal{I}} - \bar{u}_g^{\mathcal{I},CR}) = 0$ **then**
5:         **for** $t \in \{\underline{T}^{\mathcal{I}} - W, \ldots, \overline{T}^{\mathcal{I}} + W\}$ **do**
6:             Add cuts (2), $\forall n \in N(\mathcal{I})$, and (14) to *Master*
7:             Let `energizable` := false
8:         **end for**
9:     **end if**
10: **end for**
11: **if not** `energizable` **then return**; **end if**
12: **for** $\mathcal{I} \in \mathscr{I}(\bar{\boldsymbol{u}})$ **do**
13:     Let `status`, μ^*, $\boldsymbol{\lambda}^*$:= *RelaxedPowerFlow*$(\mathcal{I})$
14:     **if** `status` = Infeasible **then**
15:         Add cut (15) for $t \in \{\underline{T}^{\mathcal{I}} - W, \ldots, \overline{T}^{\mathcal{I}} + W\}$ to *Master*
16:         Let $\mathscr{I}(\bar{\boldsymbol{u}}) := \mathscr{I}(\bar{\boldsymbol{u}}) \setminus \{\mathcal{I}\}$
17:     **end if**
18: **end for**
19: **for** $\mathcal{I} \in \mathscr{I}(\bar{\boldsymbol{u}})$ **do**
20:     **if** *PowerFlow*$(\mathcal{I})$ = Infeasible **then**
21:         Add cut (16) for $t \in \{\underline{T}^{\mathcal{I}} - W, \ldots, \overline{T}^{\mathcal{I}} + W\}$ to *Master*
22:     **end if**
23: **end for**

Fig. 2. *Feasibility Oracle*. Classes of infeasibility cuts are verified in increasing order of computation complexity for a giving binary $\bar{\boldsymbol{u}}$: detecting violated cuts (2), lines 2–12, is linear on the number of branches (it requires a graph traversal, line 2, and checking whether each island has at least one online generator, line 4); detecting islands with infeasible relaxed power flow equations, lines 13–19, requires solving a linear program; and detecting islands with infeasible power flow equations, lines 20–24, requires solving a mixed-integer linear program. Feasibility cuts for infeasible island $\mathcal{I}$ are applied to all snapshots within a window W of the first and last snapshot where $\mathcal{I}$ appeared in $\bar{\boldsymbol{u}}$, $\underline{T}^I$ and $\overline{T}^I$.

calls, avoiding duplicated evaluation of islands appearing in multiple energization plans.

Second, we note that it is possible to detect infeasible islands in $\bar{\boldsymbol{u}}$ by means of a hierarchy of increasingly complex tests. The hierarchy is composed by three tests: a graph traversal, a linear program and a MILP, each associated with a different type of cut: energization cuts, hybrid Benders-binary island-based cuts and binary island-based cuts. If an island is proven infeasible by an certain test, cuts can be added immediately to the *Master* to exclude it, avoiding harder tests and speeding up the execution of the *Feasibility Oracle*.

We further improve the perfomance of our algorithm by proposing specialized greedy heuristics to construct an initial solution and to round fractional solutions within the *Branch-and-Bound* tree of the *Master*.

In what follows, we describe in detail the *Feasibility Oracle* and specialized heuristics developed for the OPSR problem.

A. Feasibility Oracle

As indicated before, we check the feasibility of a restoration plan by evaluating each of its electrical islands separately. We formally define islands as pairs $\mathcal{I} := (M, \bar{\boldsymbol{u}})$ of a subset M of buses of the system and the state $\bar{\boldsymbol{u}}$ of branches and generators connected to these buses; where series compensators, and online lines and transformers form a connected graph including all buses in the subset, disconnected from the rest of the system. Note that, under this definition, two islands $\mathcal{I}$ and $\mathcal{J}$ with the same buses $N(\mathcal{I}) = N(\mathcal{J})$ can be different if the status of components within the islands differ, i.e. $\bar{\boldsymbol{u}}^{\mathcal{I}} \neq \bar{\boldsymbol{u}}^{\mathcal{J}} \implies \mathcal{I} \neq \mathcal{J}$.

Fig. 2 presents the operations carried out by the *Feasibility Oracle*. The oracle uses 4 classes of infeasibility cuts: 2 energization cuts, 1 hybrid Benders-binary island-based cut and 1 binary island-based cut.

1) Energization cuts: Following [21], we add cuts (2) for each bus in islands without online generators.

Additionally, for each of these islands we add the strengthening inequalities (14), which are valid for any subset of buses M. This novel class of constraints has been inferred from the convex hull of small networks, computed numerically using CDD [24] through the Julia/Polyhedra interface [25].

$$\sum_{n \in M} u_{n,t} - \sum_{l \in L(M)} u_{l,t} - |SC(M)| \leq \sum_{g \in G(M)} (u_{g,t} - u_{g,t}^{CR}) + \sum_{l \in L(\partial M)} u_{l,t} + |SC(\partial M)| \quad \forall M \subseteq N, t \in T \tag{14}$$

2) Hybrid Benders-binary island-based cuts: Once verified that constraints (2) are respected, evaluating the power-flow feasibility of an island $\mathcal{I}$ corresponds to finding a solution to the mixed-integer linear program (6) – (12) with fixed $\boldsymbol{u}$ and restricted to $\mathcal{I}$. There are three possible outcomes to this process: (*i*) there exists a power flow solution, (*ii*) there exists no solution to the linear relaxation or (*iii*) there exists a solution to the linear relaxation but no solution for the mixed-integer problem. Island-based cuts remove islands falling into the second and third cases from the *Master* problem. In order to present these cuts in a compact fashion, first, we define

$$\Delta(\mathcal{I}, t) := 1_{|L(\mathcal{I})|=0} \cdot \sum_{n \in N(\mathcal{I})} u_{n,t} + \sum_{\substack{n \in L(\mathcal{I}) \cup L(\partial N(\mathcal{I})): \\ \bar{u}_l^{\mathcal{I}}=0}} u_{l,t} + \sum_{\substack{n \in L(\mathcal{I}): \\ \bar{u}_l^{\mathcal{I}}=1}} (1 - u_{l,t}) + \sum_{\substack{sc \in SC(\mathcal{I}): \\ \bar{u}_{sc}^{\mathcal{I}}=0}} u_{sc,t} + \sum_{\substack{sc \in SC(\mathcal{I}): \\ \bar{u}_{sc}^{\mathcal{I}}=1}} (1 - u_{sc,t}),$$

which indicates whether the topology of island $\mathcal{I}$ is present at snapshot t. Here $\partial N(\mathcal{I})$ refers to the boundary of the island.

Let island $\mathcal{I}$ be infeasible with respect to the linear relaxation of the power flow equations. Then, we can obtain an unbounded ray from the dual problem μ^*, $\boldsymbol{\lambda}^*$ with respect to energization decisions of generators and use the following hybrid Benders-binary cut

$$\mu^* + \sum_{g \in G(\mathcal{I})} (\lambda_g^* u_{g,t} + \lambda_g^{CR,*} u_{g,t}^{CR}) \leq \nu \cdot \Delta(\mathcal{I}, t), \tag{15}$$

where $\nu := \mu^* + \sum_{g \in G(\mathcal{I})} \max\{0, \lambda_g^*, \lambda_g^* + \lambda_g^{CR,*}\}$, to remove $\mathcal{I}$ from the feasible set of the *Master* problem. By using the information of the dual unbounded ray solely for generators, the proposed hybrid cut avoids the influence of arbitrarily large constants (used in the McCormick envelopes of branch equations) on the infeasibility cut. One hybrid cut can render multiple islands infeasible, indicating, for instance, that a certain island topology is infeasible irrespective of the status of the generators connected to the island.

Note that, since our piecewise linear approximation of the power flow equations is locally ideal [22], [26], the linear relaxation (6) – (12) for a given island is very similar to the linear programming approximation proposed in [4], sharing its inability to capture overvoltages.

3) Binary island-based cuts: If an island $\mathcal{I}$ is feasible for the linear relaxation but infeasible for the mixed-integer power flow equations, then we can exclude $\mathcal{I}$ from the feasible set of the *Master* problem using the binary cut

$$\begin{aligned} &\Delta(\mathcal{I}, t) + \sum_{\substack{g \in G(\mathcal{I}): \\ \bar{u}_g^{\mathcal{I}} = 0}} u_{g,t} + \sum_{\substack{g \in G(\mathcal{I}): \\ \bar{u}_g^{\mathcal{I}} = 1}} (1 - u_{g,t}) \; + \\ &\sum_{\substack{g \in G(\mathcal{I}): \\ \bar{u}_g^{\mathcal{I}} = 1, \bar{u}_g^{\mathcal{I},CR} = 0}} u_{g,t}^{CR} + \sum_{\substack{g \in G(\mathcal{I}): \\ \bar{u}_g^{\mathcal{I},CR} = 1}} (1 - u_{g,t}^{CR}) \geq 1, \end{aligned} \tag{16}$$

which excludes only island $\mathcal{I}$, since we cannot directly infer the feasibility of similar islands from the evaluation of $\mathcal{I}$.

B. Heuristics

Both our initialization and rounding heuristics rely on a sequential greedy logic: define the best configuration for snapshot t and fix it, then define the best configuration for $t + 1$. These heuristics differ, however, on their goals and the information available to them, as explained in the following.

1) Initialization heuristic: The initialization heuristic, presented in Fig. 3, seeks to find a fully feasible restoration plan $\bar{\boldsymbol{u}}$ to be used as incumbent by the *Branch-and-Bound* algorithm, Fig 1. In essence, the heuristic greedily (based on objective function value) selects one-by-one which elements to energize for each snapshot and then moves to the next snapshot.

2) Rounding heuristic: Let $\tilde{\boldsymbol{u}}$ be a fractional candidate. The rounding heuristicseeks to round $\tilde{\boldsymbol{u}}$ to a binary candidate $\bar{\boldsymbol{u}}$ that respects the energization constraints and that does not include any known infeasible island. The heuristic solves a sequence of L_1 projection problems (17) for each $t \in T$, fixing $\boldsymbol{u}_\tau = \bar{\boldsymbol{u}}_\tau$

1: Let $\bar{\boldsymbol{u}}_t = 0$ for $t = 0$
2: **for** $t \in T$ **do**
3: Let $\bar{\boldsymbol{u}}_t := \bar{\boldsymbol{u}}_{t-1}$
4: Let $\bar{u}_{g,t}^{CR} := 0$ and $\bar{u}_{n(g),t} := 1$ for every cranked $g \in G$
5: Detect islands $\mathscr{I}(\bar{\boldsymbol{u}}_t)$
6: Attempt to energize branches in the boundary of all $\mathcal{I} \in \mathscr{I}(\bar{\boldsymbol{u}}_t)$, following merit order (decreasing in β), until K branches have been energized
7: Start cranking additional generators
8: **end for**

Fig. 3. *Initialization heuristic*. Existence of an online generator within each island is ensured by construction. Islands are checked for power flow feasibility after each status change in lines 3, 5–6. As a result $\bar{\boldsymbol{u}}$ corresponds to a fully feasible restoration plan.

for all $\tau \in T : \tau < t$ and ignoring constraints for future snapshots.

$$\begin{aligned} \min_{u \in \{0,1\}} \quad & \|\boldsymbol{u}_t - \tilde{\boldsymbol{u}}_t\|_1 \\ \text{s.t.} \quad & u_{a,t} = 1 \quad \forall a \in N \cup L : \bar{u}_{a,t-1} = 1 \\ & (1) – (5) \text{ up to } t \\ & (15)\ \forall \mathcal{I} \in \mathscr{I}^{LP},\ (16)\ \forall \mathcal{I} \in \mathscr{I}^{MILP} \end{aligned} \tag{17}$$

The projection problem prevents the de-energization of transmission elements and bans infeasible islands kept in the hash table of the *Feasibility Oracle*, classified between islands that are infeasible with respect to the relaxed power flow equations, $\mathscr{I}^{LP}$, and islands feasible for the relaxation but infeasible for the mixed integer power flow equations, $\mathscr{I}^{MILP}$.

V. NUMERICAL EXPERIMENTS

We implement our decomposition approach in Julia 0.6.4 [27], using JuMP 0.18.2 [28] to formulate mathematical programming models and Gurobi for solving them. We parallelize the feasibility evaluation of islands using the built-in multi-threading capabilities of Julia. All our experiments are conducted using a single node of the Quartz cluster (2×Intel Xeon E5-2695, 36 cores, 128GB RAM per node) hosted at the Lawrence Livermore National Laboratory.

We present OPSR results for the IEEE39 (5-minute resolution, 40 snapshots), IEEE118 (20-minute resolution, 30 snapshots) and Chilean (1-hour resolution, 40 snapshots) networks. For each network, we solve OPSR using three – increasingly detailed – power flow models for determining the feasibility of electrical islands:

1) The DC power flow approximation (DC), implemented by replacing lines 13–24 of the *Feasibility Oracle* (Fig. 2) with a feasibility check using the DC power flow.
2) The linear relaxation of the piece-wise linear approximation of the AC power flow (LRAC), implemented by removing lines 20–24 from the *Feasibility Oracle*.
3) The mixed-integer piece-wise linear approximation of the AC power flow (MIAC), as described in section IV.

We set a target optimality gap of 1% and a solution time limit of 24 hours for all instances, except for the Chile-MIAC instance, for which it is set at 48 hours.

TABLE I
OPSR SOLUTION QUALITY AND SOLUTION TIMES

System	Power flow	Value [-]	Gap [%]	Solution time [s]	Time composition [%] B&B	Cuts	Heur.
IEEE39	DC	26.41	0.01	13.0	14.6	0.8	84.6
	LRAC	26.41	0.01	17.5	10.3	1.1	88.6
	MIAC	26.41	0.01	22.2	8.1	1.8	90.1
IEEE118	DC	18.73	0.03	22.6	13.3	2.7	84.1
	LRAC	18.73	0.03	49.6	5.8	2.6	91.5
	MIAC	18.58	0.84	19901.1	0.1	99.0	0.9
Chile	DC	32.52	0.77	33123.7	90.2	4.6	5.2
	LRAC	30.65	6.88	84044.5	69.7	23.9	6.4
	MIAC	30.64	7.26	172188.3	5.4	14.3	80.2

TABLE II
TOTAL OVERVOLTAGE AND POWER BALANCE VIOLATIONS

System	Power flow	Total violations [pu] Over-voltage	Active balance	Reactive balance
IEEE39	DC	0.00	0.00	0.00
	LRAC	0.00	0.00	0.00
	MIAC	0.00	0.00	0.00
IEEE118	DC	0.23	1.16	1.91
	LRAC	0.23	1.09	1.99
	MIAC	0.00	0.18	1.48
Chile	DC	0.00	0.00	4.09
	LRAC	0.00	0.00	6.33
	MIAC	0.00	0.00	1.49

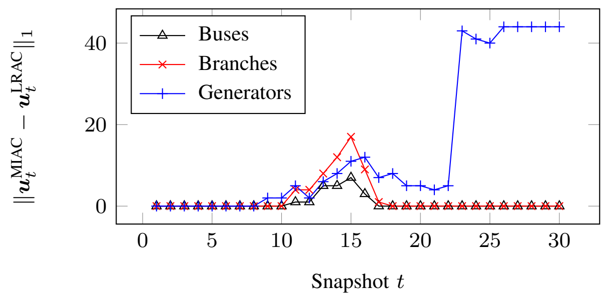

Fig. 4. Difference in energization decisions due to different power flow models (LRAC and MIAC) for the IEEE118 system.

A. Computational performance of our approach

Table I summarizes the computational performance of our specialized L-shaped algorithm. The method solves instances from the literature, IEEE39-*, IEEE118-DC and IEEE118-LRAC, in under 1 minute. For a comparison, in [12] problems with the equivalent complexity of the IEEE39-LRAC and IEEE118-LRAC problems were solved in 30 minutes and 3 hours, respectively, using 6 nodes of a high-performance computing cluster. The method also solves OPSR for the Chile-DC instance, which is larger than any OPSR instance solved in the literature, within the desired optimality gap, whereas the Chile-LRAC instance ran out of time finding only the solution provided by the initialization procedure.

Solution times for the IEEE118-MIAC and Chile-MIAC instances show how difficult is to solve OPSR using a non-convex power flow model. In the case of IEEE118-MIAC, which achieves the desired optimality gap, most of the time is spent evaluating islands that are LRAC feasible but MIAC infeasible, whereas for Chile-MIAC, most of the time was spent in the initialization procedure, within which islands that are LRAC feasible but MIAC infeasible were also the major challenge. These observations suggest the need for a more computationally efficient way to evaluate power flow feasibility under excess reactive power conditions [22].

B. Influence of power flow models on feasibility of optimal restoration plans

After the optimal restoration plan is decided on each case, we compute an actual non-linear AC power flow (NLAC) solution for each island by minimizing the L_2 distance in terms of production and consumption schedules to the approximate solution, while penalizing overvoltage and power balance violations. We solve these L_2 minimization problems using Ipopt 3.12.9 [29] compiled with HSL [30].

Table II presents the locally-minimum violations found. For the IEEE39-* instances, all power flow models lead to the same energization solution, which is NLAC feasible. In the case of IEEE118, the DC and LRAC model lead to the same energization solution. However, due to different starting points for the NLAC recovery, violations differ slightly among DC and LRAC, without any outperforming the other. The IEEE118-MIAC energization solution differs from the IEEE118-LRAC solution, as shown in Fig. 4, in terms of topology (buses and branches) for $11 \leq t \leq 16$ and in terms of generators status for $t \geq 8$. Differences for $8 \leq t \leq 22$ are triggered by overvoltages due to the Ferranti rise, which are overlooked by the DC and LRAC models. Differences observed for $t \geq 23$, on the other hand, correspond mostly to the energization of synchronous condensers in the solution to IEEE118-MIAC, which help reducing power balance violations up to the accuracy of the MIAC approximation.

The Chile-* instances present a similar behaviour as the IEEE118-* instances in respect to violations, with the MIAC model leading to smaller reactive balance violations than DC and LRAC. Surprisingly, LRAC performs worse than DC, a result that stresses the potentially harmful impacts of using power flow relaxations in extreme operating conditions.

VI. CONCLUSION

We proposed a specialized decomposition approach for OPSR, based on the integer L-shaped method, enhanced with stronger binary and hybrid cuts, initialization heuristics and rounding heuristics. Our algortihm solves instances from the literature orders of magnitude faster than the state-of-the-art and solves, for the first time, instances of realistic systems (both in terms of size and complexity of its components).

We find that power flow relaxations of the type of LRAC provide no significant advantage over the DC power flow approximation for deciding energization sequences. The piece-wise linear approximation MIAC, on the other hand, provides more accurate solutions, at the cost of significantly larger (sometimes, orders of magnitude larger) solution times.

Future research will focus (*i*) on the development of power flow relaxations that tend to be tight in extreme operating conditions, particularly, excess reactive power, and (*ii*) the development of models and algorithms for OPSR that take into account uncertainty regarding damaged components and the unobservability of the damage state of certain components, such as long lines, prior to an energization attempt.

Appendix A
Nomenclature

Sets

T	time snapshots $\{\underline{T}, \ldots, \overline{T}\}$
N	buses
L, SC	lines and series compensators (branches)
G	generators
G^{BS}	blackstart generators, $G^{BS} \subseteq G$
SH	shunt reactors
$\mathscr{I}$	islands
$I(a)$	terminals of branch a
$G(a), L(a), SC(a), SH(a)$	generators, lines, series compensators, shunts in a, where $a \in N$, $a \subseteq N$, or $a \in \mathscr{I}$
$S_a, S_g(\cdot), S_c(\cdot)$	active and reactive power feasible domain for branch a, generator g, or consumer c

Parameters

$n(a)$	bus of terminal a or generator a
CT_g	cranking time of generator g
K	maximum number of transmission element energizations in between snapshots
$\underline{V}, \overline{V}$	lower and upper voltage bounds
$P_a(\cdot)$, $Q_a(\cdot)$	active and reactive power flow equations on branch a
J_g	inertia constant of generator g
$\alpha, \Xi(\cdot)$	objective coefficient and benefit function for system inertia
β_l, β_n	benefit from energizing line l, bus n

Variables

$u_{a,t}$	energization indicator of component a at $t \in T$
$u^{CR}_{g,t}$	cranking indicator of generator g, $t \in T$
$v_{n,t}, \theta_{n,t}$	voltage magnitude and angle at bus n, $t \in T$
$p_{a,t}, q_{a,t}$	power exiting generator a, entering shunt a or entering consumer a, $t \in T$
$p_{a,i,t}, q_{a,i,t}$	power entering branch a by terminal i, $t \in T$

Acknowledgment

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. The authors would like to thank Gurobi for providing licenses of the Gurobi Optimizer.

References

[1] National Academies of Sciences, Engineering, and Medicine, *Enhancing the Resilience of the Nation's Electricity System*. Washington, DC: The National Academies Press, 2017.

[2] PJM Interconnection, "PJM's evolving resource mix and system reliability," 2017.

[3] Federal Energy Regulatory Commission, "Docket Nos. RM18-1-000 (Grid Reliability and Resilience Pricing), AD18-7-000 (Grid Resilience in Regional Transmission Organizations and Independent System Operators)," 2018.

[4] P. Van Hentenryck and C. Coffrin, "Transmission system repair and restoration," *Mathematical Programming*, 2015.

[5] A. L. Morelato and A. J. Monticelli, "Heuristic search approach to distribution system restoration," *IEEE Transactions on Power Delivery*, vol. 4, no. 4, pp. 2235–2241, 1989.

[6] D. N. Abu Talib, H. Mokhlis, M. S. Abu Talip, K. Naidu, and H. Suyono, "Power system restoration planning strategy based on optimal energizing time of sectionalizing islands," *Energies*, vol. 11, no. 5, 2018.

[7] Y. Liu, R. Fan, and V. Terzija, "Power system restoration: a literature review from 2006 to 2016," *Journal of Modern Power Systems and Clean Energy*, 2016.

[8] Y. Chou, C. Liu, Y. Wang, C. Wu, and C. Lin, "Development of a black start decision supporting system for isolated power systems," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 2202–2210, 2013.

[9] W. Sun, C. Liu, and L. Zhang, "Optimal generator start-up strategy for bulk power system restoration," *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1357–1366, 2011.

[10] C. Coffrin and P. Van Hentenryck, "A linear-programming approximation of ac power flows," *INFORMS Journal on Computing*, vol. 26, no. 4, pp. 718–734, 2014.

[11] A. Arab, A. Khodaei, S. K. Khator, and Z. Han, "Transmission network restoration considering ac power flow constraints," in *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 816–821, 2015.

[12] G. Patsakis, D. Rajan, I. Aravena, J. Rios, and S. Oren, "Optimal black start allocation for power system restoration," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6766–6776, 2018.

[13] A. Arab, A. Khodaei, S. K. Khator, and Z. Han, "Electric power grid restoration considering disaster economics," *IEEE Access*, vol. 4, pp. 639–649, 2016.

[14] A. Golshani, W. Sun, Q. Zhou, Q. P. Zheng, and Y. Hou, "Incorporating wind energy in power system restoration planning," *IEEE Transactions on Smart Grid*, pp. 1–1, 2018.

[15] G. Laporte and F. V. Louveaux, "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, no. 3, pp. 133 – 142, 1993.

[16] G. Angulo, S. Ahmed, and S. S. Dey, "Improving the integer L-shaped method," *INFORMS Journal on Computing*, vol. 28, no. 3, pp. 483–499, 2016.

[17] North American Electric Reliability Corporation, "Standard EOP-005-2 – System Restoration from Blackstart Resources," 2013.

[18] Coordinador Eléctrico Nacional, "Estudio para Plan de Recuperación de Servicio (Restoration Plan Study)," 2017.

[19] Australian Energy Market Operator, "Black System South Australia 28 September 2016 – Final Report," 2017.

[20] PJM Interconnection, "PJM Manual 36: System Restoration," 2018.

[21] G. Patsakis, D. Rajan, I. Aravena, and S. Oren, "Strong mixed-integer formulations for power system islanding and restoration," 2018.

[22] I. Aravena, D. Rajan, and G. Patsakis, "Mixed-integer linear approximations of ac power flow equations for systems under abnormal operating conditions," in *2018 IEEE PES General Meeting*, pp. 1–5, 2018.

[23] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part i — convex underestimating problems," *Mathematical Programming*, 1976.

[24] K. Fukuda, "cddlib." URL: https://github.com/cddlib/cddlib. Accessed: 2018-12-16.

[25] B. Legat, "Polyhedra." URL: https://github.com/JuliaPolyhedra/Polyhedra.jl. Accessed: 2018-12-16.

[26] J. Vielma, "Mixed integer linear programming formulation techniques," *SIAM Review*, vol. 57, no. 1, pp. 3–57, 2015.

[27] J. Bezanson, A. Edelman, S. Karpinski, and V. Shah, "Julia: A fresh approach to numerical computing," *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.

[28] I. Dunning, J. Huchette, and M. Lubin, "JuMP: A modeling language for mathematical optimization," *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.

[29] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[30] HSL, "A collection of fortran codes for large scale scientific computation," 2015.