2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Optimal Control of Cascading Power Grid Failures

Daniel Bienstock

***Abstract*— We study algorithms for computing optimal load shedding schedules in the event of a cascading power system failure. The algorithms compute an affine control at the onset of the cascade; the control is applied during the cascade as a function of observed state parameters such as line overloads. In the case of line outages that follow a deterministic rule, we obtain an efficient (polynomial time) algorithm for computing an optimal control. For the case of stochastic line outages we describe a stochastic gradients algorithm. We present computational experiments with a parallel implementation of our algorithms, tested on a snapshot of the U.S. Eastern Interconnect.**

I. INTRODUCTION

A. Underlying approach

In a cascading failure of a power transmission system, an initial event that disables a possibly small subset of the grid conspires with the laws of physics to set off a sequence of additional outages that, in the worst case, accelerates until a large subset of the network is inoperative, resulting in a significant loss of served power.

The mechanics of the process can be summarized as follows: each time a component of the system becomes outaged, a new set of power flows takes hold in the remaining network, following the laws of physics and automatic control actions. Should the new flows, for example, exceed the rating of a given line, then that line will likely become outaged in the near future. In an adverse scenario this gives rise to a vicious cycle which constitutes the cascade.

The complexity of the process is due to interaction between the physics of power flows, as described by an appropriate system of equations, and discrete (i.e., discontinuous) changes in the topology of the network as components are removed from operation. This interaction gives rise to non-monotonic behavior, both from the perspective of control actions (more of a good thing is not necessarily better) and with respect to the severity of the initial event. Non-monotonicity, from a mathematical perspective, is equivalent to non-convexity, and thus, complexity. Additionally, there is poorly understood *noise* inherent in the operation of the system; for example a thermally stressed line is more likely to 'trip' due to many factors that are difficult to model in a precise manner.

We consider algorithms that shed load (demand) as a function of observations taken in real time, with the goal of arresting the cascade with a minimum of demand lost. In principle, one could think of algorithms in this mold that are available *prior to* the cascade; i.e. algorithms independent of the nature of the cascade. However, there is an exponential amount of variability in the structure of the initial event that sets-off the cascade, and in our simulations it is clear that the nature of the cascade can depend strongly on the initial event. Instead, we focus on control algorithms that are computed soon after the onset of the cascade, and which therefore benefit from the knowledge available at that point. We assume an initially *slow-moving* cascade so that at the start of the process there is sufficient time (e.g., minutes) to compute an appropriate control algorithm; once computed, the control will be applied as the cascade unfolds.

In devising a load-shedding schedule to respond to a potential cascade, one must decide when, where and by how much demand is to be shed. Our method can be viewed as a data-driven approach for computing such actions –it is data-driven because it relies on the knowledge of the initial event. At the same time, the efficiency of the algorithms makes it potentially feasible to investigate common patterns of desirable control actions that could be directly applied in the event of a contingency.

B. Models for power flows, cascades and controls.

The behavior of power grids is commonly modeled using systems of nonlinear, nonconvex equations that describe the physics of AC power flows [3]. Under normal operating conditions, a reasonably close solution to the system can be produced; then Newton-Raphson methods will converge to a useful solution in few iterations.

Under extreme operating conditions, Newton-Raphson may fail to converge. Additionally, even though Newton-Raphson is relatively fast, it may be too slow when a large volume of power flow computations is required. For these and other reasons one often relies on the *linearized* or DC, power flow approximation, which is solved far more quickly and proves accurate under good operating conditions.

In the linearized approximation we are given a directed graph G with n nodes and m arcs (corresponding, respectively, to buses and lines). In addition

- For each arc j we are given two positive quantities: its *flow limit* u_j and its *reactance* x_j.
- We are given a *supply-demand* vector $\beta \in \mathcal{R}^n$ with the following interpretation. For a node i, if $\beta_i > 0$ then i is a *generator* (a source node) while if $\beta_i < 0$ then i is a *load* (a demand node) and in that case $-\beta_i$ is the *demand* at i. The condition $\sum_i \beta_i = 0$ is assumed to hold. For a generator node i, we indicate by the constant $\tilde{s}_i$ the maximum supply of i. We denote by $\mathcal{G}$ denote the set of generators and by $\mathcal{D}$ the set of demand nodes.

D. Bienstock is with the Departments of Industrial Engineering and Operations Research, and Applied Physics and Applied Mathematics, Columbia University, New York, NY, 10027 USA. email: dano@columbia.edu

The linearized power flow problem specifies a variable f_{ij} associated with each arc j (active power flow) and a variable ϕ_i associated with each bus i (phase angle). The problem consists in finding a solution to the system of equations:

$$Nf \;=\; \beta, \quad N^T\phi - Xf \;=\; 0, \qquad (1)$$

where N denotes the node-arc incidence matrix of G [1] and $X = \mathbf{diag}\{x_{ij}\}$.

Remark 1.1: It can easily be shown that system (1) is feasible if and only if $\sum_{i \in K} \beta_i = 0$ for each component ("island") K of G, and in that case the solution is unique in the f variables.

Template 1.2, based on [5], [6], [7], is our cascade model. It focuses on line faults, however the template and our algorithms below are easily modified to accommodate wider types of faults.

```
Template 1.2: GENERIC CASCADE TEMPLATE
Input: a power grid with graph $G$ (post-initiating event). Set $G^1 = G$.
For $r = 1, 2, \ldots$ Do
    (comment: round $r$ of the cascade)
    1. Set $f^r$ = vector of power flows in $G^r$.
    2. Set $\mathcal{O}^r$ = set of lines of $G^r$ that become outaged
       in round $r$.
    3. Set $G^{r+1} = G^r - \mathcal{O}^r$. Adjust loads and
       generation in $G^r$.
```

The adjustment in Step 3 handles the case of *islanding*, where the line outages create isolated components of the network. A newly created island might have an excess of generation over demand – in such a case we assume that the excess is removed by reducing the output of all generators in that island in equal amounts. The case of excess demand is handled similarly.

Step 2 requires a specific line outage model. For each line j, let $0 \le \alpha_j \le 1$ be a fixed parameter, and for each line j and each round r define $\tilde{f}_j^r$ by

$$\tilde{f}_j^r \;=\; \alpha_j |f_j^r| \,+\, (1 - \alpha_j)\tilde{f}_j^{r-1}, \qquad (2)$$

with $\tilde{f}_j^0$ set to the absolute value of the flow on j prior to the incident that starts the cascade. The $\tilde{f}_j^r$ are moving averages of the (absolute value) of the flow on line j; parameter α_j serves to encode "memory" (for example, so as to model thermal effects). When $\alpha_j = 1$ for all j the system is *memoryless*. We use the deterministic outage rule

$$\text{line } j \text{ becomes outaged if } \tilde{f}_j^r > u_j. \qquad (3)$$

This rule can lead to non-smooth behavior; later we will describe a (smoother) randomized rule. To incorporate control we modify Template 1.2 as follows:

II. Adaptive Control

```
Template 2.1: CASCADE CONTROL
Input: a power grid with graph $G$, and integers
$1 \le r^{min} < R$.    Set $G^1 = G$.

Step 0. Compute control algorithm.

For $r = 1, 2, \ldots, R-1$, Do
    (comment: controlled round $r$ of the cascade)
    1. Set $f^r$ = vector of power flows in $G^r$.
    2. Obtain grid measurements.
    3. If $r \ge r^{min}$, apply control.
    4. Set $g^r$ = vector of resulting power flows in $G^r$.
    5. Set $\mathcal{O}^r$ = set of lines of $G^r$ that become
       outaged in round $r$.
    6. Set $G^{r+1} = G^r - \mathcal{O}^r$. Adjust loads and
       generation in $G^r$.

Termination (round $R$). If any island of $G^R$ has line overloads,
proportionally shed demand in that island until all line overloads
are eliminated. Thus, for any island $K$ of $G^R$, set
$\Psi_K^R \doteq \min\{1,\, \max_{j \in K}\{|f_j^R|/u_j\}\}$. If $\Psi_K^R > 1$,
then any bus $v$ of $K$ resets its demand to $d_v^R/\Psi_K^R$.
```

The termination condition is specified to ensure that the cascade is stopped at the end of the planning horizon. With $r^{min} > 1$ we model delays in implementation of the control.

Next we describe the general form of our control. We use an affine law, described by a triple of values (c_v^r, b_v^r, s_v^r) (computed in Step 0 of Template 2.1) for each round r and load bus v. At round r, let d_v^r denote the current demand at v, and let κ_v^r be the maximum line overload in the island currently containing v (these are the quantities observed in Step 2). Then, in Step 3, bus v resets its demand to:

$$\begin{aligned} &\min\{1,\, [b_v^r + s_v^r(c_v^r - \kappa_v^r)]^+\}\; d_v^r, && \text{if } \kappa_v^r > c_v^r \qquad (4)\\ &d_v^r, && \text{otherwise.} \qquad (5)\end{aligned}$$

In equation (4), $[x]^+$ denotes $\max\{x, 0\}$. To understand this rule, consider the case where, for all v, $b_v^r = c_v^r = 1$ and $s_v^r > 0$. Then if $\kappa_v^r > 1$ we shed demand at v by a multiplicative factor proportional to the excess overload $\kappa_v^r - 1$, while if $\kappa_v^r \le 1$ we do not shed demand at v.

In this paper we consider the computation of a control of type (4)-(5) to be applied as in Template 2.1, with the objective of maximizing the demand being served at termination of the cascade. We assume an initially slow moving cascade so that there is enough time to perform the computation in Step 0 (which, as we discuss below, requires minutes of wall-clock time in an appropriate implementation). We also assume that the κ_v^r can be observed in Step 2. In our discussions and implementations, we will also assume that the number of rounds, R, is relatively small. This is done so as to produce controls with simple structure which do not require frequent application.

By construction, the control parameters depend on the cascade round and on the location within the grid; our approach can therefore be viewed as a distributed control. Nevertheless, as we will see below, in important special cases the optimal control has global "structure".

See [2] for a different approach to control.

III. The optimal scaling problem

Consider the special case of our control algorithm where

(i) We use *island-wise control*. That is to say, for each round r for each island K of G^r, and for each bus $v \in K$, (c^r_v, b^r_v, s^r_v) equals a fixed triple (c^r_K, b^r_K, s^r_K).

(ii) We use the memoryless version of line outage rule (3), i.e. line j becomes outaged in round r if $f^r_j > u_j$.

Rule (i) is of interest because of the resulting simplicity of the control. In fact, without loss of generality we may assume that for each round r there is a constant s^r with $s^r_J = s^r$ for every island J of G^r. This is relevant because the "s" parameter in our controls are clearly of key importance since they define the system response to increasingly large overloads; the sequence $s^1, s^2, \ldots, s^{R-1}$ reflects the timing of significant control. At the same time, the load shedding formula (4) specifies different scaling *multipliers* for different islands, according to the magnitudes of their respective maximum line overloads (and thus, the relative severity of the cascade in each different island).

We call this special case of the control computation the *optimal scaling problem*. Note that if $1 \le r < R$ and K is a component of G^r under an island-wise control, then (at round r) we will scale all demands in K by the common multiplier $0 \le \lambda^r_K \le 1$ defined by

$$\lambda^r_K \doteq \min\{1,\, [b^r_K + s^r_K(c^r_K - \kappa^r_K)]^+\}. \tag{6}$$

Conversely, given a value $0 \le \lambda^r_K \le 1$ and κ^r_K it is simple to find a triple (c^r_K, b^r_K, s^r_K) satisfying (6). Thus the optimal control problem is equivalently stated in terms of the parameters λ^r_K. At this point a remark is necessary.

Remark 3.1: Even though in principle the number of islands at any round can be exponentially large, when defining a particular control only a $O(Rn)$ islands need be considered. This follows because at the start of round 1 the islands are given; and the control parameters λ^1_K need only be stated for those islands. After applying the control the resulting set of islands will be unique (because of our deterministic outage rule). Thus the set of islands in round 2 will be unique. The result follows by induction.

In what follows we prove:

Theorem 3.2: The optimal scaling problem can be solved in time $O\left(\frac{m^R}{(R-1)!}\right)$.

Remark 3.3: As before, m indicates the number of lines. As argued above we would not use a large value for R; furthermore in practical testing of the algorithm the worst-case complexity bound in Theorem 3.2 is not attained and the algorithm proves very efficient even on large-scale grids. We suspect that a tighter bound may be possible.

In preparation for the proof of this result, we make some observations.

Notation 3.4: Let G be a graph, and let μ be a supply-demand vector on G. We denote by $\hat{f}(G, \mu)$ the unique, feasible flow vector on G when μ is the supply-demand vector (see Remark 1.1).

In what follows we assume that we have a given supply-demand vector β. Let R be the number of rounds for the cascade. Our problem is to compute a control that maximizes the total demand satisfied after R rounds, assuming that at the start of round 1, β is the supply-demand vector. We will solve this as a special case of a family of problems.

Definition 3.5: For $t \ge 0$ real, denote by $\Theta^{(R)}_G(t|\beta)$ the final total demand resulting from applying an optimal control in an R-round cascade on graph G, where the initial supply-demand vector is $t\beta$. We refer to this parameter as the *yield*.

Remark 3.6: Let α be a supply-demand vector on graph G. Let $t \ge 0$. Then (refer to notation 3.4) $\hat{f}(G, t\alpha) = t\hat{f}(G, \alpha)$.

Lemma 3.7: Let μ be a supply-demand vector. Suppose G is connected. Then $\Theta^{(1)}_G(t|\mu)$ is a nondecreasing piecewise-linear function of t with two pieces, the second one of which has zero slope.

Proof. Note that since $R = 1$, only the termination step in algorithm (2.1) will be executed. Further, writing $\hat{f} = \hat{f}(G^1, \alpha)$, when running (2.1) starting with the initial supply-demand vector $t\mu$, we will have $f^1 = t\hat{f}$ in Step 1, and writing $\psi = \max_j |\hat{f}_j|/u_j$, we have that $\max_j |f^1_j|/u_j = t\psi$. Denoting by $\tilde{D}$ the sum of demands implied by μ we have as per our cascade termination criterion that the final total demand at the end of $R = 1$ rounds will equal

$$t\tilde{D}, \qquad \text{if } \ t \le 1/\psi, \quad \text{and} \tag{7}$$

$$\frac{t}{t\psi}\,\tilde{D} = \frac{1}{\psi}\,\tilde{D}, \qquad \text{otherwise.} \quad \blacksquare \tag{8}$$

Now we turn to the general case with $R > 1$. We assume, without loss of generality, that G^1 is connected. Let $\hat{f} = \hat{f}(G^1, \beta)$.

Definition 3.8: A *critical point* is a real $\gamma > 0$, such that for some line j, $\gamma\hat{f}_j = u_j$.

Recall that we assume $u_j > 0$ for all j; thus let $0 < \gamma_1 < \gamma_2 < \ldots < \gamma_p$ be the set of all distinct critical points. Here $0 \le p \le m$. Write $\gamma_0 = 0$ and $\gamma_{p+1} = +\infty$.

Definition 3.9: For $1 \le i \le p$ let
$$F^i = \{j \in \mathcal{A} : \gamma_i|\hat{f}_j| = u_j\}.$$

Now assume that the initial supply-demand vector is $t\beta$ with $t > 0$ and let $0 < \lambda^1 \le 1$ be the optimal multiplier used to scale demands in round 1. Write

$$q = q(t) = \text{argmax}\{h : \gamma_h < t\}. \tag{9}$$

Thus, $t \le \gamma_{q+1}$, and so $\lambda^1 t \le \gamma_{q+1}$. We stress that these relationships remain valid in the boundary cases $q = 0$ and $q = p$.

Notation 3.10: Let index i be such that $\lambda^1 t \in (\gamma_{i-1}, \gamma_i]$.

Note that in Step 3 of round 1 of algorithm (2.1) we will scale all demands by λ^1, and since we assume G^1 is connected, in Step 4 we will also scale all supplies by λ^1. Thus, for any $h \le i-1$, and any line $j \in F^h$, we have that after Step 4 the absolute value of the flow on j is $\lambda^1 t\, |\hat{f}_j| > \gamma_h\, |\hat{f}_j| = u_j$, and consequently j becomes outaged in round 1. On the other hand, for any line $j \notin \cup_{h \le i-1} F^h$, the absolute value of the flow on j immediately after Step 4 is $\lambda^1\, t\, |\hat{f}_j| \le \gamma_i\, |\hat{f}_j| \le u_j$, and so j does not become outaged in round 1. In summary, the set of outaged lines is $\cup_{h \le i-1} F^h$; in other words, we obtain the same network $G^2 = G^1 - \cup_{h=1}^{i-1} F^h$ for every t with $\lambda^1 t \in (\gamma_{i-1}, \gamma_i]$.

Notation 3.11: For an index j, write $\mathcal{K}(j)$ = set of components of $G^1 - \cup_{h=1}^{j} F^h$.

Let $H \in \mathcal{K}(i-1)$. Then, prior to Step 6 of round 1, the supply-demand vector for H is precisely the restriction of $\lambda^1 t\beta$ to the buses of H, and when we adjust supplies and demands in Step 6, we will proceed as follows

- if $\sum_{s \in \mathcal{D} \cap H} (-\lambda^1 t \beta_s) > \sum_{s \in \mathcal{G} \cap H} (\lambda^1 t \beta_s)$ then for each demand bus $s \in \mathcal{D} \cap H$ we will reset its demand to $-r\lambda^1 t\beta_s$, where

$$r = \frac{\sum_{s \in \mathcal{G} \cap H} (\lambda^1 t \beta_s)}{\sum_{s \in \mathcal{D} \cap H} (-\lambda^1 t \beta_s)} = -\frac{\sum_{s \in \mathcal{G} \cap H} (\beta_s)}{\sum_{s \in \mathcal{D} \cap H} (\beta_s)},$$

  and we will leave all supplies in H unchanged.
- likewise, if $\sum_{s \in \mathcal{D} \cap H} (-t\lambda^1 \beta_s) < \sum_{s \in \mathcal{G} \cap H} (\lambda^1 t \beta_s)$ then the supply at each bus $s \in \mathcal{G} \cap H$ will be reset to $r\lambda^1 t \beta_s$, where

$$r = -\frac{\sum_{s \in \mathcal{D} \cap H} (\beta_s)}{\sum_{s \in \mathcal{G} \cap H} (\beta_s)},$$

  but we will leave all demands in H unchanged.

Note that in either case, in round 2 component H will have a supply-demand vector of the form $\lambda^1 t \beta^H$, where β^H is a supply-demand vector which is *independent of* t. Thus an optimal control on H, on rounds $2, \ldots, R$, will yield a final total demand

$$\Theta_H^{(R-1)}(\lambda_1 t | \hat{\beta}^H), \tag{10}$$

which, inductively, is a nondecreasing function of $\lambda_1 t$, and therefore is largest when $\lambda^1 = \min\left\{1, \frac{\gamma_i}{t}\right\}$.

Case 1. Suppose $i \le q$. As noted above, by definition (9) of q we have that $\gamma_i \le \gamma_q < t$. Thus, the expression in (10) is maximized when $\lambda_1 = \frac{\gamma_i}{t}$, and we obtain final ($R$-round) demand equal to

$$T_i \doteq \sum_{H \in \mathcal{K}(i-1)} \Theta_H^{(R-1)}(\gamma_i | \hat{\beta}^H), \tag{11}$$

which is independent of t.

Case 2. Suppose instead that $q < i$, and so $i = q+1$ by definition of q and $\lambda^1 \le 1$. Thus (10) is maximized by setting $\lambda^1 = 1$. Writing

$$S_H(t) \doteq \Theta_H^{(R-1)}(t | \hat{\beta}^H),$$

The final demand equals $\sum_{H \in \mathcal{K}(q)} S_H(t)$.

In summary,

$$\Theta_G^{(R)}(t|\hat{\beta}) = \max\left\{ \sum_{H \in \mathcal{K}(q)} S_H(t)\ ,\ \max_{1 \le i \le q} T_i \right\} \tag{12}$$

As a corollary, we can prove our key result.

Theorem 3.12: (i) $\Theta_G^{(R)}(t|\hat{\beta})$ is nondecreasing, piecewise-linear, with at most

$$\frac{m^{R-1}}{(R-1)!} + O\left(m^{\max\{1, R-2\}}\right)$$

breakpoints.

Proof. (i) If $t^a < t^b$ clearly $\Theta_G^{(R)}(t^a|\hat{\beta}) \le \Theta_G^{(R)}(t^b|\hat{\beta})$, since under the supply-demand vector $t^b\hat{\beta}$ we can shed demand to replicate the optimal control strategy for $t^a\hat{\beta}$. So Θ^R is nondecreasing.

We analyze the number of breakpoints using induction on R, starting from Lemma 3.7. Assume first that $R = 2$. Note that $\cup_{h=1}^{p} F^h$ is the set of all lines (here recall that p is the number of distinct critical values). Consider the effect of removing from the network, each line in F^1, followed by each line from F^2, and so on, until removing all lines. In this process we will first produce all members of $\mathcal{K}(1)$, and then all members of $\mathcal{K}(2)$, and so on until we obtain all members of $\mathcal{K}(p)$, which are the individual buses.

Prior to its removal, each line j has both ends in the same component K; the removal either creates two new components (if j is a bridge of K) or creates a new component (which differs from K in that line j is not included). Thus the removal process can be represented as a binary tree whose vertices correspond to $\cup_{h=1}^{p} \mathcal{K}(h)$ and whose leaves correspond to the n buses of G^1. Since in a binary tree the number of degree three vertices is less than the number of leaves we conclude that $|\cup_{h=1}^{p} \mathcal{K}(h)| \le m + 2n = O(m)$.

Furthermore, let H be a component in $\cup_{h=1}^{p} \mathcal{K}(h)$. Define $h = \min\{j : H \in \mathcal{K}(j)\}$ and $h' = \max\{j : H \in \mathcal{K}(j)\}$. Note that H will appear in the first term of (12) precisely when $\gamma_h < t \le \gamma_{h'}$; moreover the structure of $\Theta^{(1)}$ established in Lemma 3.7 shows that $\Theta_H^{(1)}$ will contribute at most one breakpoint to $\Theta_G^{(R)}$ in this range for t. The maximum in (12) shows that for each q, one additional new breakpoint is created. Thus, in total, $\Theta_G^{(R)}$ has at most $O(m)$ breakpoints and the result is verified for $R = 2$.

In what follows we assume that $R \ge 3$. Suppose $q = 0$ and thus $i = 1$. Since $\lambda_1 t < \gamma_1$, it follows that no lines are outaged in round 1, i.e. $G^2 = G^1 = G$, and in subsequent rounds no line will be overloaded. Thus, in this case, $\Theta_G^{(R)}(t|\hat{\beta}) = t\tilde{D}$ and there are no breakpoints. For $q > 0$ we proceed using (12). For each $H \in \mathcal{K}(q)$, inductively, $\Theta_H^{(R-1)}$ has at most

$$\frac{m_H^{R-2}}{(R-2)!} + c\, m_H^{\max\{1, R-3\}}$$

breakpoints, where m_H denotes the number of lines in H and $c \geq 0$ is a constant. So (12) implies that subject to $i = q+1$, the number of breakpoints in Θ^R_G is at most

$$1 \; + \sum_{H \in \mathcal{K}(q)} \left[\frac{m_H^{R-2}}{(R-2)!} + c\, m_H^{\max\{1,R-3\}} \right]$$
$$\leq 1 \; + \; \frac{(\,m \,-\, q)^{R-2}}{(R-2)!} \; + \; c\,(\,m\,-\,q\,)^{\max\{1,R-3\}}$$

since $\left(\cup_{h=1}^{q} F^h\right) \cap H = \emptyset$ for each $H \in \mathcal{K}(q)$. Summing this expression over all $1 \leq q \leq p$, we obtain that the total number of breakpoints is at most

$$p \; + \; \sum_{q=1}^{p} \left[\frac{(\,m\,-\,q)^{R-2}}{(R-2)!} \; + \; c\,(\,m\,-\,q)^{\max\{1,R-3\}} \right]$$
$$\leq \frac{(\,m\,-\,1)^{R-1}}{(R-1)!} \; + \; O((m-1)^{R-2})$$
$$+ \; m \; + \; c \sum_{q=1}^{m} (\,m\,-\,q)^{\max\{1,R-3\}}. \qquad (13)$$

For $R = 3$ the last three terms in (13) are $O(m)$ and we are done as desired. For $R > 3$, the last term in (13) equals

$$c\frac{(\,m\,-\,1)^{R-2}}{R-2} \; + \; O(m^{R-3}), \qquad (14)$$

and again we conclude as desired for c large enough. ■

It follows from the details of the proof above that when applying an optimal control, at least one line becomes fully loaded at each round. Such a strategy is likely non-robust. We address this issue below using a stochastic line outage rule and computing an appropriate control.

Despite the apparent shortcomings of the method, and of the simplicity of the proposed control, the ability to compute a global optimum in polynomial time (for fixed R) is a significant asset, especially as a starting point for the simulation-based methods for the general problem that are proposed below. Furthermore, we suspect that the optimal scaling problem can be formulated as single a linear program, in which case it may prove practicable to study *robust* versions of the problem.

IV. First-order methods for the general case

Given a control vector (c, b, s), given by triples (c^r_v, b^r_v, s^r_v), for each demand bus v and each round r, and using the control law given by eqs (4)-(5), denote by $\tilde{\Theta}^R(c, b, s)$ the final demand at termination of the R-round cascade controlled by (c, b, s). Our goal is to maximize $\tilde{\Theta}^R(c, b, s)$ over all controls. This is a nonconcave, in fact very combinatorial, maximization problem (see [4], [11]); it is very large (e.g. if $R = 10$ the (c, b, s) vector has more than 180000 variables in the case of the Eastern Interconnect). Toward this goal we will use an algorithm based on the following template:

Procedure 4.1: First-order algorithm
Input: a control vector (c, b, s).
For $k = 1, 2, \ldots$ **Do**

```
    1. Estimate $g = \nabla\tilde{\Theta}^R(c, b, s)$.
    2. Choose "step-size" $\mu \geq 0$ and update control to
          $(c, b, s) + \mu(g_c, g_b, g_s)$.
    3. If $\mu$ is small enough, stop.
```

Procedure 4.1, a common first-order (steepest ascent) method, should be viewed as a *local search* method with which to explore the neighborhood of a solution. In our setting the procedure could prove expensive, since each evaluation of $\tilde{\Theta}^R$ (including in the estimation of $\nabla\tilde{\Theta}^R$ through finite differences) requires a cascade simulation, each round of which requires two power flow computations in our setup.

A. The stochastic setting

We seek robust algorithms that hedge against the inherent noise in a cascade process as well as the incompleteness of the grid model. Specifically,

(1) The deterministic line outage rule (3) is unsatisfactory in that the outage of a line could be caused by a large number of complex processes (such as tree contacts) that are correlated with flow approaching and exceeding the flow limit, but would be extremely difficult to model from a methodological perspective, let alone calibrate with real data. In fact, exhaustively cataloguing all reasons for line outages is probably impossible.
(2) Further, our control rule assumes real-time observations of data (line overloads). Potentially such observations will include errors, or 'noise'. We need algorithms that are robust in the presence of errors.

In this paper we focus on (1) (though (2) is similarly handled). A simple expedient is to replace our deterministic line outage rule (3) with one incorporating stochastics. Recall that as per "memory" equation (2), for a line j, $\tilde{f}^r_j$ is the moving average at round r of $|f_j|$.

Rule 4.2: STOCHASTIC LINE OUTAGE
Parameters: $0 \leq \epsilon_r \leq 1$ for each round r.
Notation: refer to Template 1.2 and equation (2).
Application: For a line j in G^r:

$$\begin{array}{lll} \text{if} & (1+\epsilon_r)u_j \leq \tilde{f}^r_j, & \text{then } j \text{ outages,} \\ \text{if} & (1-\epsilon_r)u_j < \tilde{f}^r_j < (1+\epsilon_r)u_j, & \\ & \text{then } j \text{ outages with probability } \tfrac{1}{2}, & (15) \\ \text{if} & \tilde{f}^r_j \leq (1-\epsilon_r)u_j, & \text{then} j \text{ does not outage.} \end{array}$$

The parameters ϵ_r should be small, though not necessarily very small (see the experiments below) – the goal here is to smooth out the behavior obtained in the deterministic case. Using rule 4.2 together with Template 1.2 we obtain a stochastic cascade model and $\tilde{\Theta}^R(c, b, s)$ is a random variable. In this case Procedure 4.1 can be recast so as to become a *stochastic gradients* method (see [12], [10]).

Rule 4.2 is what we have implemented in our experiments. However, the expectation $E\tilde{\Theta}^R(c,b,s)$ under this rule is not a differentiable function of (c,b,s), which is a requirement in order to obtain a well-founded algorithm. We outline a theoretically correct version of the stochastic gradients method next.

First, rule 4.2 clearly leads to non-smooth behavior because of the abrupt change between stochastic and deterministic regimes; in fact, even though it is preferable over a purely deterministic rule, it still leads to numerically unstable behavior since it interacts with round-off errors on the part of the solvers. To bypass these issues, we modify the line outage model so that the probability of a line will outage is always strictly positive and strictly smaller than 1. To this effect consider a function

$$F : R_+ \to [0,1), \text{ s.t. } F(0) = 0 \text{ and } F(x) \to 1 \text{ as } x \to +\infty,$$

where the convergence is very rapid. An example is $F(x) = 1 - e^{-Mx}$, for large $M > 0$. Likewise, consider a function

$$G : [0,1] \to [0,1), \text{ s.t. } G(0) = 1 \text{ and } G(x) \to 0 \text{ as } x \to 1,$$

and again with rapid convergence. An example is $G(x) = e^{-Mx}$ for large $M > 0$. We can now modify rule 4.2 as follows: at round r, and given a tolerance $0 \geq \epsilon_r < 1$, line j is outaged with probability

$$\frac{1}{2}G\left(1 - \frac{\tilde{f}_j^r}{(1-\epsilon_r)u_j}\right), \text{ if } \tilde{f}_j^r \leq (1-\epsilon_r)u_j$$

$$\frac{1}{2}, \text{if } (1-\epsilon_r)u_j < \tilde{f}_j^r < u_j(1+\epsilon_r)$$

$$\frac{1}{2}\left(1 + F\left(\frac{\tilde{f}_j^r}{(1+\epsilon_r)u_j} - 1\right)\right), \text{ if } (1+\epsilon_r)u_j \leq \tilde{f}_j^r.$$

In other words, if $\tilde{f}_j^r > u_j$, the outage probability is very large, but is bounded strictly away from 1, and if $\tilde{f}_j^r < (1-\epsilon_r)u_j$ the outage probability is very small but remains strictly positive. By choosing F and G appropriately we obtain an outage model that is arbitrarily close to rule (4.2), but has smooth (or at least differentiable behavior). We call this the *fully stochastic line outage rule*.

In our limited experiments, the fully stochastic line outage rule produces smoother behavior (note that this is smoothness in average, and the speed of convergence to average behavior is adversely affected with, for example, how rapidly $F(x)$ converges to 1 as $x \leftarrow +\infty$). From a modeling perspective the rule is also useful in that it allows us to claim that at some level we are handling the incompleteness of our grid models. Thus, the rule is desirable and useful. Nevertheless, it turns out that from a theoretical perspective, the rule *still* does not guarantee differentiability of $E\tilde{\Theta}^R(c,b,s)$.

The key issue here concerns our control law (4)-(5), and in particular how it interacts with the demand/supply adjustment in Step 3 of our generic cascade template (1.2) (or in Step 6 of the cascade control template (2.1)). This is the critical difficulty, and we proceed by adding a small, random amount of demand to be shed: in round r we scale the demand at bus v by a factor of $(1-\delta_v^r)$ where δ_v^r is a uniformly distributed random variable in the range $(0,\bar{\delta})$ and $\bar{\delta}$ is a very small, fixed value (this is done in addition to any control-dictated shedding, even if it is zero). We refer to this procedure as *randomized load shedding*. We can prove:

Theorem 4.3: Using the fully stochastic line outage rule, and randomized load shedding, $E\tilde{\Theta}^R$ is a differentiable function of the control parameters.

One can sketch a proof of this result by noting that with probability 1, Θ^R is a differentiable function of the control parameters; this is done by induction on the rounds. We will skip details for brevity.

The computation of the (stochastic) gradient of the yield function at a given control vector $(\bar{c},\bar{b},\bar{s})$ can now be described. First, we *sample* a random cascade under the control $(\bar{c},\bar{b},\bar{s})$; modeling shed demand and outaged lines using the smoothed control and rules outlined above. This produces a particular sequence of lines that become outaged; i.e. at round r a certain set S^r of lines is outaged, for $r = 1,2,\ldots,R-1$. Next, we compute the change in yield that results when we perturb the control by a vector $(\epsilon^c,\epsilon^b,\epsilon^s)$ with infinitesimally small entries, while still assuming that set S^r is the set of lines outaged at round r, for each r; this computation gives us the stochastic gradient. We then proceed as in Procedure (4.1), where the line search (Step 2) is done to maximize the expectation of $\tilde{\Theta}^R$.

We stress that we have *not* yet implemented the theoretically correct version of the stochastic gradients method; however we expect that using the original control law (4)-(5) as well as outage rule 4.2 yields reasonably close behavior.

V. Experiments

For our experiments we used a snapshot of the U.S. Eastern Interconnect, with approximately 15000 buses, 23000 lines, 2000 generators and 6000 load buses. The snapshot includes generator output levels, demands, and line parameters. We developed a parallel implementation of our algorithms, using three eight-core i7 machines with 48GB of RAM each. The basic task that is run in parallel are cascade simulations; communication was performed using Unix sockets; linear system solves were done using Cplex 12.0 [8] and Gurobi 4.0 [9], both with all presolve options turned off.

In all the experiments the same approach was employed: first, we interdicted the grid according to a synthetic contingency, then we computed our affine control, and finally we studied the behavior of this control. To obtain contingencies we removed a set of K random lines from the grid, where K is a parameter. The lines were chosen so as to avoid a spanning tree of the grid (thus guaranteeing connectedness of the residual network) and giving priority to high power flow lines.

We ran our algorithm by first solving the optimal scaling problem given in Section III, which requires $\alpha = 1$ in the memory rule (2, and then switching to Procedure 4.1 (using the correct value for α). Additional heuristics were employed prior to the general first-order search, in particular segmenting demand buses into demand quantiles, and insisting that the control parameters be constant in each quantile.

Our first set of experiments, shown in Table I, concern cascades with $R = 4$ rounds. In this table and others, 'wallclock' indicates the total observed running time for the algorithm. Table IV

TABLE I
Performance of algorithm on 4-round cascades, Eastern Interconnect

K	yield, no control	yield, control	wallclock (sec)
1	90.04	95.03	268
2	1.25	50.13	174
5	32.94	81.05	214
10	2.02	36.97	194
20	1.64	27.84	220
50	0.83	16.96	477

Note that in the case $K = 1$ the interdiction has limited effect, but even so the control is able to recover additional demand. In the case $K = 5$ the demand loss in the no-control case is substantial, but so is the benefit of the control. Finally, in the cases $K = 2, 10, 20, 50$ the network collapses but the control sill recovers a significant amount of demand. More experiments of this type will be forthcoming.

In the next set of experiments, given in Table II, we use the case $K = 50$ in Table I to investigate in more detail the behavior of the algorithm as R increases. We used $\alpha = 0.5$ for all these experiments. Note that keeping α constant but increasing R effectively considers cascades that take longer from a 'real time' perspective, thereby giving more power to an agent applying control. If, instead, if we were to increase R while also decreasing α, thus giving more weight to 'history', we would be able to model cascades that last for a fixed period of time, but where the individual rounds encompass shorter spans of time.

TABLE II
Impact of increasing number of rounds on K = 50 case from Table I

R	yield no control	yield gradient	wallclock grid	grad steps
5	4.13	31.86	1340	7
6	2.02	25.86	657	6
7	2.25	25.98	434	3
8	0.78	46.97	3151	10

We see that as expected the yield produced by our algorithm does improve as the number of rounds increases. And in the case of this family of cascades, terminating the cascade early (with the termination step in the last round of Template 2.1) in the no-control case worsens the outcome. This point is further explored in Table III, where the columns labeled "C_k", for $k = 5, \ldots, 8$ represent the controls in Table II. We see that, indeed, the cascade becomes worse in the no control case as the number of rounds increase.

The next set of experiments concern the *timing* of control. We considered a case with $K = 2$ random lines removed

TABLE III
Max line overload at end of each round for K = 50 case from Table II

		Control				
		C_5	C_6	C_7	C_8	No control
round	**1**	6.47	1.83	2.22	3.79	177.83
	2	14.12	1.83	1.57	33.49	122.06
	3	36.79	1.23	1.30	6.90	114.45
	4	1.72	1.14	2.26	6.70	22.47
	5		0.99	1.18	59.33	45.43
	6			1.08	1.98	40.33
	7				1.18	114.90

as above, and $R = 20$ rounds. The random contingency we generated is severe; at the start of round 1, in fact, the maximum line overload is 40.96, indicating that, likely, several lines with low flow limits are overloaded. If no control is applied, at termination the yield is 2.47%. We computed the best control where (i) $c_v^r = b_v^r = 1$ for all v and r, (ii) $s_v^r = 0$ for all v and $10 < r$ (thus, no control is applied after round 10), and (iii) for each $1 \leq r \leq 10$, either $s_v^r = s$ (a fixed constant, to be determined) for all v, or $s_v^r = 0$ for all v. This control was computed using (parallelized) grid-search; though we point out that the optimal scaling problem computes a good approximation to the constant s.

Our control, which we denote by **c20**, only applies control on rounds 2 and 7. In fact, in this cascade, as the total number of rounds is changed, rounds 2, 7 and sometimes 5 always prove optimal, and round 1 is always a round in which not to apply control. To put it simply, in this cascade delaying action appears optimal – in our experiments this has emerged as a frequent pattern in severe cascades.

In Table IV, where "r" indicates round and, for each round,

TABLE IV
Cascade evolutions

	No control				c20			
r	κ	O	I	Y	κ	O	I	Y
1	40.96	86	1	100	40.96	86	1	100
2	8.60	187	8	99	8.60	165	8	96
3	55.51	365	20	98	61.74	303	17	96
4	67.14	481	70	95	66.63	408	44	94
5	94.61	692	149	93	131.08	492	94	93
6	115.53	403	220	91	112.58	416	146	90
7	66.12	336	333	89	99.62	326	191	78
8	47.83	247	414	87	60.95	227	248	77
9	7.16	160	457	85	32.50	72	279	76
10	7.06	245	542	84	9.50	43	292	76
11	37.55	195	606	83	45.28	35	303	76
12	13.04	98	646	82	11.60	10	306	76
13	22.61	128	688	82	3.88	6	310	75
14	10.64	107	715	81	1.46	4	312	75
15	5.03	64	721	81	1.34	1	312	75
16	84.67	72	743	80	1.13	1	312	75
17	32.15	52	756	80	1.38	2	312	75
18	6.50	43	763	80	1.26	1	312	75
19	9.97	85	812	80	0.99	0	312	75
20	32.34	39	812	2	0.99	0	312	75

"κ" indicates maximum line overload at the start of the round, "O" is the number of lines outaged during the round, "I" is the number of islands at the end of the round and "Y" is yield. We see that even though c20 last applies control in round 7, the severity of the cascade quickly dwindles after round 12 – essentially the control provides a "look ahead" capability.

Note the rapid decrease of yield from 80% to 2% in the no-control case. This is due to the termination feature in our cascades that requires all line overloads to be eliminated by the end of the last round; since the no-control cascade has very high maximum overload (32.34), at the start of round 20, the termination rule forces a drastic reduction in yield. Also, the combination of comparatively high yield (up to round 4), high number of line outages, large line overloads and large amount of islanding suggest the possibility that many of the outages involve unimportant lines, and likewise with many of the islands (though of course a 22% yield loss should indicate a severe contingency). Altogether, one wonders if somehow the no-control option *might* be attractive if *enough* time (i.e., rounds) were available.

TABLE V

Further evolution of no-control cascade from Table IV

r	25	28	29	30	31	32	33	34
O	21.63	2.00	5.70	2.50	2.38	1.35	1.07	0.99
Y	79	78	78	78	78	78	78	78

Table V displays the outcome of adding rounds to the cascade. We see that the no-control approach finally yields stability by round 34, attaining yield 78%. This is slightly better (but very close) to what c20 obtained in 20 rounds (and, furthermore, control action under c20 was restricted to rounds 1-10). Nevertheless, the no-control approach experiences significant line overloads as late as round 32. By maintaining high overloads into very late rounds, the no-control strategy becomes more exposed to the unavoidable *noise* that should be taken into account when modeling cascades, which we have up to now ignored. In the following tests we model noise by means of the stochastic fault outage rule (4.2), and set

$$\epsilon_r \;=\; 0.01 + 0.05 * \lfloor r/10 \rfloor. \qquad (16)$$

Note that under this rule, until $r = 20$ a line will experience random behavior only if the flow on that line is within 6% of the flow limit. To investigate the impact of this noise model, we consider controls where the termination takes place before round 20. For $T = 10, 15, 25$, we compute an optimal control required to terminate by round T, and otherwise subject to rules (i)-(iii), that is $c_v^r = b_v^r = 1$ for all v and r, $s_v^r = 0$ for all v and $10 < r$, and for each $1 \le r \le 10$, either $s_v^r = 0.005$ for all v, or $s_v^r = 0$ for all v. We name these controls c10, c15 and c25, respectively.

Table VI presents the comparisons between all the options we have considered. In this table, "DetY" is the yield in the deterministic case ($\epsilon_r = 0$ for all r), "MaxY" and "MinY" are the maximum and minimum yields in all the simulations (resp.), "AveY" is the average yield and "StddY" is the standard deviation of yield.

We can see that under the no control option the average yield is more two standard deviations lower than the average under c20. c20 is arguably superior to c25, though c15 and c20 are possibly comparable. The table illustrates a balance between two competing forces: on the one hand delaying control appears useful, but on the other hand doing so exposes the control to noise. In future work we plan to explore these notions using an implementation of the stochastic gradients algorithm outlined in Section IV-A.

TABLE VI

1000 cascade simulations under stochastic outage rule (4.2) with noise as in (16)

Option	DetY	MaxY	MinY	AveY	StddY
c10	37.49	39.05	0.00	11.81	11.84
c15	72.44	71.85	0.00	33.94	22.51
c20	75.19	76.30	1.17	41.90	27.47
c25	77.23	42.34	1.38	11.99	10.97
no control	77.75	36.04	0.00	7.96	9.33

VI. Conclusions and Future Works

In spite of the significant complexity entailed in modeling cascades, we believe it possible to develop effective computational tools that get at the heart of the matter, which is the timing of control decisions. Another critical matter concerns the modeling of uncertainty. Here we need robust models, since a comprehensive model for noise seems unlikely. In this context, there is an intriguing game-theoretic perspective on modeling controls and noise in a cascade context. A final ingredient in what we plan to test involves deploying much higher levels of CPU power.

VII. Acknowledgments

This work was funded by DOE award DE-SC000267. The author gratefully acknowledges the contribution of Ian Dobson and Ian Hiskens.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, NJ (1993).

[2] Anghel, M., Werley, K. A., and Motter, A. E. "Stochastic model for power grid dynamics," In Proc. of 40th Annual Hawaii International Conference on System Sciences, 113, 2007.

[3] A. Bergen and V. Vittal, *Power Systems Analysis*, Prentice-Hall (1999).

[4] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press.

[5] B.A. Carreras, V.E. Lynch, I. Dobson, D.E. Newman, "Critical points and transitions in an electric power transmission model for cascading failure blackouts," Chaos, vol. 12, 2002, pp 985-994.

[6] B.A. Carreras, V.E. Lynch, D.E. Newman, I. Dobson, "Blackout mitigation assessment in power transmission systems, " 36th Hawaii International Conference on System Sciences, Hawaii, 2003.

[7] B.A. Carreras, V.E. Lynch, I. Dobson, D.E. Newman, "Complex dynamics of blackouts in power transmission systems," Chaos, vol. 14, pp 643-652.

[8] IBM ILOG, Incline Village NV.

[9] Gurobi Optimization, Houston TX.

[10] H.J. Kushner and D.S. Clark, *Stochastic approximation methods for constrained and unconstrained systems.* Springer-Verlag Berlin, (1978).

[11] D.G. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley (1984).

[12] H. Robbins and S. Monro, On a stochastic approximation method, *Annals of Mathematical Statistics* **22** (1951), 400 - 407.