Derivative-based solution of the optimization problem(s) in DeMarco's model

Ahmed Attia
Supervised by: Mihai Anitescu

August 5, 2014

Cascading Network Failure

- To predict cascading failure in large-scale networks, solid understanding of propagations of failures in small-scale networks is vital.

Cascading Network Failure

- To predict cascading failure in large-scale networks, solid understanding of propagations of failures in small-scale networks is vital.
- This allows optimal redistribution of loads and network design.

Network Model(s): I

- Eight-node, Eleven-branch circuit is used as a toy model.
- Model state

$$\mathbf{x} = \begin{bmatrix} \phi \\ \mathbf{q} \\ \gamma \end{bmatrix} \in \mathbb{R}^{26} \tag{1}$$

where

1) ϕ is a vector of nodal flux differences ($\phi \in \mathbb{R}^{7}$),
2) $\mathbf{q}$ is the vector of nodal charges on capacitors ($\mathbf{q} \in \mathbb{R}^{8}$),
3) γ is the failure state of branches ($\gamma \in \mathbb{R}^{11}$),

Network Model(s): II

- DeMarco's original model equations

$$
\begin{aligned}
d\phi &= E_r^T C^{-1} \mathbf{q} \, dt &(2)\\
d\mathbf{q} &= \left(-E_r H_r L^{-1} H_r^T \phi - G C^{-1} \mathbf{q} + i_{\mathrm{in}} \right) \, dt, &(3)
\end{aligned}
$$

Network Model(s): II

- DeMarco's original model equations

$$d\phi = E_r^T C^{-1}\mathbf{q}\, dt \tag{2}$$
$$d\mathbf{q} = \left(-E_r H_r L^{-1} H_r^T \phi - G C^{-1}\mathbf{q} + i_{\mathrm{in}}\right)\, dt, \tag{3}$$

- Stochastic Version of the model

$$\begin{bmatrix} d\phi \\ dq \end{bmatrix} = \mathbf{M} + \mathbf{P} + \mathbf{U} \tag{4}$$
$$= \begin{bmatrix} E_r^T C^{-1}\mathbf{q}(t)dt \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -E_r H_r L^{-1} H_r^T \phi(t)dt \end{bmatrix} + \begin{bmatrix} 0 \\ -GC^{-1}\mathbf{q}(t)dt + \sqrt{2G\tau}\, dW_t \end{bmatrix}.$$

  - i_{in} : the current input, τ : system's temperature,
  - C, E_r, H_r, L, G : constant matrices,
  - M, P, U : system's mass, potential, Ornstein Uhlenbeck process.

Time stepping: A Splitting Solver

The integrator is a composition of maps

$$\begin{bmatrix}\phi(t)\\ \mathbf{q}(t)\end{bmatrix} = \mathbf{P}_{\frac{t}{2}} \circ \mathbf{M}_{\frac{t}{2}} \circ \mathbf{U}_t \circ \mathbf{M}_{\frac{t}{2}} \circ \mathbf{P}_{\frac{t}{2}} \left(\begin{bmatrix}\phi(0)\\ \mathbf{q}(0)\end{bmatrix} \right); \quad (5)$$

$$\mathbf{M}_t \left(\begin{bmatrix}\phi(0)\\ \mathbf{q}(0)\end{bmatrix} \right) = \begin{bmatrix}\phi(0) + E_r^T C^{-1} \mathbf{q}(0) t\\ \mathbf{q}(0)\end{bmatrix}, \quad (6)$$

$$\mathbf{P}_t \left(\begin{bmatrix}\phi(0)\\ \mathbf{q}(0)\end{bmatrix} \right) = \begin{bmatrix}\phi(0)\\ \mathbf{q}(0) - E_r H_r L^{-1} H_r^T \phi(0) t\end{bmatrix}, \quad (7)$$

$$\mathbf{U}_t \left(\begin{bmatrix}\phi(0)\\ \mathbf{q}(0)\end{bmatrix} \right) = \begin{bmatrix}\phi(0)\\ e^{-GC^{-1}t} \mathbf{q}(0) + \sqrt{\tau C \left(I - e^{-2GC^{-1}t} \right)}\; \mathbf{d}\end{bmatrix}, \quad (8)$$

Where t is the step size, and $\mathbf{d}$ is the stochastic force.

Optimization problems

Since we are interested in failure, we ask how the white noise might steer the system towards increasing energy.

- The exact problem:

$$\min_{\mathbf{d}_1,\mathbf{d}_2,\ldots,\mathbf{d}_N} \mathcal{J}(\mathbf{d}_1,\mathbf{d}_2,\ldots,\mathbf{d}_N) = \sum_{i=1}^{N} \mathbf{d}_i^T \mathbf{d}_i \qquad (9)$$

subject to

$$\max\left(E_N\right) > \epsilon. \qquad (10)$$

Solving the exact problem requires solving one optimization problem for each line.

Optimization problems

Since we are interested in failure, we ask how the white noise might steer the system towards increasing energy.

- The exact problem:

$$\min_{\mathbf{d}_1,\mathbf{d}_2,\ldots,\mathbf{d}_N} \mathcal{J}(\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_N) = \sum_{i=1}^{N} \mathbf{d}_i^T \mathbf{d}_i \qquad (9)$$

  subject to

$$\max\left(E_N\right) > \epsilon. \qquad (10)$$

  Solving the exact problem requires solving one optimization problem for each line.

- A proxy problem: replace constraint with:

$$\mathbb{I}_N^T E_N > \epsilon, \quad (\text{Or} \quad E_N^T E_N > \epsilon) \qquad (11)$$

  where E_N is the energy function, and $\mathbb{I}_N$ is a vector of all ones.

Energy Function

$$E_N(\phi_N) = E_N\left(\begin{bmatrix}(\phi_N)_1\\(\phi_N)_2\\(\phi_N)_3\\(\phi_N)_4\\(\phi_N)_5\\(\phi_N)_6\\(\phi_N)_7\end{bmatrix}\right) = \begin{bmatrix}\frac{1}{2}L_1^{-1}\gamma_1\left(-(\phi_N)_1\right)^2\\ \frac{1}{2}L_2^{-1}\gamma_2\left(-(\phi_N)_2\right)^2\\ \frac{1}{2}L_3^{-1}\gamma_3\left((\phi_N)_2-(\phi_N)_3\right)^2\\ \frac{1}{2}L_4^{-1}\gamma_4\left((\phi_N)_1-(\phi_N)_3\right)^2\\ \frac{1}{2}L_5^{-1}\gamma_5\left((\phi_N)_3-(\phi_N)_4\right)^2\\ \frac{1}{2}L_6^{-1}\gamma_6\left((\phi_N)_3-(\phi_N)_5\right)^2\\ \frac{1}{2}L_7^{-1}\gamma_7\left((\phi_N)_1-(\phi_N)_5\right)^2\\ \frac{1}{2}L_8^{-1}\gamma_8\left((\phi_N)_4-(\phi_N)_5\right)^2\\ \frac{1}{2}L_9^{-1}\gamma_9\left((\phi_N)_4-(\phi_N)_7\right)^2\\ \frac{1}{2}L_{10}^{-1}\gamma_{10}\left((\phi_N)_5-(\phi_N)_6\right)^2\\ \frac{1}{2}L_{11}^{-1}\gamma_{11}\left((\phi_N)_6-(\phi_N)_7\right)^2\end{bmatrix} \tag{12}$$

Probability of Failure(s)

We can compute the probability of one failure (at time t_N) from optimal noise vectors $\mathbf{d}_1, \mathbf{d}_1, \ldots \mathbf{d}_N$ via

$$P(\text{Failure}|\mathbf{d}_1, \mathbf{d}_1, \ldots \mathbf{d}_N) = e^{-\frac{\sum_{i=1}^{N} \mathbf{d}_i^T \mathbf{d}_i}{\tau}} \tag{13}$$

Multiple failure is an enumeration problem solved by exhaustive search.

Derivative Information: I

We have reformulated the splitting solver as linear discrete map:

$$\mathbf{x}_i = \mathbf{A}\mathbf{x}_{i-1} + \mathbf{B}\mathbf{d}_i, \tag{14}$$

The blocks of $\mathbf{A}$ are:

$$\mathbf{A}_{1,1} = I + (E_r^T C^{-1})(I + e^{-GC^{-1}h})(-E_r H_r L^{-1} H_r^T)(\frac{h^2}{4}) \tag{15}$$

$$\mathbf{A}_{1,2} = (E_r^T C^{-1})(I + e^{-GC^{-1}h})(\frac{h}{2}) \tag{16}$$

$$\mathbf{A}_{2,1} = \left((-E_r H_r L^{-1} H_r^T)(E_r^T C^{-1})(\frac{h^2}{4}) + I\right)(I + e^{-GC^{-1}h})(-E_r H_r L^{-1} H_r^T)(\frac{h}{2}) \tag{17}$$

$$\mathbf{A}_{2,2} = (-E_r H_r L^{-1} H_r^T)(E_r^T C^{-1})(I + e^{-GC^{-1}h})(\frac{h^2}{4}) + (e^{-GC^{-1}h}) \tag{18}$$

$$\mathbf{A}_{1,3} = \mathbf{A}_{2,3} = \mathbf{A}_{3,1} = \mathbf{A}_{3,2} = \mathbf{0};\ \mathbf{A}_{3,3} = I \tag{19}$$

Derivative Information: II

B reads

$$\mathbf{B} = \begin{bmatrix} (E_r^T C^{-1})\sqrt{\tau C(I - e^{-2GC^{-1}h})(\frac{h}{2})} \\ \left((-E_r H_r L^{-1} H_r^T)(E_r^T C^{-1})(\frac{h^2}{4}) + I\right) \sqrt{\tau C(I - e^{-2GC^{-1}h})} \\ \mathbf{0} \end{bmatrix} \tag{20}$$

the derivatives read

$$\begin{aligned} \nabla_{\mathbf{x}_{i-k}} \mathbf{x}_i &= \mathbf{A}^k \quad \forall k = 1, 2, \ldots, i-1 && (21) \\ \nabla_{\mathbf{d}_{i-k}} \mathbf{x}_i &= \mathbf{A}^k B \quad \forall k = 0, 1, \ldots, i-1 && (22) \\ \nabla_{\mathbf{d}_i} E_N &= (\nabla_{\phi_N} E_N)(\nabla_{\mathbf{d}_i} \phi_N) \\ &= (E_{N\phi})(\nabla_{\mathbf{d}_i} \phi_N) \quad \forall i = 1, 2, \ldots, N && (23) \end{aligned}$$

$E_{N\phi} = \frac{dE_N}{d\phi_N} \in \mathbb{R}^{11\times 7}$ is the Jacobian of the energy functional w.r.t flux differences.

Derivative Information: III

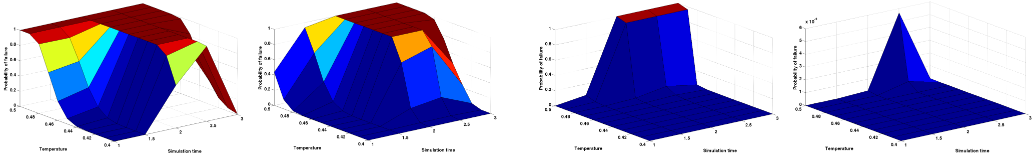
- Gradient of the cost function:

$$\nabla_{[\mathbf{d}_1^T, \mathbf{d}_2^T, \ldots, \mathbf{d}_N^T]^T} \mathcal{J} = 2 \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \mathbf{d}_3 \\ \vdots \\ \mathbf{d}_N \end{bmatrix} . \quad (24)$$

- Gradient of the constraint(s):

$$\nabla_{\phi_N} \left(\mathbb{I}_N^T E_N \right) = E_{N\phi}^T \mathbb{I}_N ; \quad \nabla_{\phi_N} \left(E_N^T E_N \right) = 2 \begin{bmatrix} \left(\nabla_{\mathbf{d}_1} E_N \right)^T E_N \\ \left(\nabla_{\mathbf{d}_2} E_N \right)^T E_N \\ \vdots \\ \left(\nabla_{\mathbf{d}_N} E_N \right)^T E_N \end{bmatrix} \quad (25)$$

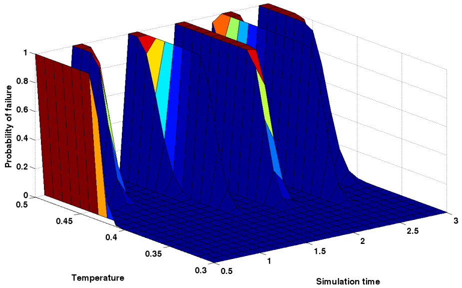
Results I: Probability of failure(s)



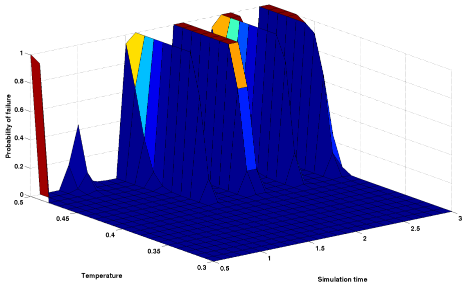
(a) One failure (b) Two failures (c) Three failures (d) Four failures

Figure: Probability of line failure(s). One, two, three, and four failures are plotted.

Results III: Probability of failure(s)



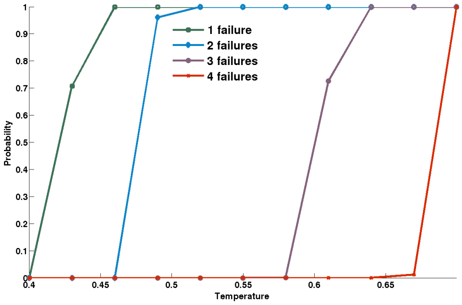
(a) One failure



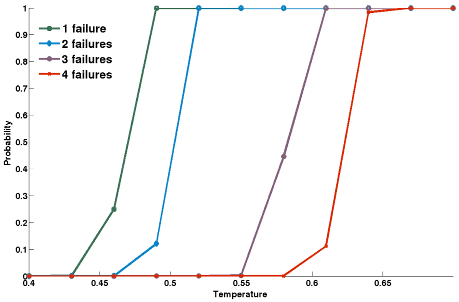
(b) Two failures

Figure: Probability of line failure(s) on higher resolution grid

Results IV: Probability of failure(s)



(a) Simulation time = 0.5



(b) Simulation time = 1

Figure: Relation between probability of branch failures and system's temperature.

Results V: Computational Time

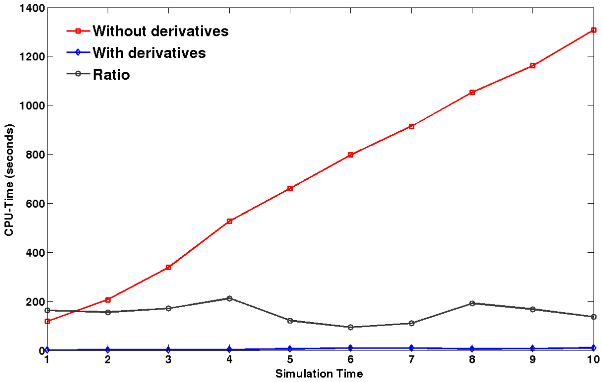


Figure: CPU-time of the optimization step for one failure case with and without derivative information.

Thanks

Questions?