

Designing A Multi-Agent System To Stop Cascading Failures After They Have Started

Sarosh Talukdar

Paul Hines

Carnegie Mellon University

Jan. 2006

Work supported by PSERC, ABB, NSF and CEIC

©Talukdar

What are cascading failures?

What do we know about them?

The electric grid is a stochastic hybrid system. Its operation involves:

- **Contingencies and other uncertainties**
- **Continuous variables (mostly voltages and currents)**
- **Discrete variables (switches whose operation changes the configuration of the grid)**

Existing grids respond to a contingency in one of two ways:

- 1. there is little, or no, change in the configuration,**
- 2. there are massive changes in the configuration.**

The continuous variables always change, though sometimes, only momentarily. The difference that is of concern here, is in the degree to which the discrete variables change.

**The grid's dynamic response to contingencies changes the stresses on generation and transmission devices.
Automatic switches operate to de-energize each and every device that is threatened by excessive stresses.**

A cascading failure is a succession of automatic switching operations that remove equipment from service, leaving the grid with less ability to generate or deliver electric energy.

The switching operations are triggered by those changes in the continuous variables that produce excessive stresses in some generation and delivery devices. Relays sense these excesses, and react to de-energize the threatened devices. De-energized devices are said to have “failed.”

“SECURITY OF SUPPLY IS TOP GLOBAL CONCERN FOR UTILITIES

Blackouts on both sides of the Atlantic have propelled security of energy supply to become the top concern for utilities companies across the world, according to the sixth annual PricewaterhouseCoopers report 'Supply Essentials: Utilities Global Survey 2004'. The report, which presents the views of 148 leading companies across 47 countries throughout Europe, the Americas, Asia Pacific, Africa and the Middle East, indicates that securing power supply has risen from the fourth concern only twelve months ago to the highest ranking issue of 2004.”

Property-1: Cascading failures are expensive

Cascading failures cause blackouts. And blackouts are expensive—from 10 to 50 billion dollars per year, depending on the measure of cost that is used.

Property-2: Cascading failures vary in size

Statistics for 1984-2000 from the North American Electric Reliability Council:

- **533 transmission or generation events**
- **324 (1 every 19 days) had power losses ≥ 1 MW**
- **46 of these (3 per year) were ≥ 1000 MW**

Property-3: Stability is not the primary concern

Very elaborate tools are available for assessing the stability of the continuous part of the electric grid (assuming that the discrete part has been disabled).

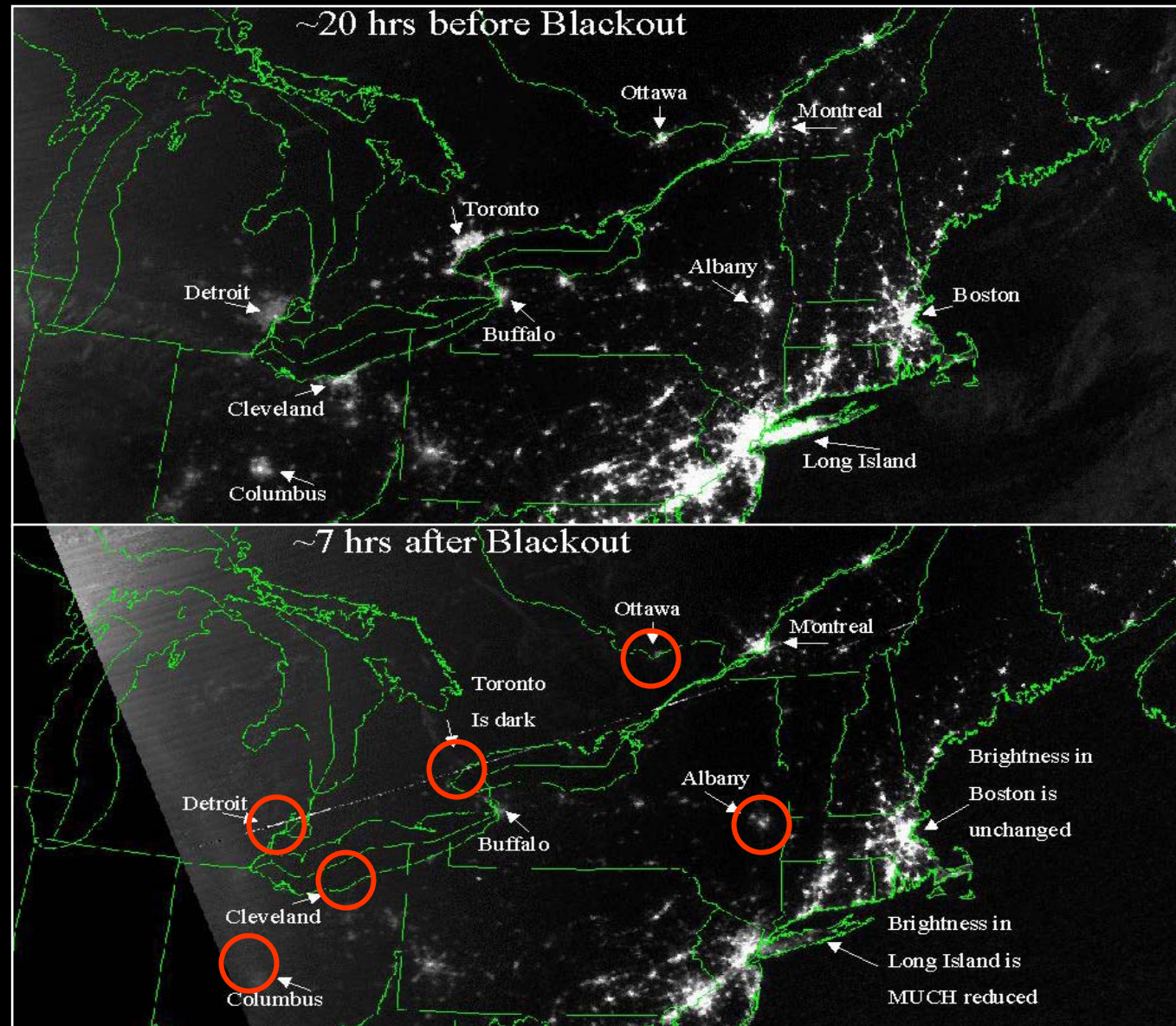
But the discrete part always acts to limit (stabilize) the continuous variables. Existing tools can, at best, provide measures of the likelihood of cascading failures.

To deal with cascading failures one needs to know more—what the size of the cascade will be.

Property-4: Humans cannot make corrections

Once a cascade starts, humans have proved to be ineffective in limiting its spread. It is unlikely that they can be trained to be more effective in the future. The scope and speed of events would seem to be too great for any but automatic reactions.

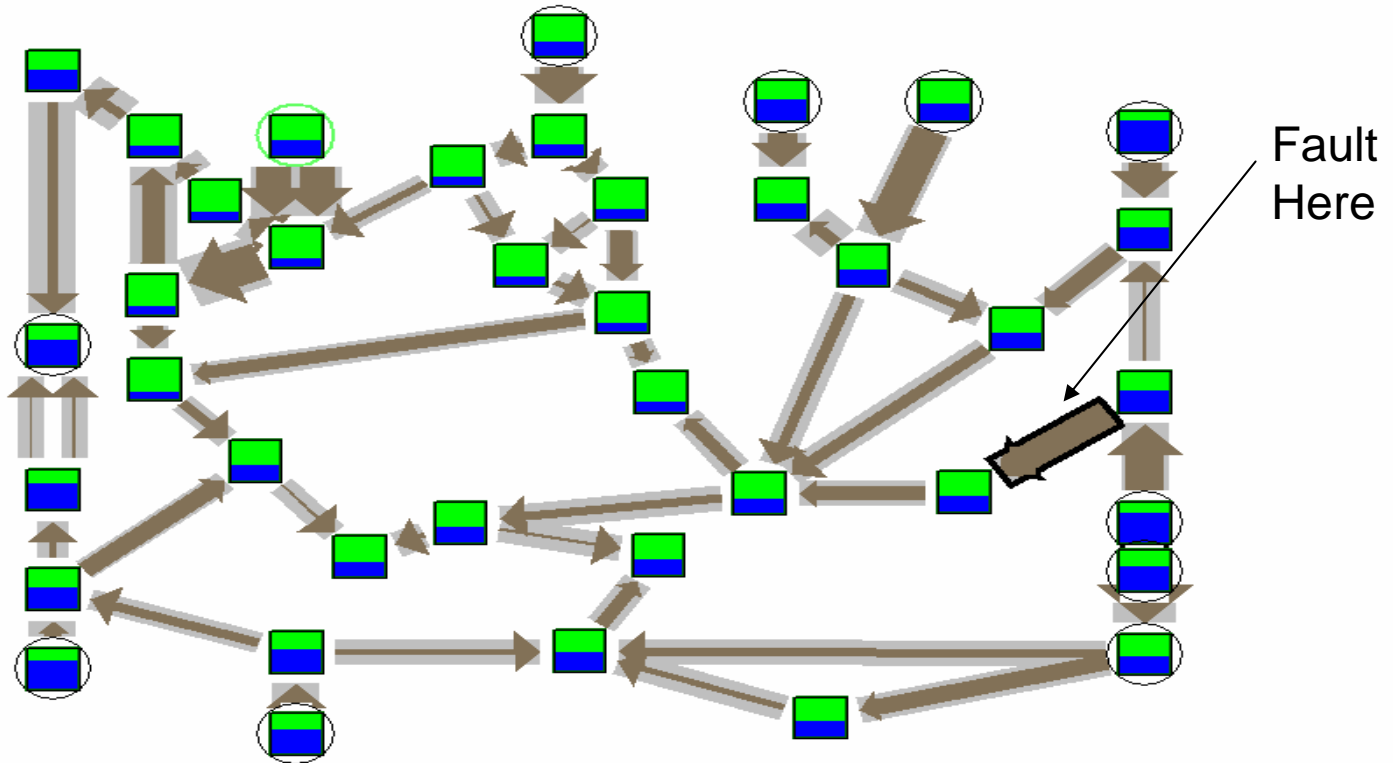
“Before” and “after” pictures” of the region affected by the 2003 blackout



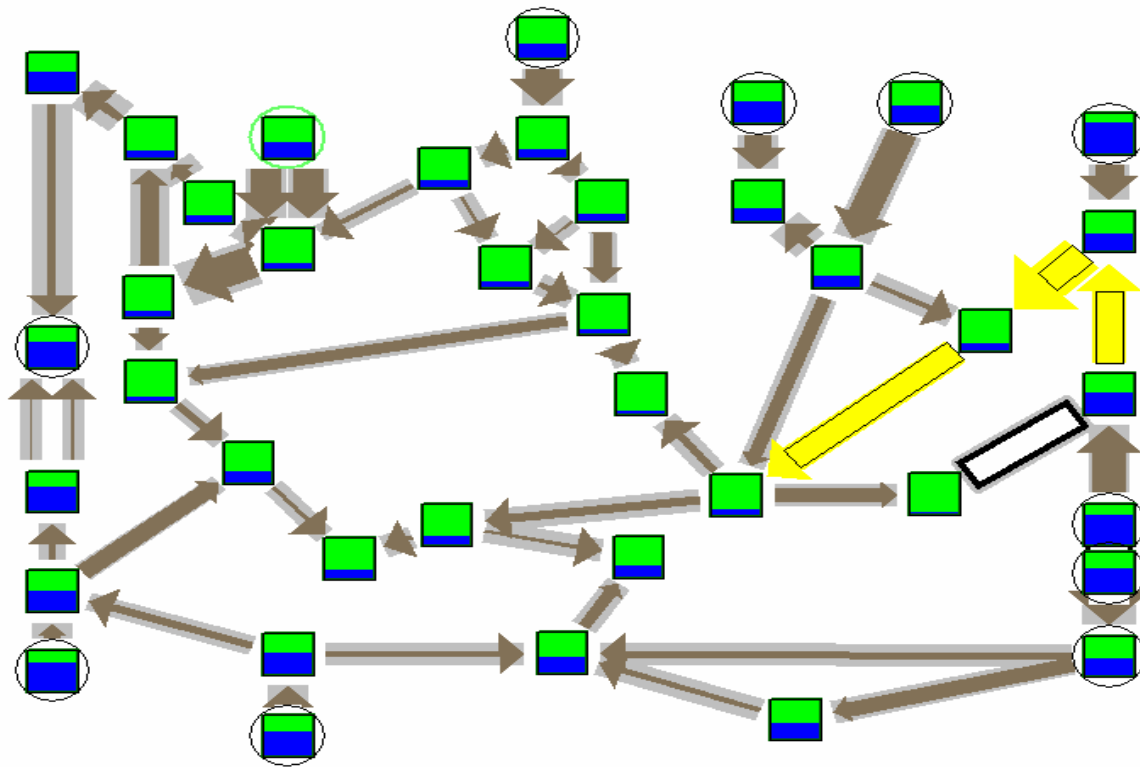
Property-5: Cascading failures are self-limiting

In large grids, a cascading failure almost never proceeds to the ends of the grid. Some part of the grid is left with power.

A simple illustration of a the cascading failure mechanism

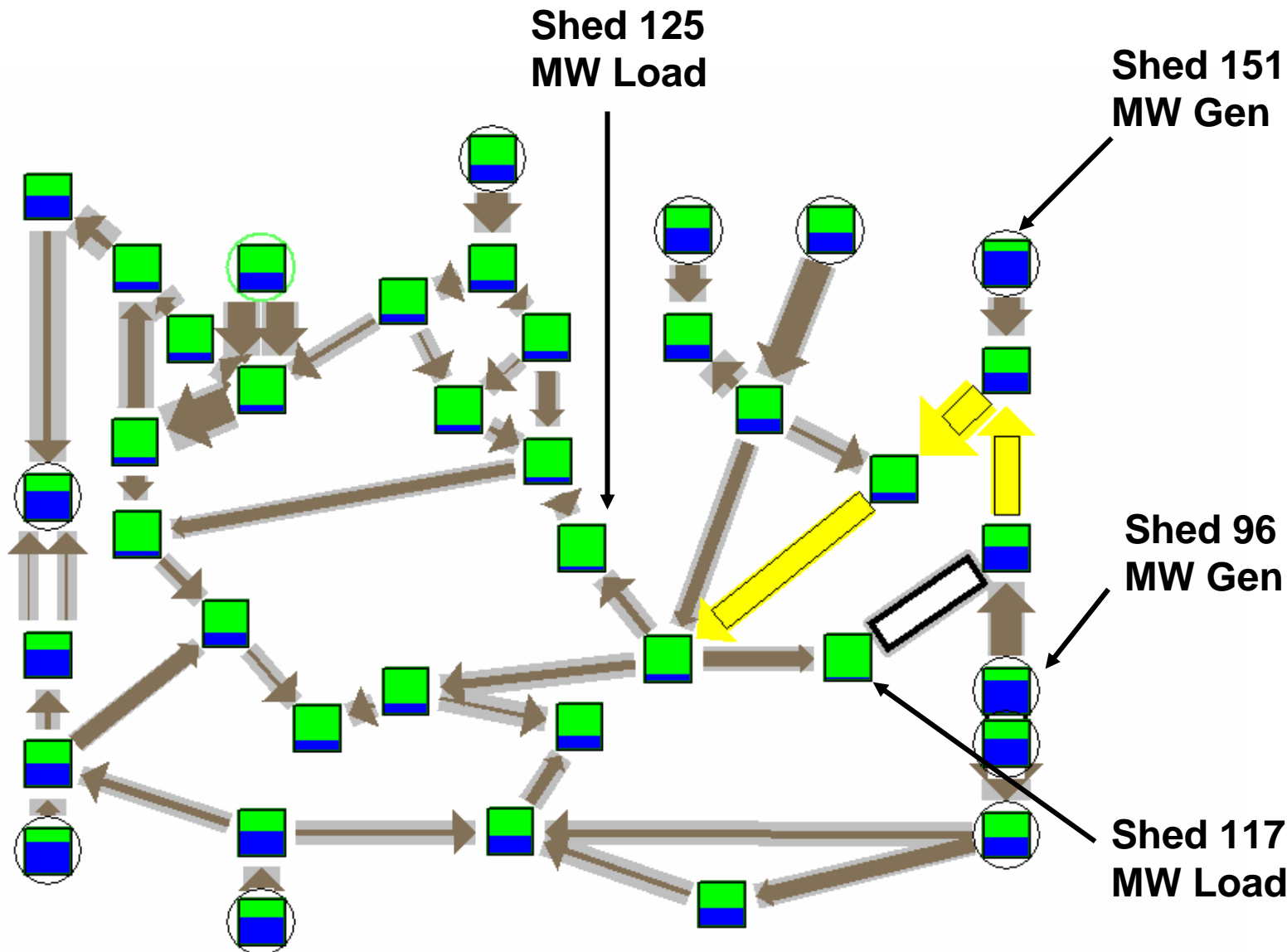


The protection (control) network de-energizes the faulted line



The removal of the faulted line causes excessive stresses in other lines. They are removed. Excessive stresses develop in some of the remaining lines. They are removed. And so on, till much of the system is de-energized.

Invariably, there is a set of relatively inexpensive actions that will stop the cascade



Property-6: Cascading failures can be shortened (made less expensive)

Post-cascade analysis invariably reveals several different sets of relatively inexpensive actions, any of which would have shortened the cascade, had the actions been taken soon after the cascade began.

These sets of actions are cascade-specific. We will refer to the best (least expensive) of these actions as: X_{OPT}

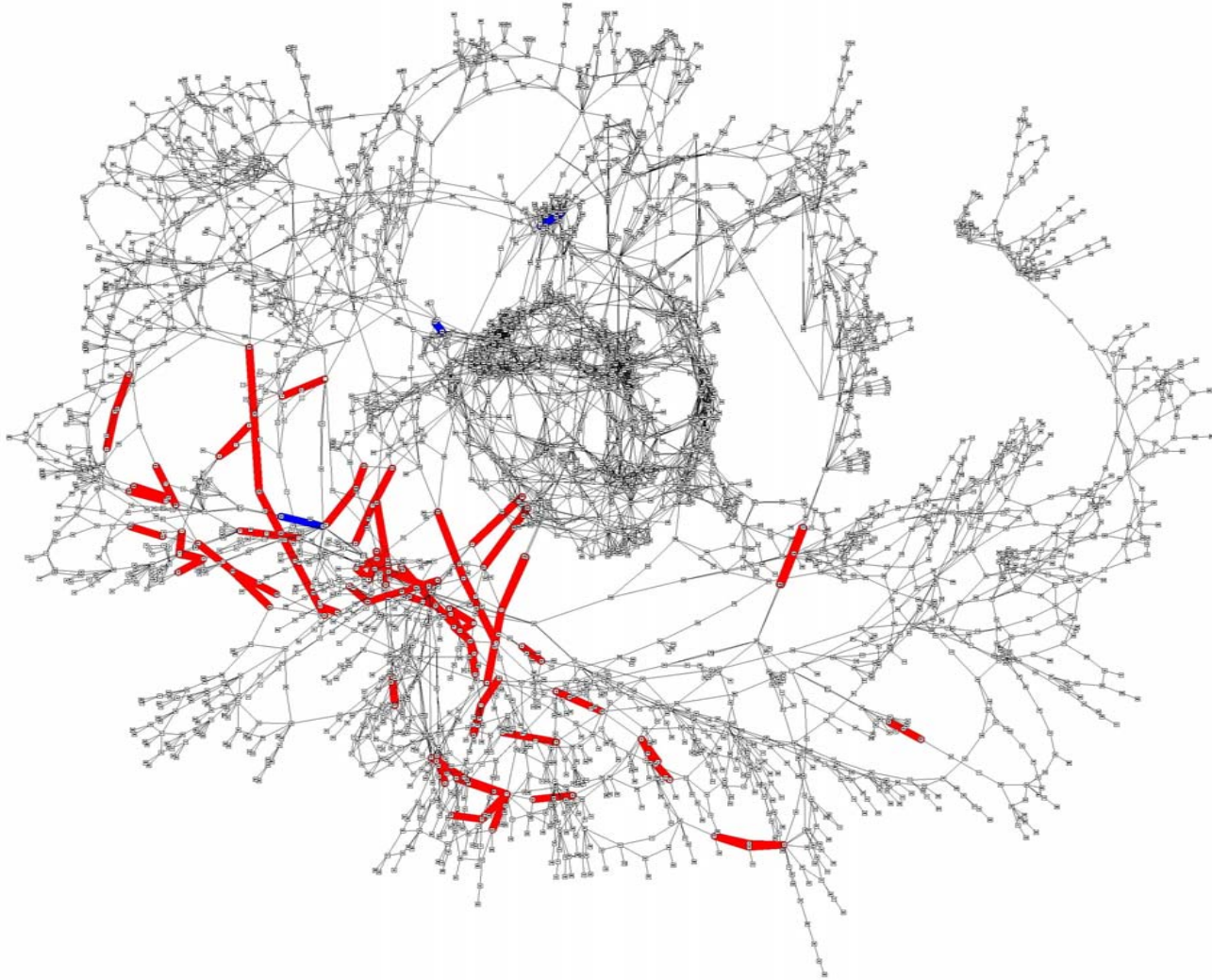
The N-1 Criterion

NERC recommends that power systems be operated so that no single contingency (transmission or generation failure) will precipitate a cascade.

Consequence: cascading failures are caused by bizarre (multiple) contingencies:

- a succession of random failures during which there is no human intervention to re-establish the N-1 criterion.**
- a single random failure in the presence of other, long standing, but hidden failures.**

How cascading failures happen in grids with (N-1) security:
A Multiple Contingency → Excessive Stresses → Outages → Excessive Stresses → More Outages → and so on



Property-7: Multiple contingencies often involve control failures

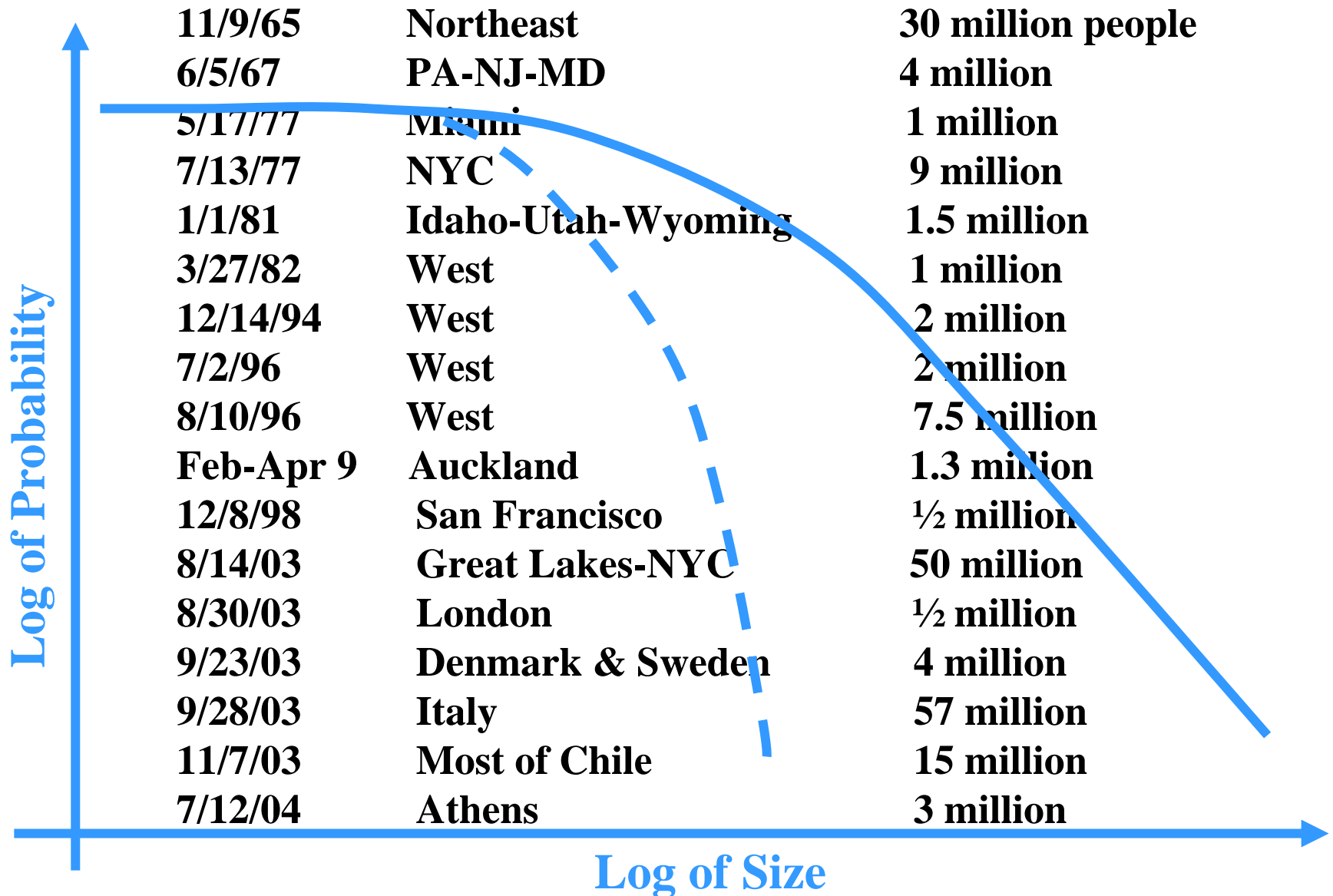
Many, if not most, multiple contingencies consist of a single random disturbance, such as a short circuit, accompanied by multiple miss-operations of the control system (often the result of long-standing, undetected relay failures).

Property-8: The development of a cascading failure proceeds through both short- and long-range effects

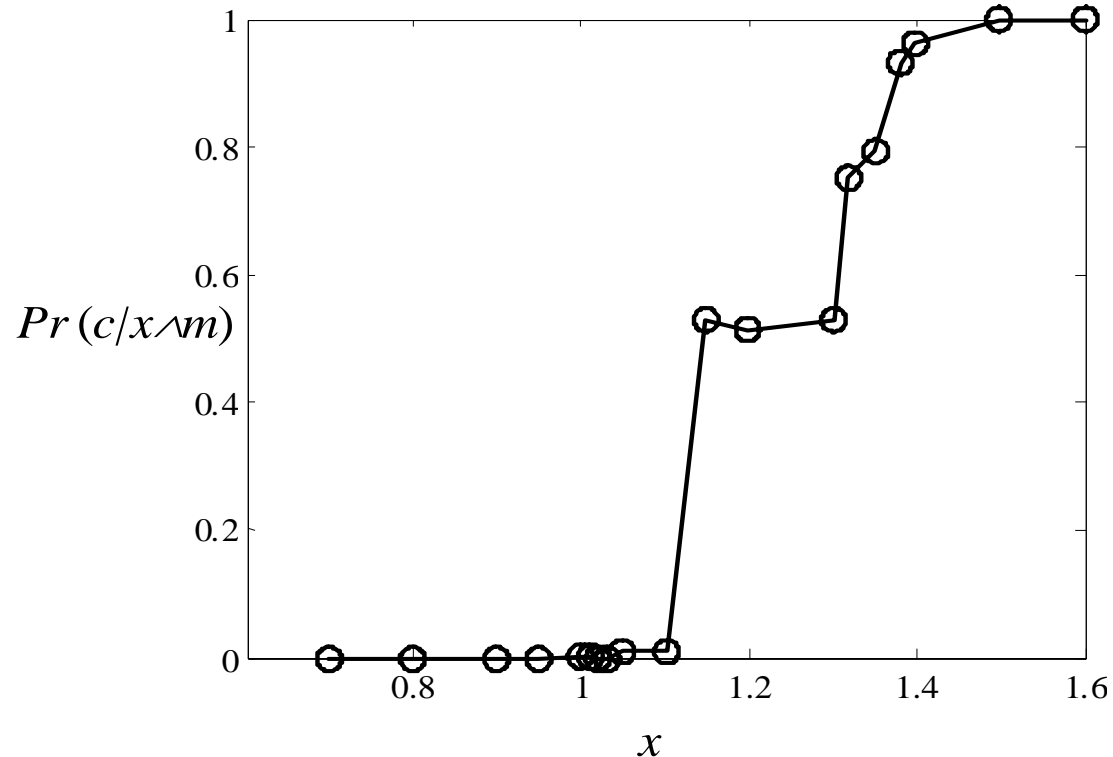
Let S_k and S_{k+1} be successive sets of switching operations in the sequence that represents a cascading failure.

Then S_k and S_{k+1} may be close together in the network, or far apart, or anywhere in between.

Property-9: Multiple contingencies have happened often enough to give the distribution of blackouts a fat tail



There is a critical stress at which the blackout-probability increases sharply. Is the underlying mechanism a phase transition? a relaxation? ...?



The probability, Pr , of a cascading failure plotted against a measure of stress, x . c is a cascading failure of 20 or more transmission lines; m is a random multiple contingency. Each point on the plot is the result of 10,000 simulations of a network with 3357 nodes. The points were generated by Huaiwei Liao.

Property-10: The statistics of cascading failures change abruptly at certain critical points, as happens with some other phenomena that have fat-tailed distributions--forest fires, earthquakes and epidemics, for instance.



Critical, fat-tailed phenomena are often difficult to control, and even more difficult to eliminate.

Consider forest fires.

Some of the events that initiate forest fires--lightning strokes, for instance--are beyond human control.

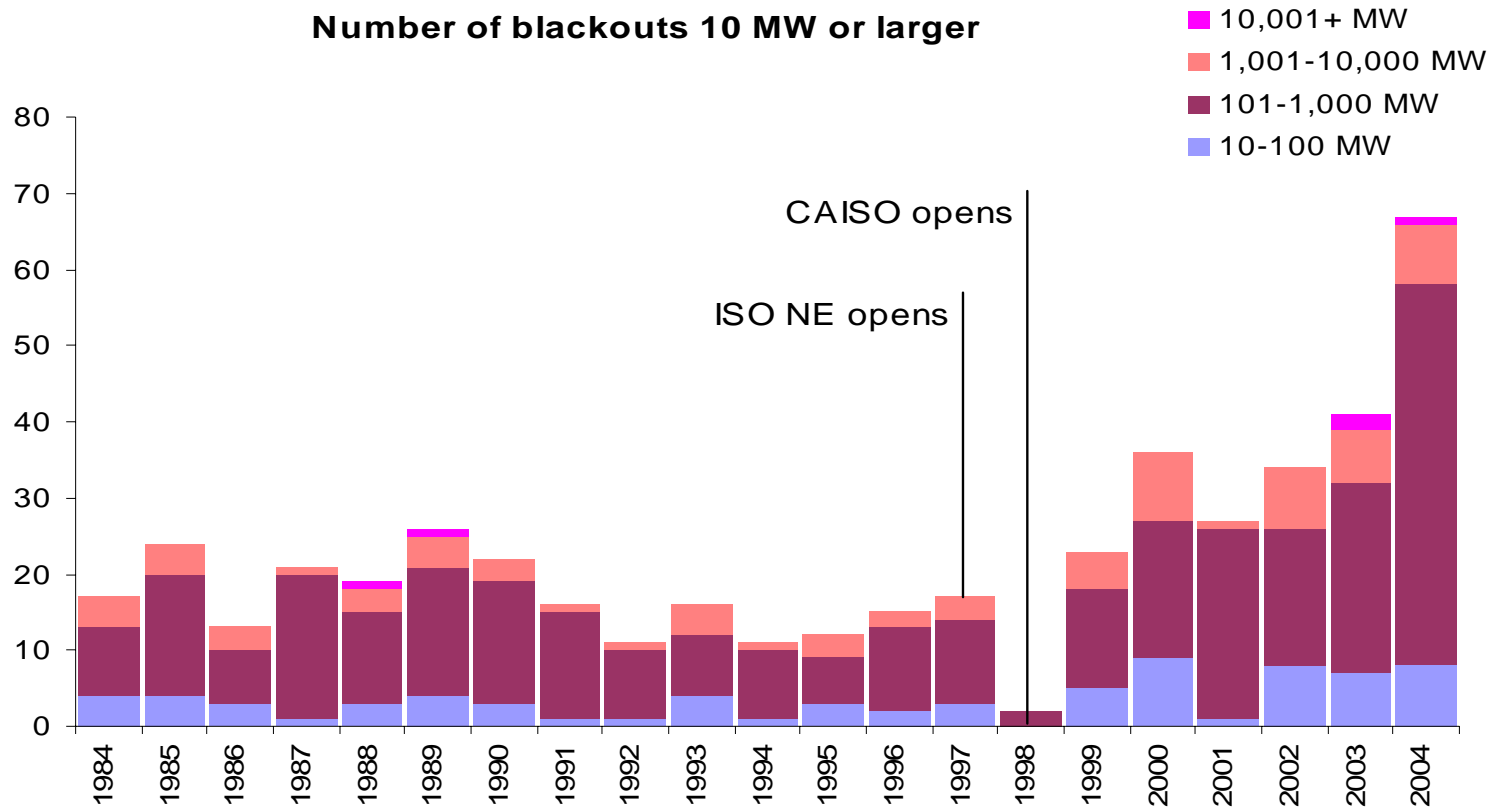
Therefore, as long as there are inflammable forests, there will be forest fires.

Obvious measures to reduce the frequency of the smaller fires often increase the frequency of larger fires, and vice-versa.

Many strategies have been proposed for controlling cascading failures. Some have been implemented. Examples:

- 1. add transmission capacity**
- 2. improve regulations**
- 3. improve coordination**
- 4. better training for human operators**
- 5. better automatic control systems**
- 6. more data collection**
- 7. more data processing**
- 8. load management**
- 9. more conservation**
- 10. more chainsaws**
- 11. RAS's (Remedial Action Schemes)**
- 12. SPS's (Special Protection Schemes)**

But the battle against cascading failures is not being won...



The reason...

Let:

c be the initial conditions of a disturbed power system.

c = [configuration, state, multiple contingency]

y be a strategy

x = y(c) be the actions dictated by the strategy for conditions, **c**

x_{OPT} be optimum actions to stop the cascade caused by **c**

$$\mathbf{x}_{\text{OPT}} \neq \mathbf{x}$$

for any of the proposed strategies

Summary of what we know and can conjecture about cascading failures

- 1. The production and delivery network of the electric grid is a hybrid system. It is described by both continuous and discrete variables. Contingencies (random failures and deliberate attacks) produce excursions of the continuous variables. The control system adjusts the values of the discrete variables to limit these excursions.**
- 2. The control system is hierarchic. The lowest level contains thousands of fairly simple, autonomous agents-
-relays**

- 3. A cascading failure is a sub-optimal reaction of the relays—a succession of automatic switching operations that take devices out of service, leaving the grid less capable of producing or delivering electric energy.**
- 4. Humans generally cannot react fast enough to stop a cascading failure, once it has begun.**
- 5. The conditions that precipitate a cascading failure usually are: a highly stressed network, and a multiple contingency**

- 6. Multiple contingencies usually involve control system failures. They happen often enough to give the distribution of cascading failures a fat tail.**
- 7. The propagation of cascading failures involves both short- and long-range influences**
- 8. Though self-limiting (in size), cascading failures are expensive. And the larger the cascade, the greater its societal cost. Reducing the size of the average failure would reduce their total cost.**
- 9. But cascading failures appear to belong to the category of critical phenomena whose ill effects are notoriously difficult to reduce.**

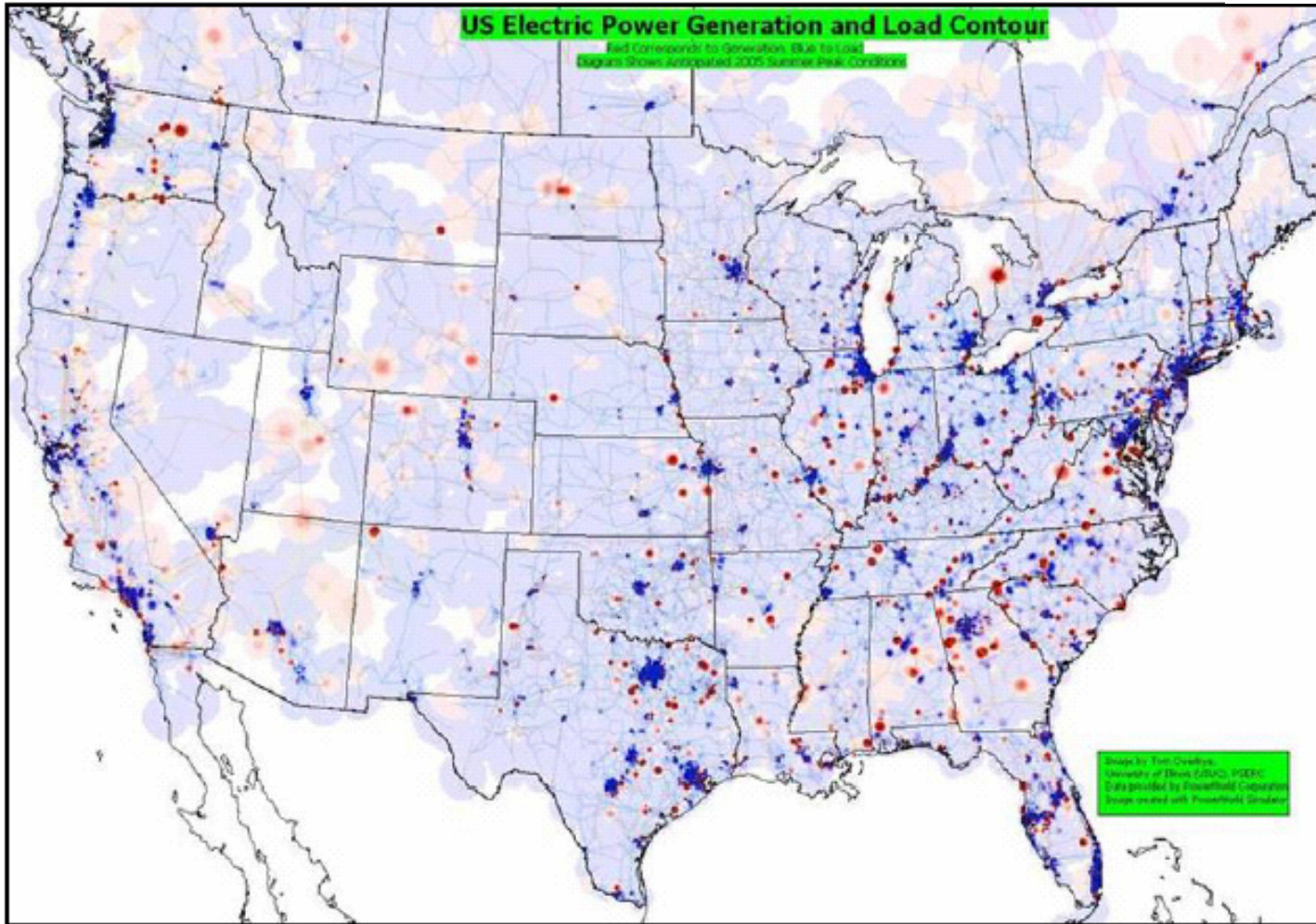
- 10. Nevertheless, there is a cascade-specific set of actions, x_{OPT} , by which the cost of the cascade can be minimized.**
- 11. So far, x_{OPT} has only been calculated post-mortem, long after the cascade is over.**
- 12. We have not been winning the battle against cascading failures. Of the many strategies that have been proposed, only RAS's and SPS's make even an attempt to calculate x_{OPT} . But they do not quite succeed because they do not use the right problem formulation, and they rely too heavily on centralized data collection and computation.**

10. To calculate x_{OPT} fast enough for it to do any good, requires a multi-agent system (many autonomous agents, distributed over the grid, all working asynchronously and in parallel).

That is, the calculation requires the upgrade or replacement of the existing multi-agent system of relays and other low level controllers.

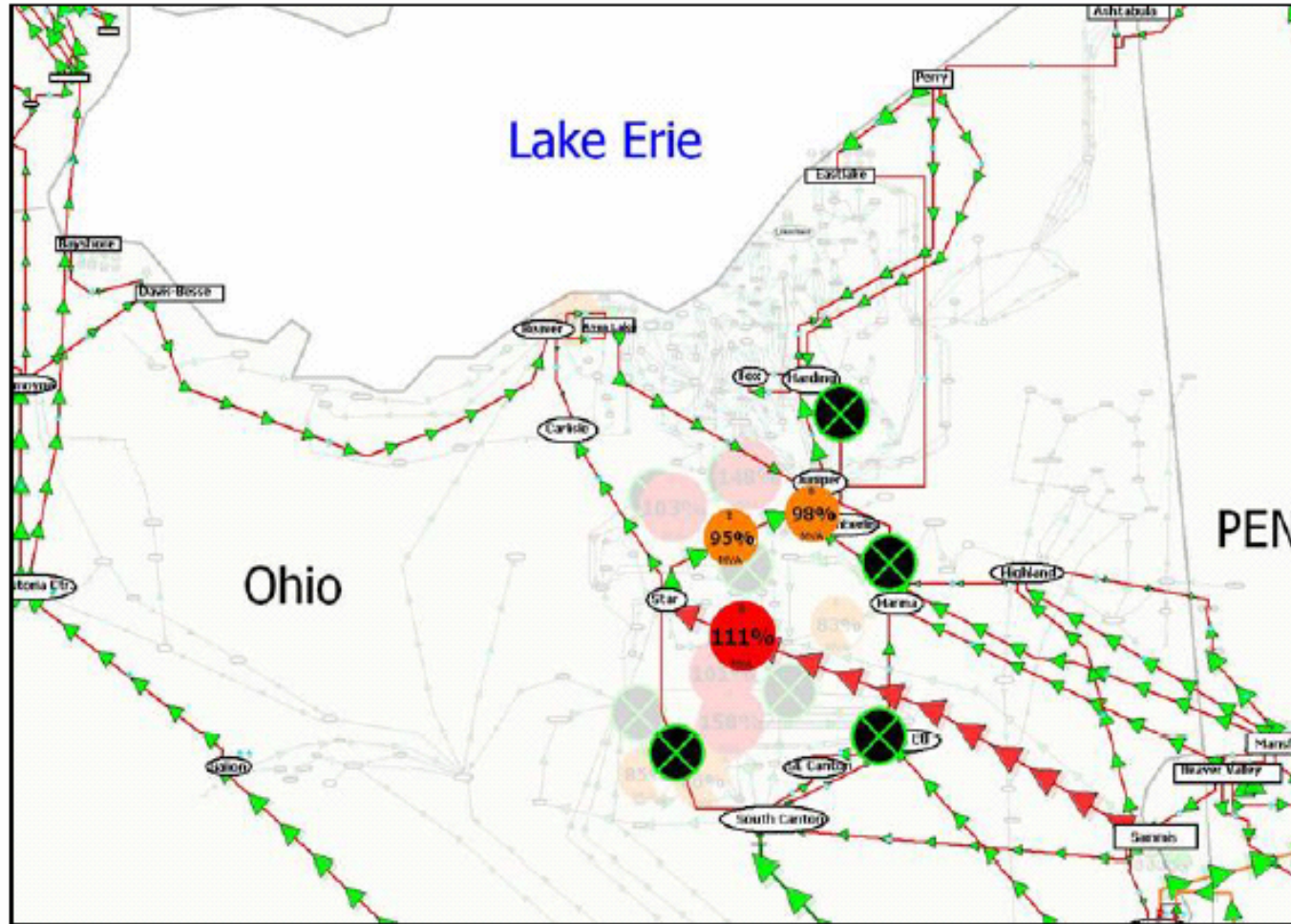
Designing a multi-agent system to calculate x_{OPT}

A typical system-state



From Overbye, Meliopoulos, Wiegmann, Rantenan

8/14/2003: A multiple contingency in the making



From Overbye, Meliopoulos, Wiegmann, Rantanan

Getting the problem formulation right is critical

After all, the best possible solution to the wrong problem is still the wrong solution.

In what follows, we will formulate several problems; some to design strategies, others, to calculate actions (\mathbf{x}_{OPT}).

Notation

Let:

$C = \{c\}$ be the space of all possible initial conditions of the grid. $c = [\text{configuration, state, multiple contingency}]$

$Y = \{y\}$ be a space of control strategies.

$x = y(c)$ be the actions (values of the controllable variables) dictated by control strategy, y , for conditions, c .

$g_k(x, c)$ be the stress on device- k as a result of condition, c , and actions, x .

$f(x, c)$ be the societal cost of condition, c , and actions, x .

More on actions (decision variables)

A power system is a hybrid system—some of its decision variables are continuous, others are discrete.

Let \mathbf{x} , the decision vector, be partitioned: $\mathbf{x} = [\mathbf{x}_1, \mathbf{T}]$, where
 \mathbf{x}_1 is a vector of the continuous decision variables
 $\mathbf{T} = [t_1, t_2, \dots, t_L]$ is a vector of times
 t_k is the time at which the k -th discrete decision variable
is toggled (changes from 0 to 1, or from 1 to 0)

The assumption is that the discrete variables are binary, and each is changed at most once during the control horizon.

P (protection) is a constraint satisfaction problem...

Find a strategy, $y \in Y$, such that:

$$g_k(y(c), c) \leq R_k \quad \text{for all } k \text{ and all } c \in C$$

Stress on device-k



Stress-Threshold beyond
which device-k could be
damaged



A solution to P:

The existing strategy, y_P , is to de-energize each and every device that approaches its threshold.

y_P is a localized strategy. It uses only data which are local to device-k to determine the actions for device-k. Thus, y_P decomposes P into **localized, uncoupled subproblems**, one for each device.

Another advantage of y_P is that it is easy to verify. y_P is unaffected by C, except for hidden failures and controller miss-operations. If we neglect this subset of conditions, then y_P can be verified device-by-device.

The disadvantage of y_P is that it takes no account of the number of devices that are de-energized

P+CFC (protection + cascading failure control), can be formulated as a constraint satisfaction problem or as a single-objective problem. A constraint satisfaction formulation is:

Find a strategy, $y \in Y$, such that:

Maximum allowable cost per incident

Societal cost

$$f(y(c), c) \leq \$$$

Stress on device-k

$$g_k(y(c), c) \leq R_k$$

Stress-threshold

Set of operating conditions

for all k and all $c \in C$

In other words, P+CFC is:

Find a strategy, y , such that

$$\mathbf{x}_{\text{OPT}} = \mathbf{y}(\mathbf{c}) \text{ for all } \mathbf{c} \in \mathbf{C}$$

There are no rigorous methods for solving $P+CFC$.

Y , and particularly, C are far too large.

f , the societal cost, is a global attribute.

The only practical alternative is to use heuristics, intuition or revelation to generate a solution, and then, verify this solution as best one can.

Verification

When a solution is obtained by non-rigorous means, the solution is only as good as the tests by which it is verified.

For P+CFC, the tests cannot be complete or absolute until there is a breakthrough in testing technology.

At best, one can verify only a small sub-set of the possible operating conditions, say (N-2) or (N-3).

The Aug. 14, 2003 blackout was an (N-k) event, where k was about 25. (See Fig 6.1 in the Final Report On The August 14, 2003 Blackout In The United States And Canada.)

An aside...

The designs of complex artifacts are difficult to verify.

For some artifacts elaborate verification technologies have been developed—bridges, computers, drugs, and airplanes, for instance.

For other artifacts verification technologies are primitive or non-existent—for instance, legislation, electricity markets, and theories of psychotherapy.

The complete verification problem for P+CFC:

Given a control strategy, y^* , show that

$$f(y^*(c), c) \leq \$$$

$$g_k(y^*(c), c) \leq R_k$$

for all k and all $c \in C(y^*)$

This problem is impossible to solve because C , the space of all operating conditions, is very large. And C depends on y^* , the control strategy to be verified. C grows dramatically with the complexity of y^* . Remember that many, if not most multiple contingencies involve miss-operations of the control system.

At best, one can verify only a small sub-set of the operating conditions, and with only approximations to stress and cost.

Partial verification:

Given y^* , a control strategy, show that

$$F(y^*(c), c) \leq \$$$

$$G_k(y^*(c), c) \leq R_k$$

for all k , and all $c \in C'$

where F and G_k are approximations to f and g_k , and C' is a small sub-set of C

The dangers of partial verification

- 1. Untested conditions can, and invariably will, occur**
- 2. There is an unavoidable tendency for the designers to optimize the solution for the conditions to be tested. Conjecture: this makes the system more vulnerable to some of the untested conditions.**

A measure of solution-quality with partial verification

**Control system y1 is better than control system y2
if y1's actions are less costly than y2's actions, that is, if:**

$$\sum_c F(y1, c) < \sum_c F(y2, c)$$

and

$$G_k(y1, c) \leq R_k, \quad G_k(y2, c) \leq R_k$$

for all $c \in C'$

**A control strategy being designed by
Ph.D candidate, Paul Hines**

A heuristic for solving P+CFC:

1. Formulate A , the action problem whose solution is x_{OPT} .
2. Decompose A into subproblems, such that the **optimal solutions of the subproblems constitute the optimal solution of A** , and the couplings among subproblems are as weak as possible.
3. Assign each subproblem to a relay or other distributed agent. Choose **how much autonomy** each agent is to have in tackling its subproblem.

4. Provide the autonomous agents with the means to solve their subproblems **with locally available information:**
5. Select C' , and verify the design over C'
6. Repeat till the verification succeeds

This heuristic is fairly general for multi-agent systems

Difficulties with a multi-agent approach

How can locally optimal solutions be made globally optimal?

The difficulties are:

Problem A cannot be decomposed into localized and decoupled subproblems.

Therefore, the optimum actions of agent-k depend on:

- state and configuration variables it can't measure**
- things it can't know (what the other agents will choose to do).**
- other things it can't know with certainty (the future effects of its actions and those of the other agents).**

General ways around these difficulties: prediction, feedback, learning, and cooperation.

Comparing P+CFC, the strategy problem, with A, the action problem

C+CFC and A are similar in many ways.

But they differ in scope, and particularly, in decision variables.

C+CFC is: find the best strategy for all possible conditions.

A is: choose the best actions for the existing conditions

Example: Car design.....

**Strategy for avoiding all potholes: give the driver ABS brakes and
4-wheel steering**

**Action by driver to avoid pothole # 6: full braking and 30°
right turn**

A, the action problem

**Measure conditions, c , and
calculate actions, x , so that:**

$$\mathbf{f}(x, c) \leq \$$$

$$\mathbf{g}_k(x, c) \leq \mathbf{R}_k \quad \text{for all } k$$

A decomposition of \mathbf{A} into $[\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_K]$, subproblems for each of the K agents. (This seems to be a good decomposition. But it took a year to find.)

Partition \mathbf{x} into parts local to each agent: $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K]$

Let $\mathbf{X}_k = \mathbf{x} \setminus \mathbf{x}_k$.

\mathbf{A}_k , the subproblem for the k -th agent is:

- 1. Measure the local parts of \mathbf{c} , estimate the rest of \mathbf{c} .**
- 2. Predict \mathbf{X}_k .**
- 3. Find \mathbf{x}_k so that:**

$$\begin{aligned} f((\mathbf{x}_k, \mathbf{X}_k), \mathbf{c}) &\leq \$ \\ g_k((\mathbf{x}_k, \mathbf{X}_k), \mathbf{c}) &\leq \mathbf{R}_k \quad \text{for all } k \end{aligned}$$

In this decomposition $\mathbf{A}_k > \mathbf{A}$

Means by which agent-k solves its subproblem, A_k

Estimate distant parts of c : Learning, Cooperation

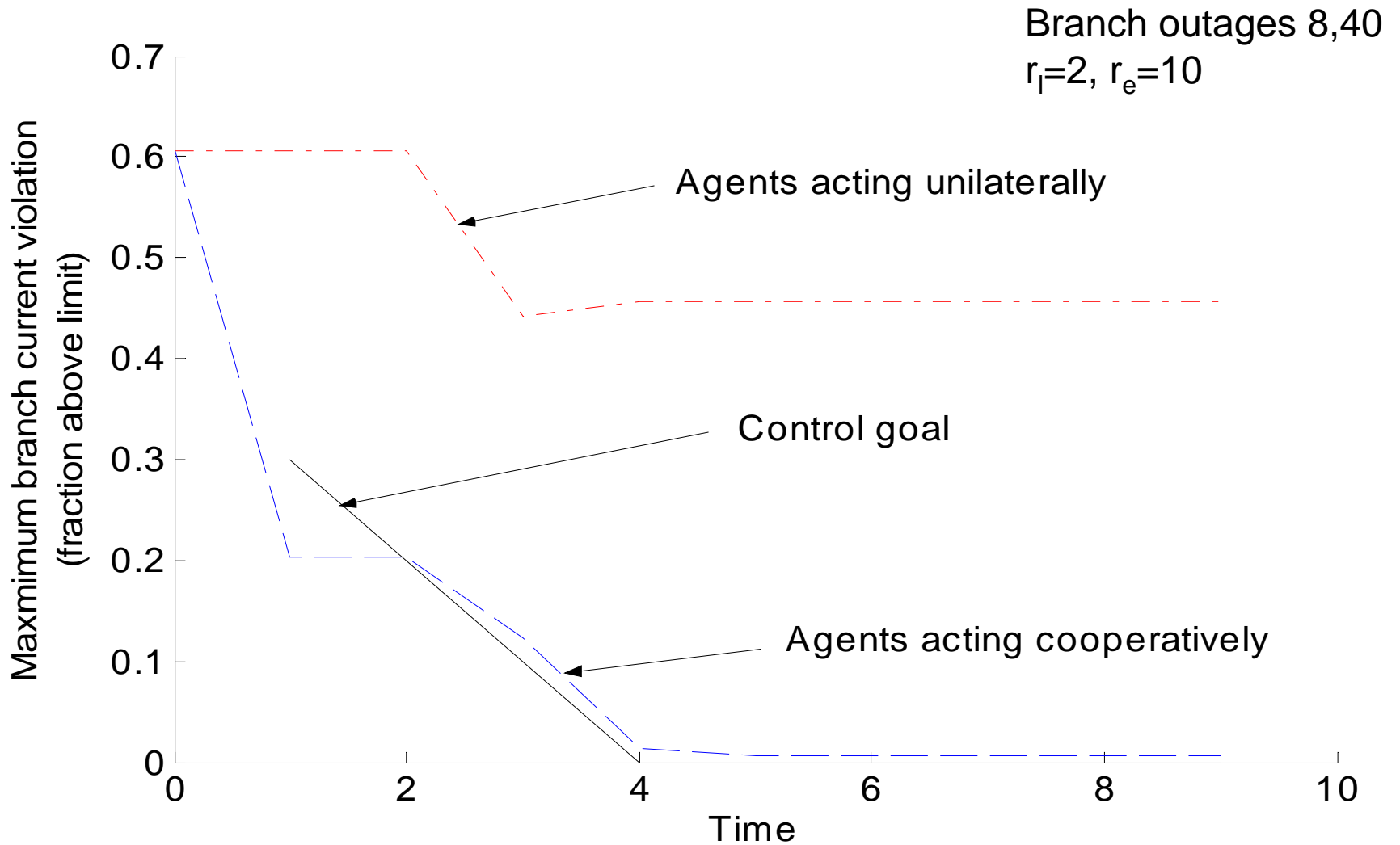
Predict X_k : Learning, Cooperation

**Predict effects of x_k and X_k on future costs and stresses:
DMPC (Distributed Model
Predictive Control)**

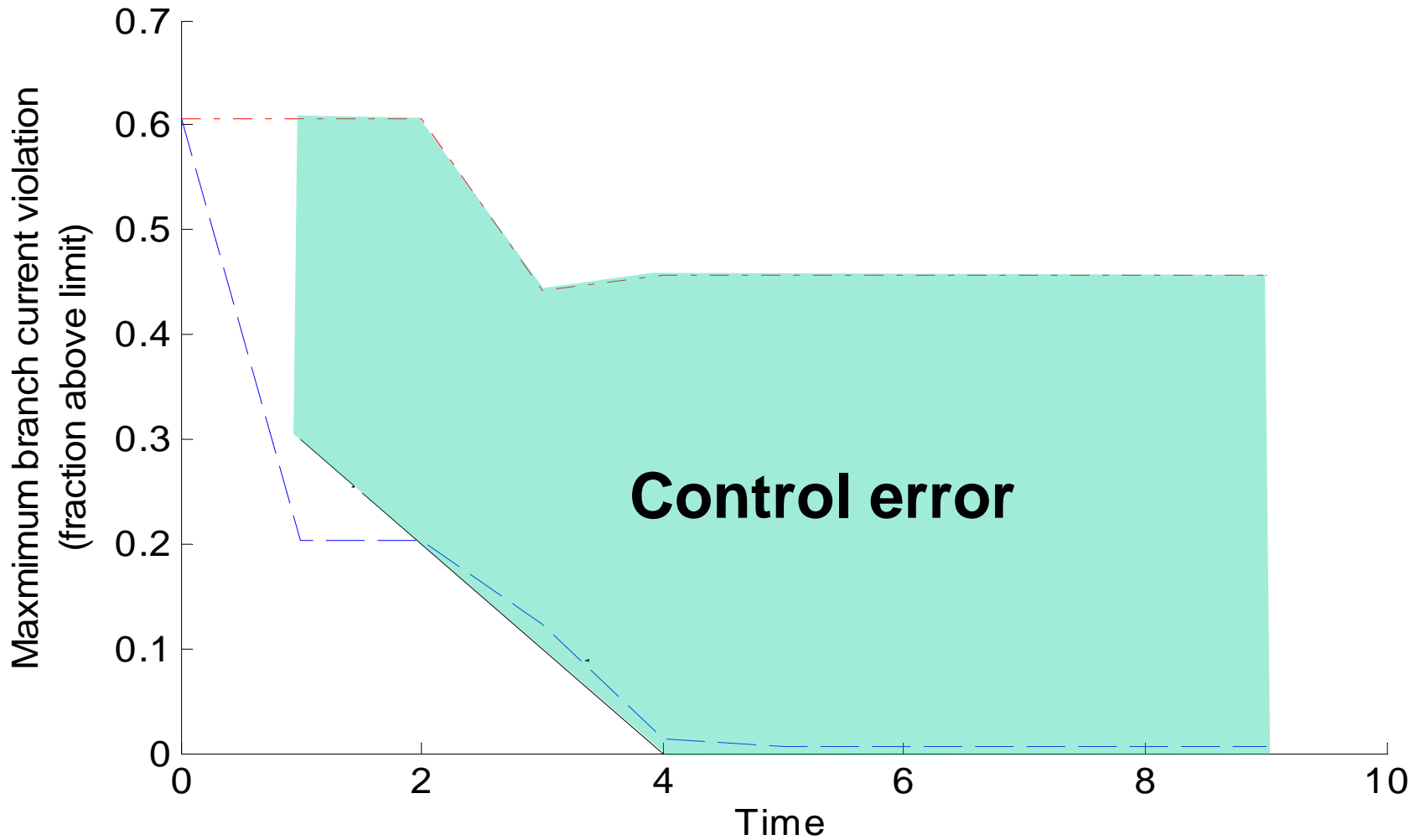
Meanings:

- 1. Learning: converting experience into competence**
- 2. Cooperation: Two agents cooperate when their problems are coupled and they exchange useful information**
- 3. DMPC: ...**

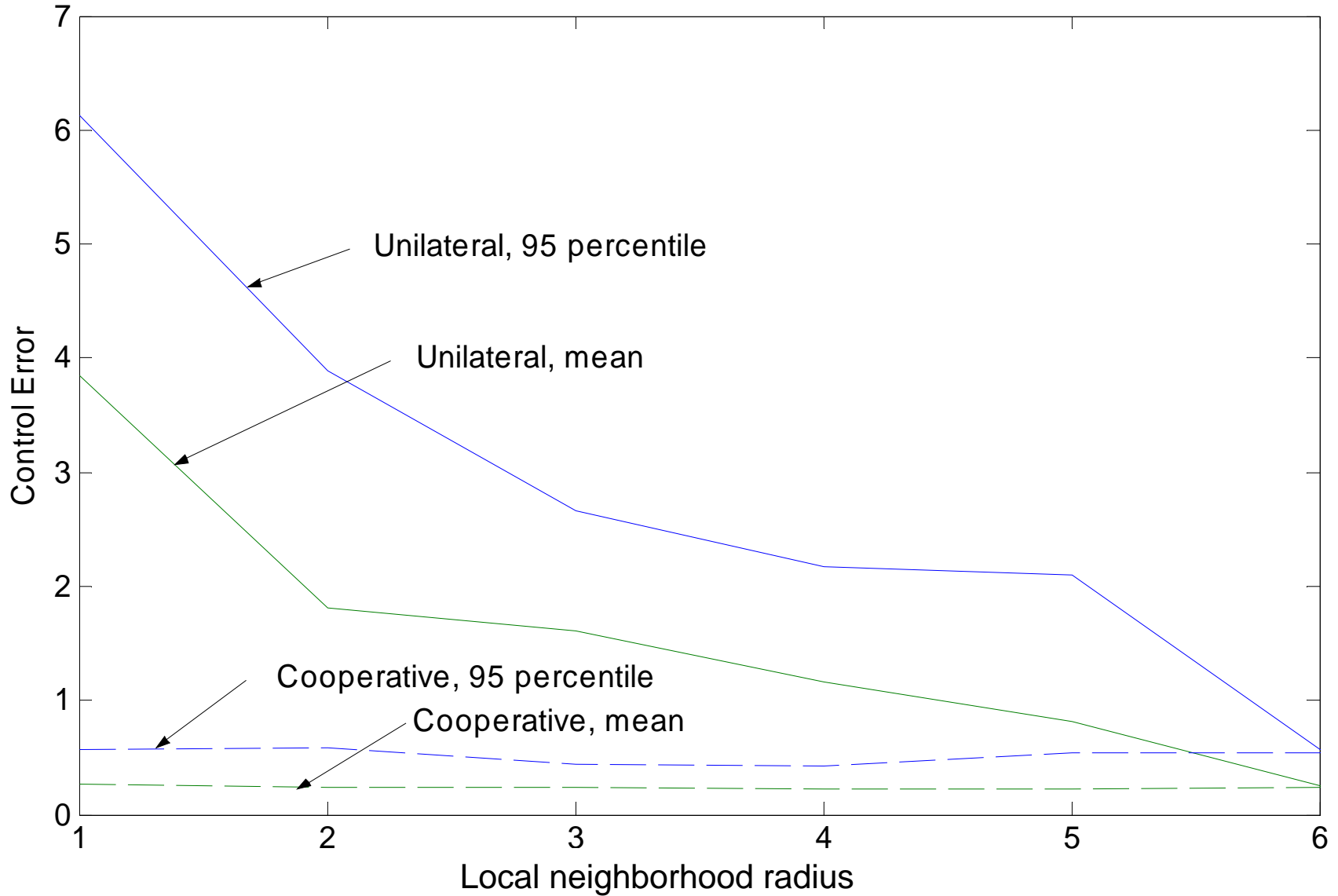
Typical results



Control error...



Control error vs. communication



Conclusions

- 1. Distributed autonomous agents that cooperate can stop cascading failures and reduce their social cost**
- 2. Many improvements can be made to the cooperation scheme**