Distributed Model Predictive Control for Electric Grids

Paul Hines, Dong Jia, and Sarosh Talukdar

***Abstract*—Cascading failures cause blackouts with high social costs. A cascading failure can be thought of as an alternating sequence of equipment outages and constraint-violations. We describe a network of fast-acting, autonomous agents for shortening such sequences. The agents work by eliminating violations before they can cause further outages. They make their decisions with DMPC—a distributed adaptation of the Model Predictive Control technique. Each agent has a suite of models, specialized for its location in the grid. It uses these models to predict what the other agents will do and how the grid will respond. Each agent optimizes its decisions with respect to the predictions. In tests on small grids, these prediction-based optima come close to the true, global optima. In other words, the agents seem able to make good decisions. Future work includes extending the tests to larger grids, and augmenting DMPC with cooperation and automatic learning.**

I. Introduction

A cascading failure is a progression of equipment outages, one outage propagating another. Long progressions result in large blackouts and usually begin with a bizarre, often compound, disturbance, such as a short circuit whose effects are compounded by several relay misoperations. Though relatively rare, such compound disturbances occur frequently enough to give the distribution of large blackouts a fat tail. Large blackouts happen much more frequently than an exponentially falling tail, as is the tail of a normal distribution, would predict [1].

We will not try to prevent all cascading failures. Indeed, there is reason to believe that complete prevention is impossible [1]. Briefly, the set of all possible compound disturbances is very large. One can design and test measures to make the grid invulnerable to only a small subset of the possible disturbances. At best, this leaves the grid still vulnerable to a great many disturbances. At worst, it makes the grid more vulnerable to the untested disturbances.

This work was supported in part by ABB Corporate Research, through the Carnegie Mellon Electricity Industry Center (www.cmu.edu/ceic).

Paul Hines is with the Eng. & Public Policy Dept., Carnegie Mellon U., Pittsburgh, PA 15213 USA (e-mail: phines@cmu.edu).

Dong Jia is with the Elect. & Comp. Eng. Dept., Carnegie Mellon U., Pittsburgh, PA 15213 USA (e-mail: djia@andrew.cmu.edu).

Sarosh Talukdar is with the Elect. & Comp. Eng. Dept., Carnegie Mellon U., Pittsburgh, PA 15213 USA, (e-mail: talukdar@cmu.edu).

Our goal, based on the conjecture that cascading failures are inevitable, is to provide grids with reflexes that minimize the social cost of these failures, and the extents of the resulting blackouts.

A. Approach

Our approach is to design a network of autonomous, software agents, each at a different node of the electric grid. Each of these agents works with locally available information (whatever it is able to collect from its own sensors and neighboring agents), and controls only a few local variables (such as the amount of load to be shed at its node). Therefore, each agent can react much faster than a centralized controller, which would have to collect information from the entire grid, and decide on the values of all the control variables. We believe that the additional speed made possible by a network of distributed agents is critical to the control of cascading failures.

Besides being much faster, networks of autonomous agents are also more robust and open than centralized agents. But distributed agents are not without disadvantages: they can be uncoordinated and parochial. To the extent that each agent is autonomous, it can do what it wants, and therefore, can work at cross-purposes to the other agents. Because each agent works with less than complete information, it can, at best, make locally correct decisions—which can be globally wrong.

B. Causes of Cascading Failures

Cascading failures occur because the grid's existing automatic control system is unable to make good tradeoffs among certain conflicting objectives.

Two of the many objectives in operating a power grid are: protect equipment from damage, and keep equipment in service. Overloads bring these two objectives into conflict. The control system (specifically, the protection subsystem) is incapable of finding good tradeoffs to resolve this conflict. Instead, whenever the overloads last long enough to endanger equipment, the protection subsystem removes the threatened equipment from service. When these removals produce further overloads, a cascade results.

C. Decomposition and Model Predictive Control

The problem we consider here is to minimize the cost of eliminating overloads (more precisely, operating-constraint violations) before they endanger equipment or trigger the protection subsystem. This overall problem is decomposed into sub-problems, one for each agent. The agents work on their sub-problems in parallel, using whatever locally available information they are able to collect. Of course to make globally optimal, or even globally feasible, decisions, each agent needs much more information—specifically, the state of the entire grid and what every other agent is going to do. To compensate for the missing information, we adapt MPC (model predictive control) to distributed agents.

Model predictive control (MPC) is a repetitive procedure that combines the advantages of long-term planning (feed-forward control based on performance predictions over an extended horizon) with the advantages of reactive control (feedback using measurements of actual performance). At the beginning of each repetition, the state of the system to be controlled is measured. A time-horizon, stretching into the future, is divided into intervals. Models are adopted to predict the effects of control actions on system-states in these intervals. The predictions are used to plan optimal actions for each interval. But only the actions for the first interval are implemented. When this interval ends, the procedure is repeated. Ref. [2] provides an overview of MPC theory and practice for centralized applications.

MPC, because of its use of optimization for making decisions, readily accommodates large numbers of complex constraints. Many other control techniques do not allow constraints. Instead, they require the designer to approximate the effects of constraints with conservative assumptions.

We adapt the MPC procedure for distributed agents by adding a second horizon in space (a horizon that stretches from each agent to the edges of the grid). Each agent is given its own suite of models that look ahead in time, and out into the grid. These models predict what the other agents will do, and how the network will respond. The models decrease in fidelity with distance from the agent at the present moment. The method that we use is related to, but not a reproduction of, the DMPC method developed by Camponogara et al. [3].

D. *Special Protection Schemes*

While this method is similar to many Special Protection Schemes (SPS), it differs from the traditional SPS in that the computation is located with the control hardware instead of at a central facility. Additionally, this method uses an optimization framework that adapts easily to arbitrary networks, and changing network conditions. Much has been written on the design of SPS. Typically SPS are designed by performing numerous network studies and pre-determining actions that tend to alleviate problems. Newer designs are able to adapt the rules to changing network conditions, but still rely on pre-determined rules [4]. Some SPS have been presented in the literature that make use of distributed agents, though using different designs than that presented here. Jung and Liu present a multi-agent method designed to avoid catastrophic power failures [5]. While their design uses agents for control, the agents are dependant on centralized facilities for planning activities. Designs also exist for augmenting standard protective relaying systems using agent technology [6], though such designs still allow violations to propagate through a network.

In what follows, we formulate the global problem of controlling the spread of cascading failures; decompose this problem into sub-problems, one for each agent; develop a DMPC (distributed MPC) procedure by which each agent can solve its sub-problem; and demonstrate that the solutions are close to being globally optimal.

II. Global Problem Definition

A. *Notation*

Let:

N	be the index set of all the nodes in the network.
n	be the index of the agent located at bus n.
Q	be the index set of all the branches in the network.
V	be a complex vector of node voltages. V_{nk} is the voltage at bus n at time step k.
I	be a complex vector of node current injections. I_n is the injection at bus n.
G	be a complex vector of generation power injections. For the sake of notational simplicity, we assume no more than one generator is located at each bus. It is fairly easy to incorporate multiple generators, but doing so complicates the notation somewhat.
L	be a complex vector of load powers. As above, we assume one load at each bus.
Y_{NN}	be the complex node admittance matrix for all the nodes in the network.
Y_Q	be the complex branch admittance matrix for the set of all branches in the network.
y_{nm}	be the single element of the node admittance matrix that is the admittance between buses n and m.

B. *Problem formulation*

As discussed above disturbances, such as short circuits and sudden generator outages, often cause violations of the network's operating constraints. If these violations persist in a network, relays operate or equipment fails, causing additional outages. If a set of violations can be eliminated before dependant outages occur, a cascading failure will not result.

With this in mind we propose to use the following control problem as a means of preventing cascading failures:

eliminate network violations before subsequent failures occur. For the sake of this paper, we consider this to be globally correct behavior. This problem can be formulated as a standard non-linear programming problem, using the steady state power network equations that would ordinarily be used in an optimal power flow formulation. Since many cascading failures are propagated by under/over-voltage conditions at buses and over-current conditions on transmission lines, we include these values as violations in our formulation. The decision space is any combination of load and generation shedding. We assume that both load and generation can be shed in continuous quantities. The objective function is the social cost of all control actions. Therefore, the global, single period control problem (P) is stated formally as follows (1a-1h).

$$\underset{G,L}{\text{minimize}} \sum_{n\in N} Cost_n\left(G_n - G_{n0}, L_n - L_{n0}\right) \qquad (1a)$$

subject to:

$$I = Y_{NN}V \qquad (1b)$$

$$G_n - L_n = V_n conj(I_n),\; n\in N \qquad (1c)$$

$$\mathrm{Re}\left(L_n/L_{n0}\right) = \mathrm{Im}\left(L_n/L_{n0}\right),\; n\in N \qquad (1d)$$

$$G_n^{\min} \le G_n \le G_n^{\max},\; n\in N \qquad (1e)$$

$$0 \le L_n \le L_{n0},\; n\in N \qquad (1f)$$

$$|V|^{\min} \le |V| \le |V|^{\max} \qquad (1g)$$

$$|I_{nm}| = |y_{nm}(V_n - V_m)| \le |I_{nm}|^{\max},\; n,m\in N, n\ne m \qquad (1h)$$

The costs associated with shedding load (from 1a) are the social costs that would be incurred from the interruption of electrical service. It may be that some loads are deemed more important to the system than others. This property can be incorporated into (1a) without difficulty. The costs associated with reducing generation come from either the potential equipment damage that rapid deceleration could cause, or the amount that would have to be paid to an independent power producer for such emergency control. Equality constraint (1b) defines the voltage-current relationships in the network. Equality constraint (1c) expresses conservation of energy at each node. Equality constraint (1d) forces the system to shed real and reactive load in equal proportions. Inequality constraints (1e) and (1f) describe the extent to which loads and generation can be adjusted. The final inequality constraints (1g and 1h) define the measures used to identify violations (bus voltage, and line currents), as well as the limits on those measures. This formulation can be extended to include constraints on the dynamic system, such as frequency or generator "out-of-phase" limits, but such extensions are beyond the scope of this paper.

Simulations using IEEE test networks indicate that power system violations can be eliminated by solving this problem and implementing the resulting control actions. As asserted in the introduction, we do not presume to be able to eliminate all cascading failures using this method. This method will not likely do much to control high speed (<1 second) cascading failures that result primarily from machine dynamics. Most cascading failures, however, are not of this type and progress over periods of seconds to minutes. While the solutions space of P is not necessarily convex, we have found that standard non-linear solvers reliably find good solutions to this problem for small networks (<200 buses). Fig. 1 shows the result of one such calculation using the IEEE 39 bus test case.

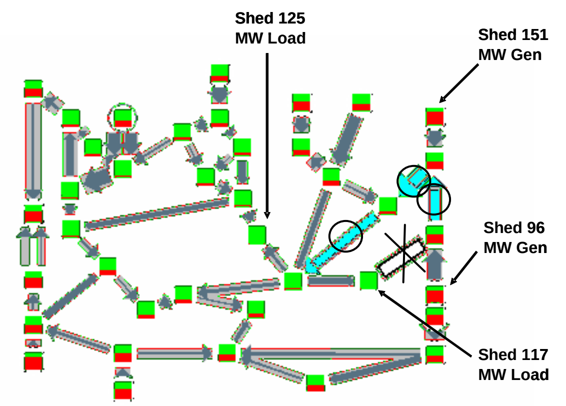

Fig. 1—Optimal load and generation shedding actions resulting from the solution of the global problem P on the IEEE 39 bus test case. These are the minimum cost actions that eliminate the violations shown. Branch current violations are marked with circles. These violations occurred after a line outage was applied at the location marked with an X. The arrows indicate power flow magnitude and direction.

III. Problem Decomposition

For large-scale power systems, it is impractical or even impossible to solve the global control problem in a centralized fashion. Doing so would require more communication and computation than current technology can provide. Additionally, doing so would require that a single control center communicate constantly with every bus in a given synchronous network. For the U.S. Eastern Interconnect, this would mean that a single control center would need to manage communication links and data from about 50,000 buses. This space is currently managed by hundreds of relatively autonomous system operators who are often reluctant to share data and control. This level of centralized control is institutionally impractical in all but a few locations. Thus a decentralized solution is necessary.

Given this necessity, our problem is to take the global control problem defined in section III and decompose it into tractable sub problems that, when solved and implemented by distributed agents, result in the desired global behavior. This is the general challenge for all distributed control problems: to design methods by which locally correct decisions result in globally correct decisions.

Our decomposition method can be described as follows:

- Place a software agent at each load and generation bus
- Allow each agent to gather measurements from the system through communication networks
- Allow each agent to control its local control variables
- Give the global control problem to each agent
- Allow each agent to reduce its problem into tractable, local versions of the global problem
- Allow each agent to solve its sub-problem iteratively using MPC, cooperation, and learning.

The result is a two-dimensional decomposition of the global control problem. Firstly, the problem is decomposed in space by assigning the problem to distributed agents with local control abilities. Secondly the problem is decomposed in time by allowing the agents to act iteratively using MPC, harnessing feedback control to improve local solution quality. A high level view of this feedback loop and how it interacts with operator control is illustrated in Fig. 2.

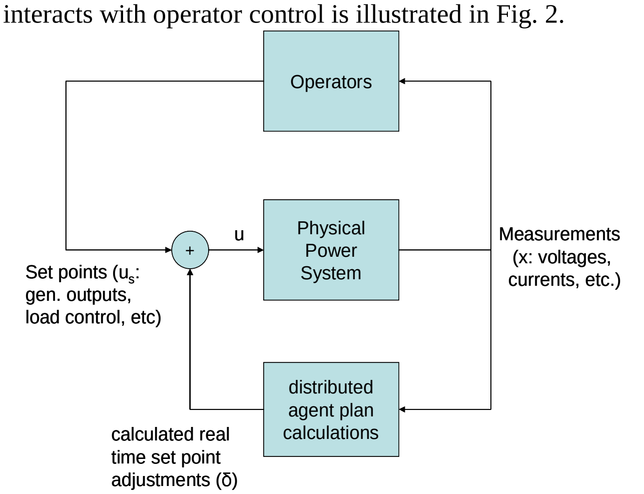

Fig. 2—Feedback diagram of the system showing how operators and the agent network interact. Under normal circumstances agents do not make adjustments to operator set points, but if a violation occurs, agents work together to eliminate the violations by adjusting operator calculated set points.

A. *Additional Notation*

Let:

M	be the index set of all nodes (or control variables) that agent *n* includes in it sub-problem. $M \subset N$
R	be the index set of all branches that agent *n* includes in its local problem. $R \subset Q$
u	be a vector of control variables: $u = \begin{bmatrix} \mathrm{Re}(G) \\ \mathrm{Re}(L) \end{bmatrix}$.
δ	be a vector of control variables changes: $\delta = u - u_0$.
u_G , u_L	be subsets of the control vector for generation or load.
c	be a vector of costs associated with control variable reductions.
D	be a matrix of branch current sensitivity factors such that $\lvert I_{Qk} \rvert \cong \lvert I_{Q,k-1} \rvert + D\delta_k$.
E	be a matrix of bus voltage sensitivity factors such that $\lvert V_{N,k} \rvert \cong \lvert V_{N,k-1} \rvert + E\delta_k$.
K	be the final time step in the control time horizon. Agents seek to eliminate violations completely by this time period.
ρ	be the discount factor used in the MPC cost function.

B. *Spatial decomposition*

Since it is infeasible for every agent to maintain a perfect model of the entire network, each agent must selectively collect enough measurements to get good results for its local control variables. In general we have found that an agent's sensitivity (the degree to which the error affects an agent's local decisions) to remote measurement errors decreases as the errors are located more remotely. Therefore we task each agent with collecting data frequently from its immediate neighborhood, and rarely or never from distant portions of the network. We assign each agent with a sub-network within which the agent collects data frequently (several times per second). All of the nodes that can be reached by traveling over no more than *r* branches are inside an agent's sub-network, where *r* is the radius of the sub-network. All other nodes are in an agent's external network. Fig. 3 illustrates spatial decomposition using the IEEE 118 bus network.

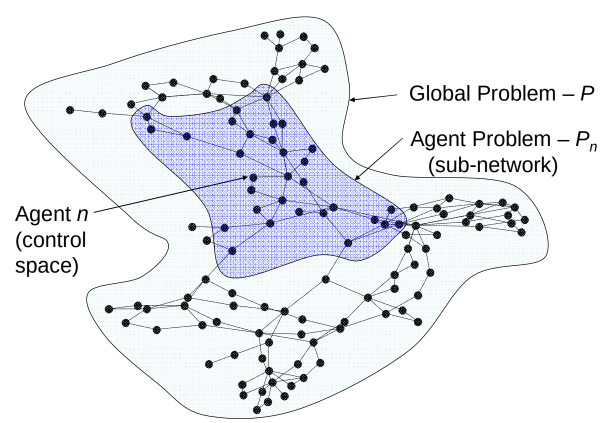

Fig. 3—Illustration of the spatial decomposition of the global problem (*P*) on the IEEE 118 bus network.

As previously mentioned, solving the full non-linear control problem given computational constraints is infeasible for a network of practical size. Therefore an agent must reduce the global problem into a tractable version that can be solved given computational and time constraints. The agents do this using a linear-difference version of the global problem. This formulation is presented below (2a-2d).

$$\underset{\delta_M}{\text{minimize}} - c_M{}^T \delta_M \quad (2a)$$

subject to:

$$\sum_{g \in G_M} \delta_g = \sum_{l \in L_M} \delta_l \quad (2b)$$

$$|I_R| = |I_{R0}| + D_{RM}\delta_M \le |I_R|_{\max} \quad (2c)$$

$$|V_M|_{\min} \le |V_{M0}| + E_{MM}\delta_M \le |V_M|_{\max} \quad (2d)$$

$$-u_{M0} \le \delta_M \le 0 \quad (2e)$$

In this simplification of (P), we reduce all of the non-linear equality constraints into a simple balancing constraint (2b) that forces the system to choose to shed load and generation in equal quantities. While this does not guarantee that the system will maintain perfect load balancing, the small errors should be easily picked up by existing frequency regulation mechanisms. If a significant frequency imbalance were detected, (2b) could be weighted to compensate for the imbalance.

This formulation has several distinct advantages over the full non-linear problem (P). Constraints that are distant or unimportant and remote variables that will not affect the final solution can be easily removed from the local problem. The difference nature of (2) allows for state measurements to be incorporated into the problem, making iterative solutions feasible. Finally, compared to (1), (2) is noticeably less sensitive to errors in remote control variable estimates. This stems from two aspects of the formulation. Firstly, it uses the DC load flow approximations to calculate the load distribution factor matrix (D) as follows:

$$D = \mathrm{Im}(Y_{QN})\,\mathrm{Im}(Y_{NN}{}^{-1})\Lambda \quad (3)$$

where Λ is a matrix that translates the control variables into bus power injections:

$$\Lambda : \; G - L = \Lambda \begin{bmatrix} G \\ L \end{bmatrix} \quad (4)$$

This allows the agent to calculate D without any knowledge of the network state (voltage, current) and control (generation, load) variables. An agent must only know the status of the branches in the network to obtain the matrix. Even if some status errors exist in the agent's model of distant parts of the network, the elements of D will be nearly correct. Secondly, the agent only needs to know the amount of load and generation at a particular location if it decides that the entire quantity should be eliminated from the network. Since this case occurs relatively rarely, large errors are infrequent. The disadvantage is that solutions to (2) will be sub-optimal solutions to (1), even given perfect information. This sub-optimality is acceptable since we are primarily concerned with eliminating violations. Doing so at minimum cost is secondary.

C. *Temporal decomposition*

It is not difficult to adapt the linear difference formulation (2) to make use of MPC. We add a time dimension to the control vector (δ), giving us a decision matrix (Δ)—a two dimensional control plan. The cost function therefore is a summation of control costs over the time horizon. The costs are discounted so that the least expensive actions will be chosen first, and more expensive actions later. The solution should be independent of the discount rate chosen—any discount rate between zero and one should give the same result. Additionally, we add some slack to the constraints so that the violations need not be entirely eliminated during the first period, but can be gradually eliminated over time. Finally, we add ramp rate constraints on the generators since there are naturally limits to how fast a generator can decelerate. Thus, the DMPC problem for agent n at time k_0 (P_{n,k_0}) can be written:

$$\underset{\Delta_{MK}}{\text{minimize}} \sum_{k=k_0}^{K} e^{-\rho k} c_M{}^T \Delta_{Mk} \quad (5a)$$

subject to (for $k=k_0 \ldots K$):

$$\sum_{g \in G_M} \delta_{gk} = \sum_{l \in L_M} \delta_{lk} \quad (5b)$$

$$|I_{Rk}| \cong |I_{Rk-1}| + D_{RM}\delta_{Mk} \le f_1(I_{R,0\ldots k-1})|I_R|^{\max} \quad (5c)$$

$$f_2(V_{M,0\ldots k})|V_M|_{\min} \le |V_{Mk}| \cong |V_{M,k-1}| + E_{MM}\delta_M \le f_3(V_{M,0\ldots k})|V_M|_{\max} \quad (5d)$$

$$RR_g \le \delta_{gk} \le 0, \; g \in G_M \quad (5e)$$

$$-u_{l0} \le \delta_{lk} \le 0, \; l \in L_M \quad (5f)$$

$$-u_{M0} \le \sum_{k=k_0}^{K} \Delta_{Mk} \le 0 \quad (5g)$$

System operators should be able to estimate how quickly a violation must be eliminated to prevent relay operation. This will depend on both the magnitude of the violation, as well as the time that the violation has remained on the system. To prevent a zone three or time over-current operation, the violation will need to be eliminated fairly quickly (1-2 seconds). In order to minimize the risk of a line sagging and causing a fault, longer time delays (seconds to minutes) will likely be acceptable. This relationship between time and violation magnitude is encoded into the formulation in the constraint multiplier functions f_1, f_2, and f_3 in (5c, 5d). In this paper we use four period simulations and define f such that agents will seek to reduce violations to 130% of the rating in the first period, 120% in the second, 110% in the third, and 100% in the last period. If the violation persists past the original planning horizon the agent continues to act to reduce the violation below the threshold. The result of each calculation is a control plan Δ_{WK}*. This plan is illustrated in fig. 4.

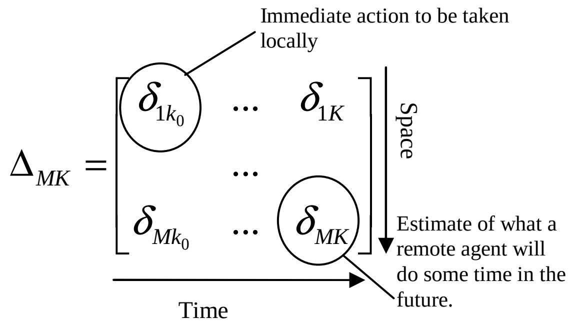

Fig. 4—Diagram of the 2 dimensional decision plan calculated by agent *n*.

D. *Implementation*

Once an agent has calculated its control plan it takes the local portion of this plan (the immediate action to be taken locally), implements it locally, takes some additional measurements, advances the control horizon, calculates a new plan, and acts again. This process differs from standard MPC in that the time horizon is reduced after each control action, and that the time horizon is finite. In traditional MPC, an infinite (or at least large), constantly advancing control horizon is used.

IV. Experimental Results

In this section we describe the results of numerous simulations designed to evaluate this method using standard IEEE test networks.

A. *Simulation model description*

For the simulations that follow we use a standard, non-linear, power flow network model with constant real and reactive power loads and constant power, voltage generators. The network is assumed to perform frequency regulation through a single slack bus (standard power flow assumption). We assign load shedding costs of \$1000/MW to all loads in the system. We assume that the cost of reducing generator output is constant for all generators at \$30/MW. All agents use a four period (K=4) solution horizon, but continue to act using a single period (K=1) formulation if one or more violations persist after the initial control horizon. For each simulation we choose a communication neighborhood radius, r. All of the buses that can be reached by traveling over no more than r branches are within an agent's neighbor set (M). Within this radius we assume that the agent is able to obtain good (zero error) measurements constantly. An agent obtains no real-time measurements from the buses outside of its neighborhood, but can estimate the quantity of load and generation at these buses in these areas if needed. We model this estimation process using Gaussian noise with 15% coefficient of variation (CV). Each simulation begins with a violation inducing disturbance. The simulations in this paper are only for branch-outage disturbances, though future work will include generator outages as well.

Therefore, the simulation process proceeds as follows:

1. Choose an IEEE network
2. Increase the loading so that the network is somewhat stressed.
3. Run an optimal power flow to obtain the initial network conditions.
4. Allow the agents to take noisy measurements from the system such that they have estimates of the network control variables.
5. Inject a disturbance (one or more branch outages) that causes at least one violation.
6. Run a power flow to obtain the modified network state.
7. Set k=0, K=4.
8. Allow the agents to take measurements from their local neighborhoods.
9. Allow the agents to calculate control plans for the control horizon (k+1…K), and implement the portion for period k+1.
10. Increment k (and K if $k+1>K$).
11. Recalculate the power flow.
12. Repeat from 8 until all of the violations are eliminated, or until it is clear that the agents will not be able to eliminate the remaining violations.

B. *Communication requirement computation*

The following experiment was designed to determine how much communication is required to obtain good results from this method. First, we chose a set of 15 disturbances that cause severe branch current violations on a heavily loaded IEEE 118 bus network. The disturbances were chosen randomly, without regard for the system's ability to control the resulting violations. Second, we chose a neighbor radius (r) between 2 and 10 (the diameter of the IEEE 118 bus network is 20). We then simulated the system for each disturbance-radius combination. Fig. 5 shows the control goal for line current violations and the set of violation trajectories resulting from the simulations with r = 5.

The results from these simulations demonstrate the relationship between network radius and solution error. We define control error for a single time step to be the amount by which the worst violation exceeds the control goal. Therefore, the total control error is the sum of the errors over the simulation time horizon. Fig. 6 shows this relationship between communication and control error.

While few of the solutions resulted in zero control error, the solutions for agent networks with $r \geq 7$ have small solution errors even in the worst case. The maximum value shown in fig. 6 was a single outlier case. Some additional cooperation among the agents could vastly reduce the error in this case. Fig. 7 shows the relationship between the number of time steps required to effectively eliminate all violations and the amount of communication.

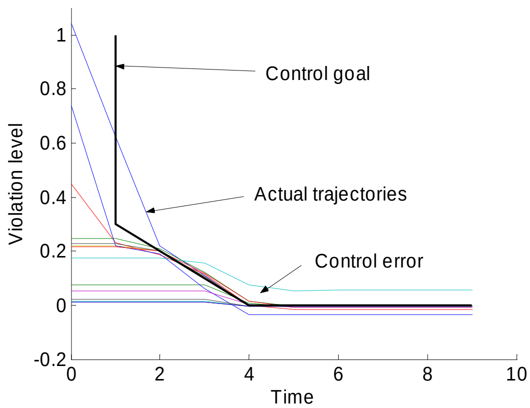

Fig. 5—Control goal and actual violation trajectories for the set of 15 simulated disturbance with an *r*=5 communication radius.

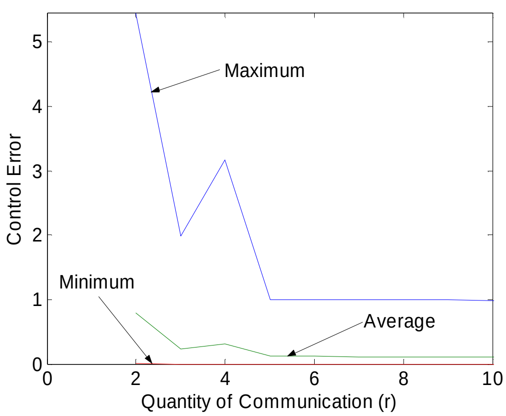

Fig. 6—Plot of control error vs. quantity of communication for the 15 disturbances simulated. Control error is measured by calculating the integral over time of the maximum violation. Communication is in terms of the radius of an agent's neighbor set.

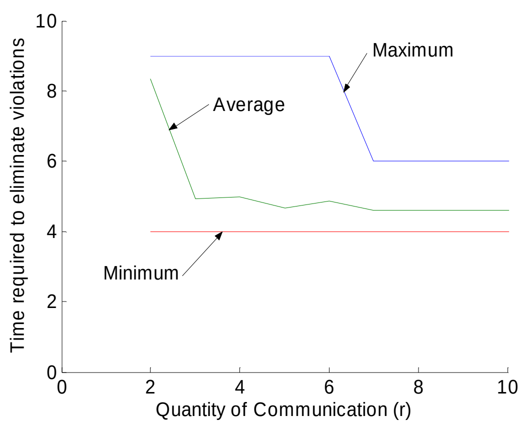

Fig. 7—Plot showing the relationship between the quantity of agent communication and the time required to eliminate violations. 4 is the minimum value since the control goal is to eliminate the violations in 4 control steps.

V. SUMMARY

In this paper we show that it is possible to design a network of autonomous agents, one at each node of a power network, that work locally to eliminate global network violations. Initial simulations indicate that this method can meet its objective without global knowledge. Implementing such a network should make the transmission network more robust to unexpected disturbances. Doing so should reduce the risk of cascading failures, and has the potential for increasing the network's ability to use existing transmission capacity.

Several outstanding issues and questions remain. This method can be improved substantially by adding cooperation and learning to the agent capabilities. Additionally we hope to characterize the costs, benefits and risks of this method in more detail as a means of studying what barriers exist to its adoption. One barrier to adoption is that this method requires implementation in a fairly large portion of the network in order to effectively control cascading failures. Such large-scale adoption is unlikely without regulatory interventions that provide incentives for transmission system operators to upgrade the network.

REFERENCES

[1] S. N. Talukdar, J. Apt, M. Ilic, L. B. Lave, and M. G. Morgan, "Cascading failures: Survival versus Prevention," *The Electricity Journal*, Nov. 2003.

[2] J. B. Rawlings, "Tutorial Overview of Model Predictive Control," *IEEE Control Systems Magazine*, vol. 20, no. 3. pp. 38-52, 2000.

[3] E. Camponogara, D. Jia, D., B.H. Krogh and S. Talukdar, "Distributed Model Predictive Control," *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 44-52, 2002.

[4] V. Madani, M. Adamiak, and M. Thakur, "Design and Implementation of Wide Area Special Protection Schemes," in *Proc. 57th Annual Texas A&M University Conference for Protective Relay Engineers*, 2004.

[5] J. Jung and C. Liu, "Multi-agent technology for vulnerability assessment and control," *Proc. Power Engineering Society Summer Meeting, 2001*, July 2000.

[6] C. Yanxia, Y. Xianggen, Z. Zhe, C. Deshu, Z. Xiangjun, "Resarch on Protective Relaying Systems Based on Multi-Agent Systems," *Proc. PowerCon 2002,* Oct. 2002.