Reciprocally Altruistic Agents for the Mitigation of Cascading Failures in Electrical Power Networks

Paul Hines, *Member, IEEE*, Sarosh Talukdar, *Fellow, IEEE*

***Abstract*—Cascading failures in electrical power networks often come with disastrous consequences. A variety of schemes for mitigating cascading failures exist, but the vast majority depend upon centralized control architectures. Centralized designs are frequently more susceptible to communications latency and bandwidth limitations and can be vulnerable to random failures and directed attacks. This paper proposes a decentralized approach. We place control agents at each substation in a power network, each of which uses decentralized model predictive control to select emergency control actions. When making decisions the control agents consider not only their own goals, but also the goals of nearby agents. Thus the agents act with reciprocal altruism. Results from simulations of extreme cascading failures within the IEEE 300 bus test network indicate that this approach can dramatically reduce the average size and social cost of large cascading failures. Simulations also show that the bandwidth required for message passing is well within the limits of current technology.**

***Index Terms*—multi-agent systems, model predictive control, reciprocal altruism, cascading failures, power system blackouts**

I. Introduction

CASCADING failures are found in many, if not all, large network systems. Small failures in or attacks on financial markets, the Internet, commercial air-traffic systems and electrical power grids occasionally initiate cascades of dependant failures. The results are often tremendously costly. As did the large North American blackouts of 1965 and 1977, the cascading failures in the North American (August 14) and Southern European (September 28) grids in 2003 spurred a wave of new interest into the mechanics of [6], [5] and mitigation methods for cascading failures in power networks. Because of their complex discrete and continuous dynamics, cascading failures in power networks provide a challenging test case for mitigation strategies that could be useful in other infrastructure systems.

Power networks, like most infrastructure systems, are operated by both decentralized and centralized controllers. The decentralized controllers (relays for example) work with local data to quickly make local decisions. The centralized controllers (human operators for example) work with larger data sets to make decisions along longer time horizons according to global goals. Sometimes local controllers with limited information do not, or cannot, make decisions that align well with the global goals of the system. Distance relays that can propagate a cascading failure in a power network provide a good example of this behavior. The goal of this work is to develop decentralized control agents that gather information from their neighbors to make real-time decisions that are good with respect to the system as a whole. Such "reciprocally altruistic" agents would bridge the gap between high-speed, low-intelligence decentralized control and low-speed, high-intelligence centralized control.

Paul Hines is with the School of Engineering, University of Vermont, Burlington, VT, e-mail: paul.hines@uvm.edu.

Sarosh Talukdar is with the Electrical and Computer Engineering Dept., Carnegie Mellon U., Pittsburgh, PA, e-mail: talukdar@ece.cmu.edu.

This work was supported in part by ABB Corporate Research in Raleigh, NC.

A. Centralized and decentralized cascading failure mitigation

Numerous centralized methods for cascading failure mitigation exist. Wide Area Control Systems (WACS) or Special Protection Schemes (SPS) have been in development for many years, with limited adoption and some success in the electricity industry [1], [23]. When well designed, a SPS can provide a power network with automated stress responses that can arrest the spread of many cascading failures. SPS can enable a network to operate with smaller margins [1], [13]. In some cases the predetermined control actions that are common to many SPS can act to increase the size of a blackout, rather than decreasing its size. More recent WACS employ adaptive concepts, including an algorithm that uses correlation among generator phase angles to divide a grid into islands [22] and optimization-based methods such as Model Predictive Control [11], [9], [18], [16].

Few of the adaptive, centralized approaches have been widely adopted. One arguable explanation for this is that infrastructure networks in general, and power systems in particular, are inherently decentralized systems. When speed is important, as it is during a cascading failure, it is useful for the decision makers to be co-located with the sensors and actuators. This is largely a function of communications latency, which has been shown to negatively affect some WACS [17]. Some emerging designs use decentralized controllers as a part of the overall design. The Strategic Power Infrastructure Defense system described in [12] employs a hierarchical, multi-agent architecture. A multi-agent approach was used in [15] to coordinate the actions of generator controllers to damp inter-area oscillations. Decentralized control agents were used in [7] to reroute power flows in a DC shipboard power system by adapting the max flow problem from graph theory to a decentralized formulation of the power flow problem.

The method proposed in this paper differs from the above decentralized algorithms in several ways. Firstly, it can manage transmission-line overloads, which are particularly important for cascading failure propagation. Secondly, it does not require any real-time interaction with centrally located operators or blackboards. Thirdly, the method is sufficiently general that it could have application in other problem domains.

B. A formulation for emergency control in power networks

A good objective for power system operations is to maximize the difference between the benefit of service provided (electrical energy) and the cost of providing that service (fuel cost, maintenance costs, etc.). In the economics literature this is known as social welfare maximization, a form of which is used in many Optimal Power Flow formulations [20]. When a power system is stressed, where a cascading failure is likely or is already in progress, fuel cost is of lesser importance relative to the losses associated with interrupted service and damaged equipment. Thus during stressed operations ("Alert" or "Emergency" stages from [8]), a good operating policy is to simultaneously minimize the costs associated with mitigating control actions (wear and tear on the equipment and interrupted service) and the risk of interrupted service that could result from future dependant failures. Eqs. 1-3 provide a formal description of this problem, which is here referred to as the Emergency Control Problem (ECP).

Given a discrete time horizon starting with the current time (t_0) and ending at some point in the not too distant future (t_K), a state matrix: $\mathbf{X} = [\mathbf{x}_0, .., \mathbf{x}_k, ...\mathbf{x}_K]$, a control variable set point matrix: $\mathbf{U} = [\mathbf{u}_0, ..., \mathbf{u}_k, ..., \mathbf{u}_K]$, and a state predictor function $\mathbf{g}(...)$, we can formulate the ECP as:

$$\min_{\mathbf{U}} \quad c(\mathbf{U}, \mathbf{X}) + r(\mathbf{U}, \mathbf{X}) \tag{1}$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{u}_{k+1}) \tag{2}$$

$$\underline{\mathbf{u}}(\mathbf{u}_k) \leq \mathbf{u}_{k+1} \leq \overline{\mathbf{u}}(\mathbf{u}_k) \tag{3}$$

where c and r are estimates of the cost and risk associated with a trajectory of control and state variables. Assuming (1) that the system is deterministic within the time horizon and accurately modeled by $\mathbf{g}$, (2) that r provides an accurate measure of the expected losses after t_K (the end of the time horizon), (3) that the problem is convex, and (4) that the time steps are small enough to ensure that all of the useful information about the trajectory is represented by the discrete signals, an optimal solution to ECP ($\mathbf{U}_*$) will be at least as good as a control signal produced by any other controller. When there is uncertainty (random variables) in a system, $\mathbf{g}$ will not provide perfect next state predictions. There are two approaches for dealing with uncertainty in an optimization-based control problem. The first approach is to reformulate the problem using stochastic programming, explicitly modeling the random variables from estimated probability distributions. This approach can improve the quality of the output signal, particularly when the variables do not follow Gaussian distributions, but substantially increases computational complexity. Another approach is to use feedback and small time steps to compensate for modeling errors and uncertainty. While a combination of these two methods may be considered in future work, this paper takes the second approach (feedback) to uncertainty. The result is a Model Predictive Control (MPC) problem that can be used for centralized cascading failure mitigation given good real-time system observability and communications bandwidth at the operator's facility.

While it may be feasible to solve the ECP from a central location (Section 2 describes the centralized approach), a decentralized solution could have a number of advantages, particularly in terms of increased robustness and decreased communications bandwidth. With this in mind this paper (in Sections 3 and 4) describes a way to use reciprocally altruistic software agents to solve the ECP in real time. Section 5 describes experimental results that indicate the effectiveness of this decentralized approach to emergency control and Section 6 provides some conclusions.

II. Centralized MPC for Cascading Failure Mitigation

This section describes how the general ECP formulation (Eqs. 1-3) can be used to select load-shedding and generator adjustments to quickly mitigate the effects of a cascading failure from a central location. The problem will use as its control variables ($\mathbf{u}_k$): generator power output set points ($\mathbf{P}_G$) generator voltage magnitude set points ($|\mathbf{V}_G|$) and load reduction. We assume that load can be reduced continuously through carefully selected switching actions and that buses are defined such that generators and loads are at different buses. Demand reduction is represented by a reduction factor $\lambda_d \in [0, 1]$ for each bus $d \in D$, such that the injection by loads at bus d is $P_d + jQ_d = \lambda(\overline{P_d} + j\overline{Q_d})$. Here $\overline{P_d} + j\overline{Q_d}$ represents the demand at bus d before demand reduction.

With these definitions and using the Δ operator to represent differences in discrete time ($\Delta x_k = x_{k+1} - x_k$) the cost function ($c$ in Eq. 1) can represent the costs associated with changes to the control variables:

$$c(\mathbf{X}, \mathbf{U}) = \tag{4}$$
$$\sum_{k=1}^{K} \rho^k \left(\mathbf{c}_D^T \Delta \lambda_k \quad + \quad \mathbf{c}_G^T \Delta \mathbf{P}_{G,k} + \mathbf{c}_{\Delta V}^T \left| \Delta |\mathbf{V}_{G,k}| \right| \right)$$

The three cost vectors ($\mathbf{c}_D$, $\mathbf{c}_G$, $\mathbf{c}_V$) represent demand reduction costs ($\mathbf{c}_D$), fast generator ramping costs ($\mathbf{c}_G$) and costs associated with rapid changes to voltage set points ($\mathbf{c}_G$). We assume that these costs can be determined and set off-line by operators or regulators. By using a separate cost term for each control variable one can tune the system to give preference toward load shedding at less critical locations and thus reduce the probability of interrupting critical loads. The discount factor, ρ^k, places higher importance on actions in the near future. Discount factors are useful in planning problems, such as an MPC formulation, because uncertainty increases with time.

The risk term (r in Eq. 1) estimates the impact of stress on the risk associated with future cascading failures. The risk function described below is simple, but has proved useful in practice. It assigns a cost to excessively high branch current magnitudes and low voltages over the time horizon. When all voltage and current magnitudes are within their limits over the time horizon, r evaluates to zero. When currents are high or voltages are low, r is proportional to the excess:

$$\begin{aligned} r(\mathbf{X}, \mathbf{U}) \quad = \quad & \sum_{k=1}^{K} \rho^k \left(\mathbf{c}_I^T \max\left(|\mathbf{I}_k| - \beta_I(k) \odot \overline{|\mathbf{I}|}, \mathbf{0} \right) + \right. \\ & \left. \mathbf{c}_V^T \max\left(|\mathbf{V}_k| - \beta_V(k) \odot \underline{|\mathbf{V}|}, \mathbf{0} \right) \right) \end{aligned} \tag{5}$$

where $\mathbf{x} \odot \mathbf{y}$ represents the element-by-element product of $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{c}_{\overline{I}}$ and $\mathbf{c}_{\underline{V}}$ are cost vectors for excess currents and low voltages, $\overline{|\mathbf{I}|}$ and $\underline{|\mathbf{V}|}$ are vectors of current and voltage limits, and $\beta_I(k) > 1$ and $\beta_V < 1$ are scalars that progressively move the effective current and voltage stress thresholds ($\beta_I(k)\overline{|\mathbf{I}|}$ and $\beta_V(k)\underline{|\mathbf{V}|}$) toward the actual limits over the time horizon. In the examples used in this paper $\beta(k)$ are chosen to force the effective voltage and current limits toward the actual limits linearly with a fixed reduction rate. For example, if the current on branch i is $|I_i| = 150$A, the current limit is $\overline{|I_i|} = 100$A, and the reduction rate is set at 10% ($\alpha_I = 0.1$), $\beta_i(k)$ will be the sequence: $\beta_i(k) \in \{1.4, 1.3, 1.2, 1.1, 1.0\}$.

The state-predictor function, $\mathbf{g}(...)$ provides a mapping between state and control variables and system state at next time step. For this application we use a linear state predictor function with the following form:

$$\mathbf{A}\mathbf{x}_{k+1} = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}\Delta\mathbf{u}_k \qquad (6)$$

which becomes the direct function $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{A}^{-1}\mathbf{B}\Delta\mathbf{u}_k$. When $\mathbf{A}$ and $\mathbf{B}$ are built to form a linearized version of the AC power-flow constraints ($\mathbf{P} + j\mathbf{Q} = \mathbf{V} \odot (\mathbf{Y}\mathbf{V})^*$), Eq. 6 is a fairly accurate predictor so long as the changes at each MPC time steps are small.

The control variable limits (Eq. 3) limit the control variable set points over the time horizon ($\mathbf{0} \leq \lambda_k \leq \mathbf{1}$, $\underline{\mathbf{P}_G} \leq \mathbf{P}_{G,k} \leq \overline{\mathbf{P}_G}$, and $\underline{|\mathbf{V}_G|} \leq |\mathbf{V}_{G,k}| \leq \overline{|\mathbf{V}_G|}$) and changes to the set points, as shown in Eqs. 7-9.

$$-\mathbf{P}_{G,0} \leq \quad \Delta\mathbf{P}_{G,k} \quad \leq \mathbf{0} \qquad (7)$$

$$-\overline{\Delta|V_G|} \leq \quad \Delta|\mathbf{V}_{G,k}| \quad \leq \overline{\Delta|V_G|} \qquad (8)$$

$$-\mathbf{1} \leq \quad \Delta\lambda_k \quad \leq \mathbf{0} \qquad (9)$$

The voltage change limit $\overline{\Delta|V_G|}$ is chosen to force generators to change their set points incrementally. Table I shows typical values for the parameters used in the EPC formulation.

Table I
TYPICAL PARAMETERS FOR THE GLOBAL ECP FORMULATION

Parameter	Typical range	Value used for Fig. 1	Units
$\overline{\Delta\lvert V_G\rvert}$	$[0, 0.01]$	0.01	p.u. V
ρ	$(0, 1]$	0.09	–
c_I	$[10^6, 10^{10}]$	10^8	\$/p.u. A
c_V	$[10^6, 10^{10}]$	10^8	\$/p.u. V
c_{θ_s}	$[10^{10}, 10^{14}]$	10^{12}	\$/radian
α_I	$[0.05, 0.20]$	0.10	fraction of $\lvert I\rvert$ limit
α_V	$[0.001, 0.02]$	0.01	p.u. V
$\mathbf{c}_D$	$[100, 10000]$	Randomly assigned	\$/MW
$c_{\Delta V}$	$[0, 10^5]$	\$10,000	\$/p.u. V
c_G	$[0, 100]$	30	\$/MW

A. Illustrative results

To demonstrate that the global ECP can be used to reduce the size of cascading failures, we simulate a global ECP controller interacting with a power-flow model of the IEEE 300 bus test case as available with MATPOWER [24]. To simulate cascading failures, random branch outages were injected into the network to find contingencies that cause over-current conditions. After injecting a contingency (removing branches), the system is simulated in time by a sequence of AC power flow calculations. At each time step a set of emergency control actions ($\Delta\mathbf{P}_G$, $\Delta|\mathbf{V}_G|$, and $\Delta\lambda$) are calculated from which loads and generators are adjusted. Figure 1 shows the ECP control actions, along with currents and voltages, for a seven branch contingency on the 300 bus network. If the overloads had been allowed to persist, the system would have cascaded to a near total blackout. By shedding 0.43% of demand (in MW) with the ECP controller a system-wide cascading failure was averted. Because the algorithm chooses load shedding based on cost-priorities substantially less of the total demand value was lost (0.035% of the total value in \$ vs. 0.43% of the load in MW). Also, experimental results show that the inclusion of generator bus voltage settings in the control variable set can significantly reduce the cost of mitigating control actions. Without voltage control the amount of load shedding required increases from 100 MW to 164 MW.

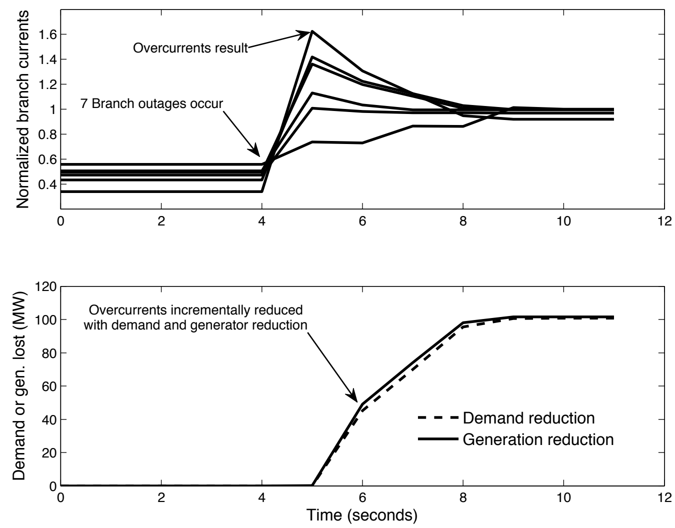

Figure 1. Time domain simulation results showing mitigating control actions after removing 7 branches from the IEEE 300 bus system. Branch currents are shown normalized by their limits: $|I|/\overline{|I|}$.

III. PROBLEM DECOMPOSITION, COOPERATION AND RECIPROCALLY ALTRUISTIC AGENTS

In resource-constrained systems there are many problems that are more effectively solved by groups of agents, rather than a single problem-solving agent. If a team of problem-solving agents is to arrive at a good solution, the problem must be decomposed into subproblems, each of which is in some way more tractable than the global problem. The process of dividing a large problem into sub-tasks is known as problem decomposition. There are many ways to decompose a large problem into subtasks. With some problems, decomposition is a fairly straightforward process: find portions of the problem that are loosely connected to other portions of the problem, assign the sub-problems to agents, and assemble the sub-problem solutions after the agents complete their work. Take for example the solution of a large Monte-Carlo simulation problem. Once a mathematical system model is established each random perturbation of the problem can be solved in parallel without interaction among the solver-agents. The distribution of outcomes is relatively easy to compute after the agents complete their calculations. However, many problems have tightly interconnected components, in which vastly superior solutions can be found when problem-solving agents share information and coordinate their actions while they are still working on their solutions. One approach to coordinated work is to decompose the problem into tasks that can be completed sequentially. An assembly line is an example of sequential decomposition. The Gauss-Seidel power flow solution method [14] is another. Series work can provide good results in many, but not all, problems (see [3] for a proof regarding the effectiveness of sequential work). When a problem has many sub-problems and tight time constraints, series work is not practical. For the case of large, time-critical network control problems such as the ECP in a power network, series work is impractical with realistic communications constraints. Such problems require a problem decomposition that facilitates high-speed, highly-coordinated, real-time decision making.

Because we are interested in problem decomposition methods that are practical for real-time problems (with time horizons in the seconds to minutes range), we will restrict our attention to problems that can be divided among n control agents that work in parallel and which are located at nodes in a network. Thus the feasible control space, $\mathbf{u}(t_k) \in \mathfrak{U}(t_k)$, and state space, $\mathbf{x}(t_k) \in \mathfrak{X}(t_k)$, can be divided disjointly among the agents such that each variable is assigned to exactly one agent. If $\mathbf{u}_{N_a}$, $\mathbf{u}_{N_b}$, $\mathbf{x}_{N_a}$, and $\mathbf{x}_{N_b}$ represent the control and state vectors for two arbitrarily chosen agents a and b, the subscript sets N_a and N_b have the following properties:

$$\begin{aligned}\bigcup_{i=1}^{n} N_i &= N : \mathbf{u}_{N,k} = \mathbf{u}_k \\ N_a \cap N_b &= \emptyset, \forall a, b \in \{1, \ldots, n\}\end{aligned}$$

After dividing the variables among agents in this way, assuming that the objective function can be separated into a sum of sub-objectives, the global problem can be written as follows:

$$\min_{\mathbf{u}} \quad f(\mathbf{u}, \mathbf{x}) = \sum_{i=1}^{n} f_i\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}\right) \tag{10}$$

$$\text{s.t.} \quad \mathbf{g}(\mathbf{u}, \mathbf{x}) = \begin{bmatrix} \mathbf{g}_{M_i}\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}\right) \\ \mathbf{g}_{\overline{M_i}}\left(\mathbf{u}_{\overline{N_i}}, \mathbf{x}_{\overline{N_i}}\right) \end{bmatrix} \leq \mathbf{0} \tag{11}$$

Here the constraints are divided such that each constraint that includes any of agent i's variables is included in the set M_i. The constraints that are not affected by agent i's local variables are in the set $\overline{M_i}$. Symbolically:

$$M_i = \{\forall a : \nabla_{\mathbf{u}_{N_i}, \mathbf{x}_{N_i}} g_a \neq \mathbf{0}\}$$

We assume that the control and state variables can be defined such that interactions occur only through state variables ($\mathbf{x}$). In this form it is fairly easy to divide the objective and constraints among agents. Each agent i is responsible for a sub-objective f_i and a set of constraints $\mathbf{g}_{M_i}$. Note that the sets M_i may overlap.

For most network problems there exists a fairly natural way to assign terms in a global objective function to locations (agents) in the network. In a power network, for example, we can assign the costs associated with loads and generators to agents located at substations from which these variables are controlled. With the global problem thus defined it is straightforward to divide the problem into a set of agent sub-problems of the form:

$$\min_{\mathbf{u}_{N_i}} \quad f_i(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}) \tag{12}$$

$$\text{s.t.} \quad \mathbf{g}_{M_i}\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}\right) \leq \mathbf{0} \tag{13}$$

One candidate decomposition of this problem would be to allow each agent to solve its sub-problem independently, with some assumed value for the external state variables ($\mathbf{x}_{\overline{N_i}}$), and implement their chosen control actions $\mathbf{u}_{N_i}$. Under very strict conditions, an "independence decomposition" of this sort will give optimal results. Specifically, if every agent's decision at optimality will have no effect on other agents' decision, the results of independent decision-making will be optimal. Symbolically, if $\nabla_{\left(\mathbf{x}_{\overline{N_i}}\right)_*} \mathbf{g}_{M_i} = \mathbf{0}$[1] for all i, the sub-problems are completely separable and the results of independent optimization will be optimal. When the external variables ($\mathbf{x}_{\overline{N_i}}$) do appear in agent i's constraints, some cooperation among the agents is needed to obtain good results. The following describes some common types of cooperation using this optimization framework.

A. *Voting*

Agents that cooperate through a voting system must abide by additional constraints (generally laws enforced by public officials) as decided upon by the majority. We can represent a voting system in the context of optimizing agents by adding additional constraints to the agent sub-problems:

$$\min_{\mathbf{u}_{N_i}} \quad f_i(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}) \tag{14}$$

$$\text{s.t.} \quad \mathbf{g}_{N_i}\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}\right) \leq \mathbf{0} \tag{15}$$

$$\mathbf{v}\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}\right) \leq \mathbf{0} \tag{16}$$

Eq. 16 is a set of constraints that every agent must consider while making decisions. For example drivers agree to constrain their driving behavior in a environment with voted-upon traffic laws. Constraints, such as a speed limit, could be represented by inserting the speed constraint into $\mathbf{v}$. When democratically selected constraints are well-chosen, agent behaviors are more nearly optimal, though voting systems certainly do not guarantee optimality [2].

B. *Iterative information exchange*

In the independence decomposition (Eqs. 12-13) agents will obtain good results if every agent can predict the external state variables ($\mathbf{x}_{\overline{N_i}}$) after the optimization process has completed. One way for agents to estimate these variables is to iteratively exchange preliminary estimates of these variables with neighbors before making a final decision. This process was employed in [4] and shown to lead to optimal results under some conditions. The decentralized optimal power flow

[1]The subscript asterisk (x_*) is used to indicate optimality.

methods described in [10] also fall into this category. Unfortunately the number of iterations required to converge to arrive at optimality can make this type of cooperation impractical for time-critical decision making.

C. Perfect altruism

One way to guarantee optimality would be for every agent to share all of its information with every other agent and have every agent solve the global problem, implementing only the local portion of the solution. If every agent could have perfect information and would agree to act with perfect altruism (global welfare optimization) optimal outcomes could be guaranteed. Every agent would be working with the global problem and would thus arrive at the same, globally optimal, conclusion. Unfortunately this type of decomposition is impractical in all but the simplest problems for two reasons. For one it is impractical for every agent to exchange all information with every other agent. If there are n agents, the communication system will need to carry $n(n-1)$ messages to get every agent's state to every other agent. This level of communication becomes increasingly impractical as the size of the problem grows. Also, having every agent solve the global problem means that the agents must have substantial computational abilities to quickly arrive at an answer.

D. Reciprocal altruism

Between perfect altruism and complete independence (Eqs. 12, 13) is reciprocal altruism (RA). According to Trivers [19] altruism is, "behavior that benefits another organism, not closely related, while being apparently detrimental to the organism performing the behavior." In other words an agent acts altruistically when it considers the goals of other agents when making local decisions, even if doing so could be detrimental to its local goals. Reciprocal altruism occurs when agents choose to consider the goals of other agents (though not necessarily all other agents), while assuming that the other agents will act reciprocally. With these definitions we can write the goals (objectives and constraints) of reciprocally altruistic agent a as follows:

$$\min_{\mathbf{u}_{N_i} \forall i \in [a \cup R_a]} \quad f_a(\mathbf{u}_{N_a}, \mathbf{x}_{N_a}) + \sum_{b \in R_a} f_b(\mathbf{u}_{N_b}, \mathbf{x}_{N_b})$$
$$s.t. \quad \mathbf{g}_{N_i}\left(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}\right) \leq \mathbf{0},\ \forall i \in [a \cup R]$$

For reciprocal altruism to be effective, each agent must choose a set of agents (R_a) whose goals it will consider while making local decisions. There are many ways that agents can choose neighbors for goal-sharing. Wilkinson [21], for example, found that vampire bats choose to regurgitate food to other bats based on relational proximity (kinship) and the potential for reciprocation. In a community of agents that can be represented as a graph, one way to illustrate the choice of reciprocal sets based on relational proximity is to have each R_a be the set of agents that can be reached by traveling over no more than r links (see Figure 2). While there are other ways to measure proximity, this method has proved useful for our application.

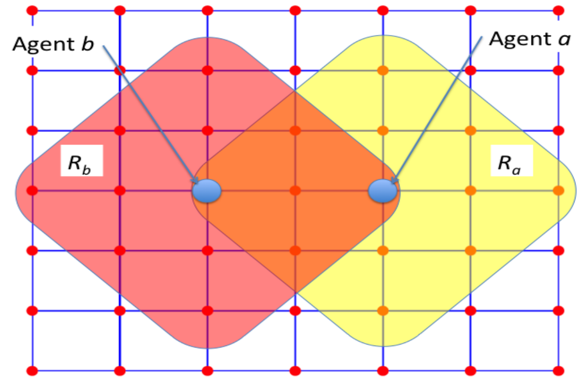

Figure 2. Agents a and b with their reciprocal sets (R_a, R_b) chosen based on a graph distance of $r = 2$.

IV. Decentralized MPC for Cascading Failure Mitigation

With a few additional details, it is possible to combine the reciprocally altruistic agents described in Section III with the ECP in Section II to design a control system that can solve the ECP with near optimality. For the power systems case, we place one agent at each substation (node) in a power network. The decision and state variables (load, generator set points, voltages and currents) and their associated costs (c and r in 1) and box constraints (3) are assigned to agents based on proximity. Agent a at bus a will be responsible for the generator or loads connected at bus a and the voltages and currents at this bus.

Each agent forms a linear state prediction function (Eq. 2) for each of its local state variables using the power-flow Jacobian. In order to form the power-flow Jacobian for a given system an agent needs four types of data: (1) the structure of the power network (branch locations and impedances), (2) the status (open or closed) of circuit breakers in the system, (3) bus voltage magnitudes and (4) the phase-angle shift across transmission lines. The initial structure of the network (1) can be provided to each agent off-line by the system operator. When an agent does not have recent measurements for (2-4), it assumes that (2) circuit breakers are closed, (3) bus voltages are at 1.0 p.u. and (4) phase-angle difference are zero. Thus each agent maintains a model of the entire network, but uses default data to make up for data that it cannot collect from its neighbors.

Once agent a assembles an initial network model, it chooses its reciprocal set (R_a), which includes two sets of agents: a set of local neighbors (L_a) and a set of extended neighbors (E_a). L_a is defined as the set of agents that can be reached by traveling no more than r_l branches and E_a the agents that are not in L_a and can be reached by traveling over no more than r_e branches. R_a is thus the union of agent a's local and extended neighbors: $R_a = L_a \cup E_a$. Agent a exchanges information (state and control variables) with agents in L_a frequently (at least once per second) and with agents in E_a occasionally (once per day to once per week). When an agent does not have good data for a given variable it uses the default values for voltages, phase-angles and circuit breakers and assumes that no significant changes will occur at remote generators or loads and that all remote currents are well below their limits. Thus the control agent at bus a uses the following formulation to make decisions:

$$\min_{\mathbf{u}_W} \quad \sum_{k=0}^{K} \rho^k \left(c(\Delta \mathbf{u}_{Wk}) + r(\mathbf{x}_{Wk}) \right) \tag{17}$$

$$\text{s.t.} \quad \mathbf{x}_{W,k+1} = \mathbf{x}_{Wk} + \mathbf{B} \Delta \mathbf{u}_{Wk} \tag{18}$$

$$\underline{\mathbf{u}}(\mathbf{u}_{Wk}) \leq \mathbf{u}_{W,k+1} \leq \overline{\mathbf{u}}(\mathbf{u}_{Wk}) \tag{19}$$

where $\mathbf{B}$ is a matrix that comes from the linearized AC power flow equations and $W = \bigcup_{i \in [R_a \cup a]} N_i$. All of the remaining variables and constraints are the same as described in Section 2. In addition to cooperating by sharing goals and information with its neighbors, each agent uses a simple negotiation protocol to improve on its decision before taking, thus combining RA and iterative information exchange. The following describes the resulting decision process for agent a:

1) Set $t_0 = t_0 + \Delta t$.
2) Gather measurements from local sensors.
3) Share local measurements with local neighbors (L_a).
4) Share measurements that are outside of limits with extended neighbors (E_a).
5) Calculate a set of control actions ($\Delta \mathbf{u}^{[a]}$) by optimizing Eqs. 17-19. When $r(\mathbf{x}_W) = 0$ the optimal action is $\Delta \mathbf{u}_k = 0$, thus skip 5-8.
6) Choose a set of agents (P) with whom to negotiate. P is chosen such that $P = \{\forall i : ||\Delta \mathbf{u}^{[a]}|| < \epsilon\}$. ϵ is chosen to ensure that agents only negotiate with agents who appear to need to take significant action.
7) For each agent $b \in P$ send a set of measurements that are current in a's model, but are not likely to be current in b's model, given the locations of a and b. If this is the first round of negotiation this set of measurements ($\mathbf{x}_Q$ and $\mathbf{u}_Q$) will be: $Q = \bigcup_{i \in L_a \cap \overline{L_b}} N_i$. Each agent b incorporates this new information into its model, which can expand W.
8) Repeat from 5 until a predefined control deadline ($t_c = t_0 + 0.8\Delta t$) is reached.
9) At time t_c implement the locally calculated control actions: $\Delta \mathbf{u}_{N_a}^{[a]}$.
10) When the current time reaches $t_0 + \Delta t$ repeat from 1.

V. Simulation Model and Results

We test the decentralized control system described in Section IV by applying it to an AC power-flow model of the IEEE 300 bus power network. Ten heavily loaded permutations of this case were generated by randomly adjusting the generator outputs and loads and setting the branch current and bus voltage limits such that no single branch outage results in a current or voltage violations. Parallel lines were added in locations where a single branch outage would have separated the grid into separate islands. Each load is given a random demand-reduction cost in the range of \$100-\$10,000/MW. For each of the ten cases, ten sets of branch outages were randomly selected by random draws from a Bernoulli distribution, selecting those cases that caused at least one current or voltage violation. Each resulting disturbance consists of between 4 and 15 branch outages.

We estimate the impact of the cascading failures, with and without mitigating control actions, using a pseudo-dynamic power flow model. The model includes simulated time overcurrent relays that remove branches when the integral of the excess current exceeds a threshold. At each time step (which is assumed to represent one second in the model) the model uses a Newton-Raphson AC power flow to find the result of switching and control actions in the previous time step. The model roughly simulates under-frequency load shedding by reducing load and generation in 25% chunks if the power-flow calculation does not converge. Using this simulation model the mean demand loss over the 100 test cases is 7,593 MW, or 32.6% of the initial demand. The simulated cascading failure costs range from \$39 to \$147,000,000, with a mean of \$5,600,000.

This same cascading failure model was used to estimate the effectiveness of the centralized and decentralized control schemes described in Sections 2 and 4. As described in Section 3, the selection of neighbors, or reciprocal sets (R_a), is essential to the effectiveness of a community of RA agents. If the reciprocal sets are too large, message-passing can overwhelm the communication channels and the agents' decision processes becomes overly complicated. If the sets are too small, agents will not make good decisions with respect to the global problem. To determine an appropriate size for R_a, we simulate the agent-based control algorithm for different r_l, measuring both the distribution of blackout costs and the amount of data exchanged by the agents. The radius of the extended neighborhood is fixed at $r_e = 10$ for these simulations. All other parameters are as described in Table I. Data exchange, or bandwidth, is measured by logging the size (in kilobytes) of each message the comes in to or goes out from each agent. Figure 3 shows that as r_l increases the average blackout size decreases and the amount of data exchanged increases. As the agents increase the extent to which they exchange information and share goals (their altruism) the system approaches global optimality, though with increasing bandwidth requirements. Simulations also indicate that the negotiation protocol (steps 6 and 7 in Section IV) can reduce the impact of the cascading failure, but substantially increase the communications burden.

VI. Conclusions

This paper describes a new approach to decentralized control, which uses reciprocally altruistic control agents and model predictive control. The agents make local decisions based on locally maintained models of the system to be controlled and goals that are shared with neighboring agents. We show that as agents share more of their goals, effectively increasing their altruism, their quality of their decisions increases but with the side effect of substantial increases in message passing. For the power network test cases used, we find that nearly globally optimal results obtain from this approach, with fairly reasonable bandwidth requirements.

While this paper focuses on the mitigation of cascading failures in electricity networks, the method described here

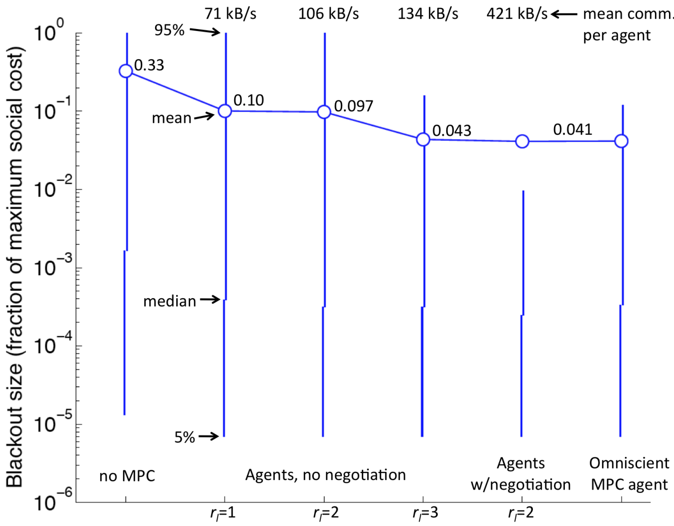

Figure 3. Cascading failure size distributions, and communications requirements, for different control scenarios. As the size of the local neighborhood (r_l) increases, the control quality approaches the results obtained from a single global, omniscient MPC agent (right), though with increasing communication bandwidth requirements. The negotiation protocol gives results that are nearly as good as the omniscient controller, but further increases the bandwidth requirements.

should be useful for a variety of network control problems, as would be found in many infrastructure systems. Future research will focus on the application of this approach to other infrastructure control problems to determine if indeed reciprocally altruistic control agents can be used within other problem domains.

References

[1] P. M. Anderson and B. K. LeReverend, "Industry experience with special protection schemes," *IEEE TRANSACTIONS ON POWER SYSTEMS*, vol. 11, no. 3, pp. 1166–1176, Aug. 1996.

[2] K. J. Arrow, "A difficulty in the concept of social welfare," *The Journal of Political Economy*, vol. 58, pp. 328–346, 1950.

[3] E. Camponogara, "Controlling networks with collaborative nets," Ph.D. dissertation, Carnegie Mellon University, 2000.

[4] E. Camponogara and S. N. Talukdar, "Designing communication networks to decompose network control problems," *INFORMS JOURNAL ON COMPUTING*, vol. 17, no. 2, pp. 207–223, 2005.

[5] B. A. Carreras, V. E. Lynch, I. Dobson, and D. E. Newman, "Critical points and transitions in an electric power transmission model for cascading failure blackouts," *CHAOS*, vol. 12, no. 4, pp. 985–994, Dec. 2002.

[6] I. Dobson, B. Carreras, V. Lynch, B. Nkei, and D. Newman, "Estimating failure propagation in models of cascading blackouts," *Probability in the Engineering and Informational Sciences*, vol. 19, pp. 475–488, 2005.

[7] A. Feliachi, K. Schoder, S. Ganesh, and H.-J. Lai;, "Distributed control agents approach to energy management in electric shipboard power system," in *Proceedings of the IEEE Power Engineering Society General Meeting*, 2006.

[8] L. Fink and K. Carlsen, "Operating under stress and strain," *IEEE Spectrum*, vol. 15, no. 3, pp. 48–53, 1978.

[9] I. Hiskens and B. Gong, "MPC-Based Load Shedding for Voltage Stability Enhancement," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005.

[10] B. H. Kim and R. Baldick, "A Comparison of Distributed Optimal Power Flow Algorithms," *IEEE Transactions on Power Systems*, vol. 15, pp. 599–604, 2000.

[11] M. Larsson, "An Adaptive Predictive Approach to Emergency Frequency Control in Electric Power Systems," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005.

[12] H. Li, G. Rosenwald, J. Jung, and C. ching Liu, "Strategic power infrastructure defense," *Proceedings of the IEEE*, 2005.

[13] Y. Makarov, V. Reshetov, V. Stroev, and N. Voropai, "Blackout Prevention in the United States, Europe, and Russia," *Proceedings of the IEEE*, vol. 93, pp. 1942–1955, 2005.

[14] J. Meisel and R. Barnard, "Application of fixed-point techniques to load-flow studies," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, pp. 136 – 140, 1970.

[15] H. Ni, G. T. Heydt, and L. Mili, "Power system stability agents using robust wide area control," *IEEE Transactions on Power Systems*, vol. 17, pp. 1123–1131, 2002.

[16] W. Shao and V. Vittal, "Corrective switching algorithm for relieving overloads and voltage violations," *IEEE Transactions on Power Systems*, vol. 20, pp. 1877 – 1885, 2005.

[17] J. Stahlhut, T. Browne, G. Heydt, and V. Vittal, "Latency viewed as a stochastic process and its impact on wide area power system control signals," *IEEE Transactions on Power Systems*, vol. 23, pp. 84 – 91, 2008.

[18] C. Taylor, D. Erickson, K. Martin, R. Wilson, and V. Venkatasubramanian, "WACS-wide-area stability and voltage control system: R&D and online demonstration," *Proceedings of the IEEE*, vol. 93, pp. 892– 906, 2005.

[19] R. L. Trivers, "The evolution of reciprocal altruism," *Quarterly Review of Biology*, vol. 46, pp. 35–57, 1971.

[20] J. Weber and T. Overbye, "An individual welfare maximization algorithm for electricity markets," *IEEE Transactions on Power Systems*, vol. 17, pp. 590 – 596, 2002.

[21] G. S. Wilkinson, "Reciprocal food sharing in the vampire bat," *Nature*, vol. 308, pp. 181–184, 1984.

[22] H. You, V. Vittal, and X. Wang, "Slow coherency-based islanding," *IEEE Transactions on Power Systems*, vol. 19, pp. 483–491, 2004.

[23] M. Zima, M. Larsson, P. Korba, C. Rehtanz, and G. Andersson, "Design Aspects for Wide-Area Monitoring and Control Systems," *Proceedings of the IEEE*, vol. 93, pp. 980–996, 2005.

[24] R. Zimmerman and D. Gan, "Matpower user's manual, version 2.0," PSERC, Tech. Rep., 1997. [Online]. Available: www.pserc.cornell.edu/matpower

Paul Hines (M'96) is an Assistant Professor in the School of Engineering at the University of Vermont. He received the Ph.D. degree in Engineering and Public Policy from Carnegie Mellon U. in 2007 and the M.S. degree in Electrical Engineering from the U. of Washington in 2001. Formerly he worked as a Research Scientist at the US National Energy Technology Laboratory, for the US Federal Energy Regulatory Commission, where he studied interactions between nuclear power plants and transmission networks, for Alstom ESCA, where he designed a short-term load forecasting tool and for Black and Veatch, where he worked on substation design projects. His main research interests are in the areas of complex systems and networks, the control of cascading failures in power systems and energy system reliability.

Sarosh Talukdar is Professor Emeritus in the Departments of Electrical and Computer Engineering and Engineering and Public Policy at Carnegie Mellon University.