Reciprocally altruistic agents for the mitigation of cascading failures in power grids

Paul Hines*

School of Engineering
University of Vermont
Burlington, VT, USA
E-mail: paul.hines@uvm.edu
*Corresponding author

Sarosh Talukdar

Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA, USA
E-mail: talukdar@ece.cmu.edu

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 centralised control architectures. Centralised 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 decentralised approach. We place control agents at each substation in a power network, each of which uses decentralised Model Predictive Control (MPC) 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 (RA). 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.

Keywords: problem decomposition; model predictive control; MPC; reciprocal altruism; cascading failures; power systems; blackouts.

Reference to this paper should be made as follows: Hines, P. and Talukdar, S. (2009) 'Reciprocally altruistic agents for the mitigation of cascading failures in power grids', *Int. J. Critical Infrastructures*, Vol. 5, No. 4, pp.340–356.

Biographical notes: Paul Hines is an Assistant Professor at the School of Engineering in the University of Vermont, USA. He is also a member of the Carnegie Mellon Electricity Industry Center Adjunct Research Faculty and a Commissioner for the Burlington Electric Department. He received his PhD Degree in Engineering and Public Policy from Carnegie Mellon University, USA in 2007 and MS Degree in Electrical Engineering from the University of Washington in 2001. His main research interests are in the areas of complex systems and networks, the control of cascading failures in power grids and energy reliability.

Sarosh Talukdar is a Professor Emeritus in the Departments of Electrical and Computer Engineering and Engineering and Public Policy at Carnegie Mellon University, USA. He studied at the Indian Institute of Technology-Madras and Purdue University, West Lafayette, Indiana, USA.

1 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 long chains 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 (14 August) and Southern European (28 September) grids in 2003 spurred a wave of new interest into the mechanics of Dobson *et al*. (2005) and Carreras *et al*. (2002) and mitigation methods for cascading failures in power networks (You *et al*., 2004; Yang *et al*., 2006; Zima *et al*., 2005; de la Ree *et al*., 2005). Because of their complex discrete and continuous dynamics, and the relatively short time constants over which these dynamics can progress, cascading failures in power networks provide a challenging test case for mitigation strategies that could be useful in other infrastructures.

Power grids, like most infrastructure systems, are operated by both decentralised and centralised controllers. The decentralised controllers (relays for example) work with local data to quickly make local decisions. The centralised controllers (human operators for example) work with larger data sets to make decisions along longer time horizons according to broader (global) goals. Sometimes local controllers with limited information do not, or cannot, make decisions that align well with global goals. For example, distance relays designed to interrupt fault current, and thus mitigate local risks in a power grid, can propagate a cascading failure, thus increasing global risk. The goal of this work is to develop decentralised control agents that gather information from their neighbours 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 decentralised control systems and low-speed, high-intelligence centralised systems.

This paper is organised as follows. Section 1 is this introduction. Section 2 describes a global Model Predictive Control (MPC) problem and shows that this paper can be used to mitigate cascading failures from a central location given perfect information. In Section 3 we present a general framework for describing optimising agents within a network, and use this framework to describe reciprocally altruistic agents. Section 4 describes the specific changes needed to design reciprocally altruistic agents to solve the global cascading failure mitigation problem. The final two sections describe the simulation model, simulation results, and draw some conclusions.

1.1 Centralised and decentralised cascading failure mitigation

Numerous centralised 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 (Anderson and LeReverend, 1996; Zima *et al*., 2005). 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 (Anderson and LeReverend, 1996; Makarov *et al*., 2005). However, if an SPS responds to stress with predetermined, non-adaptive, control actions it is possible that the actions can increase, rather than decrease, the size of a blackout. More recent WACS employ adaptive concepts, including an algorithm that uses correlation among generator phase angles to divide a grid into islands (You *et al*., 2004) and optimisation-based methods such as MPC (Larsson, 2005; Hiskens and Gong, 2005; Taylor *et al*., 2005; Shao and Vittal, 2005).

Few of these adaptive, centralised approaches have been widely adopted. One potential explanation for this is that infrastructure networks in general, and power systems in particular, are inherently decentralised 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 latency in internet communications, which has been shown to negatively affect some WACS (Stahlhut *et al*., 2008).

Some emerging designs use decentralised controllers as a part of the overall design. Li *et al*. (2005) describe a "Strategic Power Infrastructure Defence" system that employs a hierarchical, multi-agent architecture. Ni *et al*. (2002) use a multi-agent approach to coordinate the actions of generator controllers to damp inter-area oscillations. Feliachi *et al*. (2006) use decentralised control agents to reroute power flows in a DC shipboard power system by adapting the max flow problem from graph theory to a decentralised formulation of the power flow problem.

The method proposed in this paper differs from the above decentralised 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.

1.2 A formulation for emergency control in power networks

A good objective for power system operations is to maximise 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 maximisation, a form of which is used in many Optimal Power Flow formulations (Weber and Overbye, 2002). When a power system is stressed, however, 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 Fink and Carlsen (1978)), a good operating policy is to simultaneously minimise 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. Equations (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:

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

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

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

where c and r are estimates of the cost and risk associated with a trajectory of control and state variables. Assuming:

- that the system is deterministic within the time horizon and accurately modelled by **g**
- that r provides an accurate measure of the expected losses after t_K (the end of the time horizon)
- that the problem is convex
- 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 control trajectory solution ($\mathbf{U}_*$) to ECP will be at least as good as a control signal produced by any other controller.

When there is uncertainty (random variables) in a system, **g** will not provide perfect next state predictions. There are two approaches for dealing with this type of prediction uncertainty in an optimisation-based control problem. The first approach is to reformulate the problem using stochastic programming, explicitly modelling 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 modelling 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 MPC problem that can be used for centralised 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 centralised approach), a decentralised 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 decentralised approach to emergency control and Section 6 offers some concluding remarks.

2 Centralised MPC for cascading failure mitigation

This section describes how the general ECP formulation (Equations 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 formulation includes the following control variables in its control vector, $\mathbf{u}_k$: 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 approximately 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 load bus $d \in D$, such that the injection by loads at bus d is $P_d + jQ_d = \lambda_d(\overline{P_d} + j\overline{Q_d})$. Here $\overline{P_d} + j\overline{Q_d}$ represents the demand at bus d before load shedding.

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 Equation 1) represents the costs associated with changes to the control variables:

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

$$c(\Delta\mathbf{P}_{G,k}) = \mathbf{c}_G^T \Delta\mathbf{P}_{G,k} \tag{5}$$

$$c(\Delta|\mathbf{V}_{G,k}|) = \mathbf{c}_{\Delta V}^T |\Delta|\mathbf{V}_{G,k}||. \tag{6}$$

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 human operators and/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, thus allowing the system to attempt low-cost actions before high-cost actions. Discount factors are useful in planning problems, such as an MPC formulation, because uncertainty increases with time.

The risk term (r in Equation 1) estimates the impact of stress on the risk associated with foreseeable cascading failure. The risk function described below is simple, but has proved useful in practise. 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:

$$r(\mathbf{X},\mathbf{U}) = \sum_{k=1}^{K} \rho^k (\mathbf{c}_I^T \mathbf{r}_I + \mathbf{c}_V^T \mathbf{r}_V)$$

$$r_v = \max(\beta_V(k)|\underline{V_v}| - |V_{v,k}|, 0), \forall v \tag{7}$$

$$r_i = \max(|I_{i,k}| - \beta_I(k)|\underline{I_i}|, 0), \forall i \tag{8}$$

where:

$\mathbf{c}_{\overline{I}}$ and $\mathbf{c}_{\underline{V}}$	= cost vectors assigned to excess currents and low voltages
$\overline{\lvert I_i\rvert}$ and $\underline{\lvert V_v\rvert}$	= lower voltage and upper current limits
$\beta_I(k) > 1$ and $\beta_V(k) < 1$	= scalars that progressively move the effective current and voltage stress thresholds ($\beta_I(k)\overline{\lvert\mathbf{I}\rvert}$ and $\beta_V(k)\underline{\lvert\mathbf{V}\rvert}$) 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 $\overline{|I_i|} = 100\text{A}$, 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, $\mathbf{g}(\ldots)$ 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:

$$\mathbf{A}\mathbf{x}_{k+1} = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}\Delta\mathbf{u}_k \tag{9}$$

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 linearised version of the AC power-flow constraints ($\mathbf{P} + j\mathbf{Q} = \mathbf{V} \odot (\mathbf{Y}\mathbf{V})^*$), Equation (9) is a fairly accurate predictor so long as the changes at each MPC time steps are small.

Control variable limits (Equation 3) limit the control variable set points over the time horizon ($\mathbf{0} \le \lambda_k \le 1$, $\underline{\mathbf{P}_G} \le \mathbf{P}_{G,k} \le \overline{\mathbf{P}_G}$, and $\underline{|\mathbf{V}_G|} \le |\mathbf{V}_{G,k}| \le \overline{|\mathbf{V}_G|}$) and changes to the set points, as shown in Equations (10–12):

$$-\mathbf{P}_{G,0} \le \Delta\mathbf{P}_{G,k} \le \mathbf{0} \tag{10}$$

$$-\overline{\Delta|V_G|} \le \Delta|\mathbf{V}_{G,k}| \le \overline{\Delta|V_G|} \tag{11}$$

$$-\mathbf{1} \le \Delta\lambda_k \le \mathbf{0}. \tag{12}$$

The voltage change limit $\overline{\Delta|V_G|}$ is chosen to force generators to change their set points incrementally. Table 1 shows typical values for the parameters used in the ECP formulation.

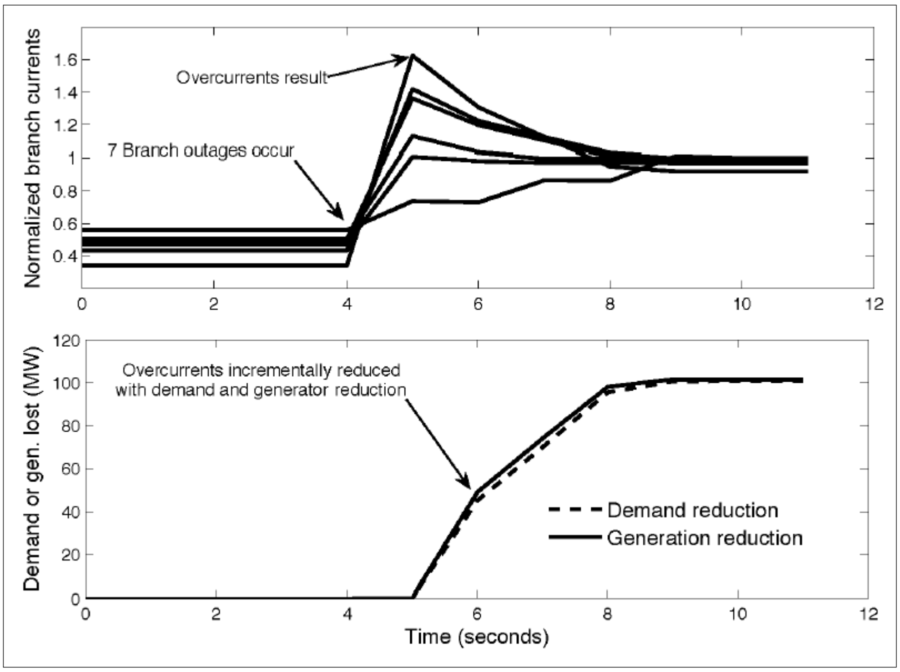
Table 1 Typical parameters for the global emergency control problem formulation

Parameter	*Typical range*	*Value used for Figure 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, 10 000]	Randomly assigned	\$/MW
$c_{\Delta V}$	[0, 10^5]	\$10,000	\$/p.u. V
c_G	[0, 100]	30	\$/MW

2.1 Illustrative results for centralised mitigation

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 (Zimmerman and Gan, 1997). Branch current limits were assigned randomly to the case using the procedure described by Hines (2007). 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. Details of the cascading failure simulator are available in Hines (2007). 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 incrementally shedding 100 MW (0.43%) of demand the ECP controller averts a system-wide cascading failure. Because the algorithm chooses load shedding based on cost-priorities substantially less of the total demand value was lost (0.035% of the demand 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 required load shedding increases from 100 MW to 164 MW.

Figure 1 Time domain simulation results showing mitigating control actions after removing seven branches from the IEEE 300 bus system



Note: Branch currents are shown after normalising such that $|I|_{fig} = |I|/|\overline{I}|$.

3 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 sub-problems, 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. The literature on problem decomposition is vast. Wooldridge (2002) and Wooldridge and Jennings (1995) provide a good overview of the subject in the context of solving computer problems using cooperative agents. Horling and Lesser (2004) discuss some more recent paradigms for multi-agent organisations. Durfee (1999) builds on game theory to describe the challenges associated with coordinating agents with different goals. A large literature exists on decomposing computer problems for parallel computation, within which the work by Bertsekas and Tsitsiklis (1997) is particularly notable.

Clearly 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 (Meisel and Barnard, 1970) is another. Series work can provide good results in many, but not all, problems (see Camponogara, 2000) 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 grid, 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 (Assumption 1). We assume (Assumption 2) that the agents are working on a generic problem that can be written in the form:

$$\min_{\mathbf{u}} \quad f(\mathbf{u},\mathbf{x})$$

$$\text{s.t.} \quad \mathbf{g}(\mathbf{u},\mathbf{x}) \leq \mathbf{0}.$$

We also assume (Assumption 3) that the feasible control space, $\mathbf{u} \in \mathfrak{U}$, and state space, $\mathbf{x} \in \mathfrak{X}$, 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:

$$\bigcup_{i=1}^{n} N_i = N : \mathbf{u}_{N,k} = \mathbf{u}_k$$

$$N_a \cap N_b = \emptyset, \forall a, b \in \{1,\ldots,n\}.$$

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

$$\min_{\mathbf{u}} \quad f(\mathbf{u},\mathbf{x}) = \sum_{i=1}^{n} f_i(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}) \tag{13}$$

$$\mathbf{g}(\mathbf{u},\mathbf{x}) = \begin{bmatrix} \mathbf{g}_{M_i}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \\ \mathbf{g}_{\overline{M_i}}(\mathbf{u}_{\overline{N_i}}, \mathbf{x}_{\overline{N_i}}) \end{bmatrix} \le \mathbf{0}. \tag{14}$$

Here the constraints are divided such that the set M_i includes each of the constraints that includes one or more of agent i's variables. The constraints that are not affected by agent i's local variables are in the set $\overline{M_i}$. If $\nabla_{\mathbf{u}_{N_i},\mathbf{x}_{N_i}} g_a$ represents the partial derivatives of constraint g_a with respect to agent i's variables, then M_i is:

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

We further assume (Assumption 4) that the control and state variables can be defined such that interactions occur only through state variables (**x**). In this form it is possible 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 straight-forward to divide the problem into a set of agent sub-problems of the form:

$$\min_{\mathbf{u}_{N_i}} \quad f_i(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}) \tag{15}$$

$$\text{s.t.} \quad \mathbf{g}_{M_i}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \le \mathbf{0}. \tag{16}$$

3.1 Regarding the generality of the decomposition assumptions

While the decomposition framework described above cannot be applied to all problems, it does work for most network problems where agents have known interactions with neighbours. From Assumption 1 we are considering only problems where agents are located in a network of some sort. If the agents are mobile in this network, this formulation would require some additional modification. Assumption 2 is not restrictive as almost any optimisation problem can be written in this form, and optimisation formulations can be used to represent almost any problem. Assuming that we can assign variables to agents (Assumption 3) is also not restrictive. It may not be natural to assign each variable to only one agent, but it is certainly feasible to do so for any problem with at least 1 agent. Assumption 4 is arguably the most restrictive assumption, in that it forbids interactions among agent-variables within the objective function. There are certainly cases where two agents' utility contains common variables. For example two agents' utility functions may vary with the price of some good (oil for example). But, in most cases at least, it is possible to redefine the utility function as depending only on a local variable (personal wealth for example), and then add a constraint for each agent, which defines how that local variable varies with the common variable. In the oil example agent *a* and agent *b* would both have constraints that define their personal wealth as a function of the price of oil. So long as interactions can be moved from objectives to constraints, Assumption 4 is not restrictive.

3.1.1 Independence decomposition

One candidate decomposition of the problem in Equations (15) and (16) 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 $\nabla_{(\mathbf{x}_{\overline{N_i}})^*} \mathbf{g}_{M_i} = \mathbf{0}$ for all *i*, the sub-problems are completely separable and the results of independent optimisation will be optimal. For example, in a perfect market, where prices are well-known and no agent has a significant affect on those prices, fully autonomous decision making leads to optimal outcomes.

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 within this optimisation framework.

3.1.2 Voting

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

$$\min_{\mathbf{u}_{N_i}} \quad f_i(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}) \tag{17}$$

$$\text{s.t.} \quad \mathbf{g}_{N_i}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \leq \mathbf{0} \tag{18}$$

$$\mathbf{v}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \leq \mathbf{0}. \tag{19}$$

Equation (19) is a set of constraints that every agent must consider while making decisions. For example drivers agree to constrain their driving behaviour in a environment with voted-upon traffic laws. Constraints, such as a speed limit, could be represented by inserting the speed constraint into **v**. When selected constraints are well-chosen, agent behaviours are more nearly optimal, though voting systems certainly do not guarantee optimality Arrow (1950).

3.1.3 Iterative information exchange

In the independence decomposition (Equations 15–16) agents will obtain good results if every agent can predict the external state variables $(\mathbf{x}_{\overline{N_i}})$ after the optimisation process has completed. One way for agents to estimate these variables is to iteratively exchange preliminary estimates of these variables with neighbours before making a final decision. This process was employed in Camponogara and Talukdar (2005) and shown to lead to optimal results under some conditions. The decentralised optimal power flow methods described in Kim and Baldick (2000) 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.

3.1.4 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. Thus agent *i*'s problem becomes:

$$\min_{\mathbf{u}_{N_i}} \quad f_i(\mathbf{u}_{N}, \mathbf{x}_{N_i}) + \sum_{j \neq i} f_j(\mathbf{u}_{N_j}, \mathbf{x}_{N_j}) \tag{20}$$

$$\text{s.t.} \quad \mathbf{g}_{M_i}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \leq \mathbf{0} \tag{21}$$

$$\mathbf{g}_{\overline{M_i}}(\mathbf{u}_{\overline{N_i}}, \mathbf{x}_{\overline{N_i}}) \leq \mathbf{0}. \tag{22}$$

If every agent has perfect information about external variables $(\mathbf{u}_{\overline{N_i}}, \mathbf{x}_{\overline{N_i}})$ and constraints $(\mathbf{g}_{\overline{M_i}})$, this decomposition leads to globally optimal outcomes because Equations (20–22) is equivalent to the global problem. Unfortunately perfect altruism 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. Given finite communications bandwidth, this level of communication becomes increasingly impractical as the size of the problem grows. For biological agents, it is certainly infeasible for every individual of a given species to regularly exchange messages with every other agent. High-bandwidth communications systems certainly increase the limits on feasible message passing, but if *n* is large and message passing needs to occur with some frequency, $n(n-1)$ messages will quickly result in delays and dropped messages. Similarly, requiring that every agent solve the global problem means that the agents must have substantial computational abilities to quickly arrive at an answer, which would be cost-prohibitive for most decentralised systems.

3.1.5 Reciprocal altruism

Between perfect altruism and complete independence (Equations 15 and 16) is Reciprocal Altruism (RA). According to Trivers (1971) altruism is, "behaviour that benefits another organism, not closely related, while being apparently detrimental to the organism performing the behaviour". 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. RA 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. Trivers found, using simple theoretical models, that there are conditions under which natural selection will select for altruism. In his models genetic alleles for altruism and non-altruism determined altruistic behaviour. To illustrate one of these models, let agent *a* be an altruist and agent *b* a non-altruist. When agent *b* fails to act altruistically toward agent *a*, agent *a* reacts by refraining from altruistic acts that benefit agent *b*. In this model there is a small net benefit over a lifetime of actions to the altruist, thus biasing natural selection toward the altruistic allele, which will therefore eventually dominate.

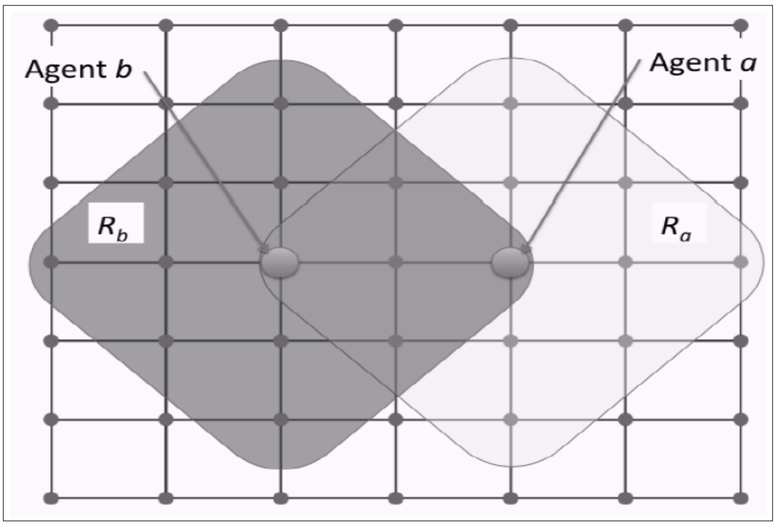
With RA thus defined we can write the goals (objectives and constraints) of reciprocally altruistic agent *a* as follows:

$$\min_{\mathbf{u}_{N_i} \forall i \in [a \cup R_a]} \quad f_a(\mathbf{u}_{N_a}, \mathbf{x}_{N_a}) + \sum_{b \in R_a} f_b(\mathbf{u}_{N_b}, \mathbf{x}_{N_b})$$

$$s.t. \quad \mathbf{g}_{N_i}(\mathbf{u}_{N_i}, \mathbf{x}_{N_i}, \mathbf{x}_{\overline{N_i}}) \leq \mathbf{0}, \ \forall i \in [a \cup R].$$

For RA 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 neighbours for goal-sharing. Wilkinson (1984), 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 travelling 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 $r = 2$ (see online version for colours)



4 Decentralised MPC for cascading failure mitigation

With a few additional details, it is possible to assign the ECP from Section 2 to reciprocally altruistic agents, and obtain nearly optimal solutions to the problem. 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 voltage and current constraints at this bus.

Each agent forms a linear state prediction function (Equation 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

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. The resulting prediction matrix thus comes from the AC power flow equations for known locations and the DC power flow simplifications for unknown locations. Each agent maintains a model of the entire network, but uses default data to make up for data that it cannot collect from its neighbours.

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 neighbours (L_a) and a set of extended neighbours (E_a). L_a is defined as the set of agents that can be reached by travelling no more than r_l branches and E_a the agents that are not in L_a and can be reached by travelling over no more than r_e branches. R_a is thus the union of agent *a*'s local and extended neighbours: $R_a = L_a \cup E_a$. 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:

$$\min_{\mathbf{u}_W} \quad \sum_{k=0}^{K} \rho^k (c(\Delta \mathbf{u}_{Wk}) + r(\mathbf{x}_{Wk})) \tag{23}$$

$$\text{s.t.} \quad \mathbf{x}_{W,k+1} = \mathbf{x}_{Wk} + \mathbf{B}\Delta \mathbf{u}_{Wk} \tag{24}$$

$$\underline{\mathbf{u}}(\mathbf{u}_{Wk}) \le \mathbf{u}_{W,k+1} \le \overline{\mathbf{u}}(\mathbf{u}_{Wk}) \tag{25}$$

where **B** is a matrix that comes from the linearised 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 neighbours, 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*:

Step 1 Set $t_0 = t_0 + \Delta t$.

Step 2 Gather measurements from local sensors.

Step 3 Share local measurements with local neighbours (L_a).

Step 4 Share measurements that are outside of limits with extended neighbours (E_a).

Step 5 Calculate a set of control actions ($\Delta \mathbf{u}^{[a]}$) by optimising Equations (23–25). When $r(\mathbf{x}_W) = 0$ the optimal action is $\Delta \mathbf{u}_k = 0$, thus skip 5–8.

Step 6 Choose a set of agents (P) with whom to negotiate. P is chosen such that $P = \{\forall i : ||\Delta \mathbf{u}^{[a]}|| < \varepsilon\}$. ε is chosen to ensure that agents only negotiate with agents who appear to need to take significant action.

Step 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*.

Step 8 Repeat from 5 until a predefined control deadline ($t_c = t_0 + 0.8\Delta t$) is reached.

Step 9 At time t_c implement the locally calculated control actions: $\Delta \mathbf{u}_{N_a}^{[a]}$.

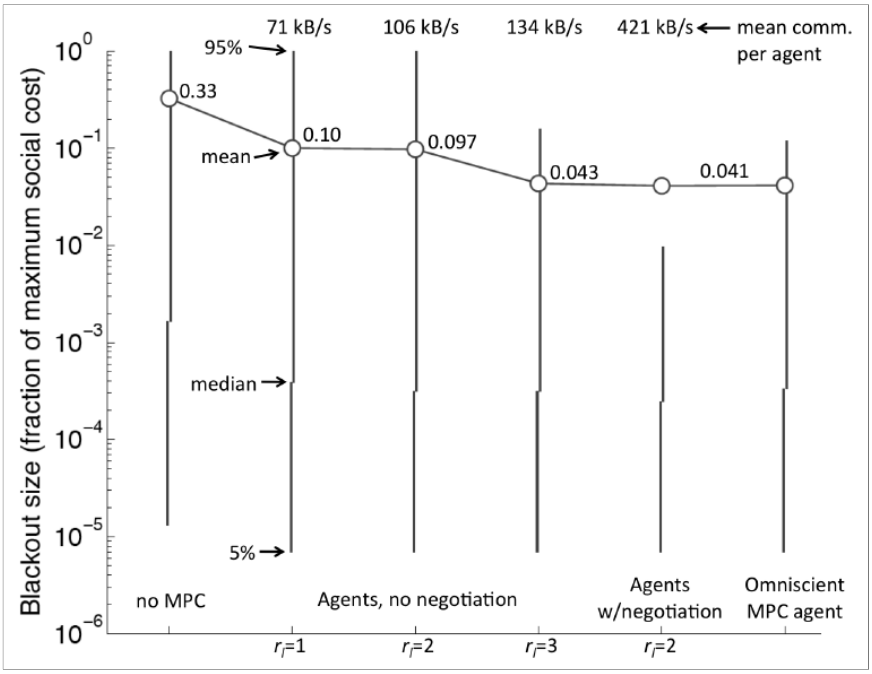
Step 10 When the current time reaches $t_0 + \Delta t$ repeat from 1.

5 Simulation model and results

We test the decentralised control system described in Section 4 by applying it to an AC power-flow model of the IEEE 300 bus power network (see Hines, 2007) for detailed model data). 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 assigned 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 over-current 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 7593 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.

Figure 3 Cascading failure size distributions, and communications requirements, for different control scenarios applied to 100 test cases



Notes: As the size of the local neighbourhood (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 produces results that are nearly as good as the omniscient controller, but further increases the bandwidth requirements.

This same cascading failure model was used to estimate the effectiveness of the centralised and decentralised control schemes described in Sections 2 and 4. As described in Section 3, the selection of neighbours, 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 neighbourhood is fixed at $r_e = 10$ for these simulations. All other parameters are as described in Table 1. Data exchange, or bandwidth, is measured by logging the size (in kilobytes) of each message that 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 4) can reduce the impact of the cascading failure, but substantially increase the communications burden.

6 Conclusions

This paper describes a new approach to decentralised control that 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 neighbouring agents. We show that as agents share more of their goals, effectively increasing their altruism, their decision-making ability 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 the proposed approach, with reasonable bandwidth requirements.

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

Anderson, P.M. and LeReverend, B.K. (1996) 'Industry experience with special protection schemes', *IEEE Transactions on Power Systems*, August, Vol. 11, No. 3, pp.1166–1176.

Arrow, K.J. (1950) 'A difficulty in the concept of social welfare', *The Journal of Political Economy*, Vol. 58, pp.328–346.

Bertsekas, D.P. and Tsitsiklis, J.N. (1997) *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific.

Camponogara, E. (2000) 'Controlling networks with collaborative nets', PhD thesis, Carnegie Mellon University.

Camponogara, E. and Talukdar, S.N. (2005) 'Designing communication networks to decompose network control problems', *Informs Journal on Computing*, Vol. 17, No. 2, pp.207–223.

Carreras, B.A., Lynch, V.E., Dobson, I. and Newman, D.E. (2002) 'Critical points and transitions in an electric power transmission model for cascading failure blackouts', *CHAOS*, December, Vol. 12, No. 4, pp.985–994.

De la Ree, J., Liu, Y., Mili, L., Phadke, A.G. and DaSilva, L. (2005) 'Catastrophic failures in power systems: causes, analyses, and countermeasures', *Proceedings of the IEEE*, Vol. 93, pp.956–964.

Dobson, I., Carreras, B.A., Lynch, V.E., Nkei, B. and Newman, D.E. (2005) 'Estimating failure propagation in models of cascading blackouts', *Probability in the Engineering and Informational Sciences*, Vol. 19, pp.475–488.

Durfee, E.H. (1999) 'Practically coordinating', *AI Magazine*, Vol. 20, No. 1, pp.99–116.

Feliachi, A., Schoder, K., Ganesh, S. and Lai, H-J. (2006) 'Distributed control agents approach to energy management in electric shipboard power system', *Proceedings of the IEEE Power Engineering Society General Meeting*, Montreal, Canada.

Fink, L.H. and Carlsen, K. (1978) 'Operating under stress and strain', *IEEE Spectrum*, Vol. 15, No. 3, pp.48–53.

Hines, P. (2007) 'A decentralized approach to reducing the social costs of cascading failures', PhD thesis, Carnegie Mellon University.

Hiskens, I.A. and Gong, B. (2005) 'MPC-based load shedding for voltage stability enhancement', *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain.

Horling, B. and Lesser, V. (2004) 'A survey of multi-agent organizational paradigms', *The Knowledge Engineering Review*, Vol. 19, No. 4, pp.281–316.

Kim, B.H. and Baldick, R. (2000) 'A comparison of distributed optimal power flow algorithms', *IEEE Transactions on Power Systems*, Vol. 15, pp.599–604.

Larsson, M. (2005) 'An adaptive predictive approach to emergency frequency control in electric power systems', *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain.

Li, H., Rosenwald, G.W., Jung, J. and Liu, C.C. (2005) 'Strategic power infrastructure defense', *Proceedings of the IEEE*, Vol. 93, No. 5, pp.918–933.

Makarov, Y.V., Reshetov, V.I., Stroev, V.A. and Voropai, N.I. (2005) 'Blackout prevention in the United States, Europe, and Russia', *Proceedings of the IEEE*, Vol. 93, pp.1942–1955.

Meisel, J. and Barnard, R.D. (1970) 'Application of fixed-point techniques to load-flow studies', *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS-89, No. 1, pp.136–140.

Ni, H., Heydt, G.T. and Mili, L. (2002) 'Power system stability agents using robust wide area control', *IEEE Transactions on Power Systems*, Vol. 17, pp.1123–1131.

Shao, W. and Vittal, V. (2005) 'Corrective switching algorithm for relieving overloads and voltage violations', *IEEE Transactions on Power Systems*, Vol. 20, pp.1877–1885.

Stahlhut, J.W., Browne, T.J., Heydt, G.T. and Vittal, V. (2008) '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.

Taylor, C.W., Erickson, D.C., Martin, K.E., Wilson, R.E. and Venkatasubramanian, V. (2005) 'WACS-wide-area stability and voltage control system: R&D and online demonstration', *Proceedings of the IEEE*, Vol. 93, pp.892–906.

Trivers, R.L. (1971) 'The evolution of reciprocal altruism', *Quarterly Review of Biology*, Vol. 46, pp.35–57.

Weber, J.D. and Overbye, T.J. (2002) 'An individual welfare maximization algorithm for electricity markets', *IEEE Transactions on Power Systems*, Vol. 17, pp.590–596.

Wilkinson, G.S. (1984) 'Reciprocal food sharing in the vampire bat', *Nature*, Vol. 308, pp.181–184.

Wooldridge, M. (2002) *Introduction to MultiAgent Systems*, John Wiley and Sons.

Wooldridge, M. and Jennings, N.R. (1995) 'Intelligent agents: theory and practice', *The Knowledge Engineering Review*, Vol. 10, pp.115–152.

Yang, B., Vittal, V. and Heydt, G.T. (2006) 'Slow-coherency-based controlled islanding: a demonstration of the approach on the August 14, 2003 blackout scenario', *IEEE Transactions on Power Systems*, Vol. 21, pp.1840–1847.

You, H., Vittal, V. and Wang, X. (2004) 'Slow coherency-based islanding', *IEEE Transactions on Power Systems*, Vol. 19, pp.483–491.

Zima, M., Larsson, M., Korba, P., Rehtanz, C. and Andersson, G. (2005) 'Design aspects for wide-area monitoring and control systems', *Proceedings of the IEEE*, Vol. 93, pp.980–996.

Zimmerman, R.D. and Gan, D. (1997) 'Matpower user's manual, version 2.0', Technical report, PSERC, www.pserc.cornell.edu/matpower.