Doctoral Dissertations 2014-current

Off-campus UMass Amherst users: To download campus access dissertations, please use the following link to log into our proxy server with your UMass Amherst user name and password.

Non-UMass Amherst users: Please talk to your librarian about requesting this dissertation through interlibrary loan.

Dissertations that have an embargo placed on them will not be available to anyone until the embargo expires.

Making Networks Robust to Component Failures

Download

Daniel Gyllstrom, *Computer Sciences*

Follow

Included in

OS and Networks Commons

SHARE

Document Type
Open Access Dissertation

Degree Name
Doctor of Philosophy (PhD)

Degree Program
Computer Science

Year Degree Awarded
Spring 2014

First Advisor
Jim Kurose

Second Advisor
Prashant Shenoy

Third Advisor
Deepak Ganesan

Keywords
networking, smart grid, reliability, algorithms, NP-Complete

Subject Categories
OS and Networks

Abstract
In this thesis, we consider instances of component failure in the Internet and in networked cyber-physical systems, such as the communication network used by the modern electric power grid (termed the *smart grid*). We design algorithms that make these networks more robust to various component failures, including failed routers, failures of links connecting routers, and failed sensors. This thesis divides into three parts: recovery from malicious or misconfigured nodes injecting false information into a

distributed system (e.g., the Internet), placing smart grid sensors to provide measurement error detection, and fast recovery from link failures in a smart grid communication network.

First, we consider the problem of malicious or misconfigured nodes that inject and spread incorrect state throughout a distributed system. Such false state can degrade the performance of a distributed system or render it unusable. For example, in the case of network routing algorithms, false state corresponding to a node incorrectly declaring a cost of 0 to all destinations (maliciously or due to misconfiguration) can quickly spread through the network. This causes other nodes to (incorrectly) route via the misconfigured node, resulting in suboptimal routing and network congestion. We propose three algorithms for efficient recovery in such scenarios and evaluate their efficacy.

The last two parts of this thesis consider robustness in the context of the electric power grid. We study the use and placement of a sensor, called a Phasor Measurement Unit (PMU), currently being deployed in electric power grids worldwide. PMUs provide voltage and current measurements at a sampling rate orders of magnitude higher than the status quo. As a result, PMUs can both drastically improve existing power grid operations and enable an entirely new set of applications, such as the reliable integration of renewable energy resources. However, PMU applications require *correct* (addressed in thesis part 2) and *timely* (covered in thesis part 3) PMU data. Without these guarantees, smart grid operators and applications may make incorrect decisions and take corresponding (incorrect) actions.

The second part of this thesis addresses PMU measurement errors, which have been observed in practice. We formulate a set of PMU placement problems that aim to satisfy two constraints: place PMUs "near" each other to allow for measurement error detection and use the minimal number of PMUs to infer the state of the maximum number of system buses and transmission lines. For each PMU placement problem, we prove it is NP-Complete, propose a simple greedy approximation algorithm, and evaluate our greedy solutions.

In the last part of this thesis, we design algorithms for fast recovery from link failures in a smart grid communication network. We propose, design, and evaluate solutions to all three aspects of link failure recovery: (a) link failure detection, (b) algorithms for pre-computing backup multicast trees, and (c) fast backup tree installation.

To address (a), we design link-failure detection and reporting mechanisms that use OpenFlow to detect link failures when and where they occur *inside* the network. OpenFlow is an open source framework that cleanly separates the control and data planes for use in network management and control. For part (b), we formulate a new problem, Multicast Recycling, that pre-computes backup multicast trees that aim to minimize control plane signaling overhead. We prove Multicast Recycling is at least NP-hard and present a corresponding approximation algorithm. Lastly, two control plane algorithms are proposed that signal data plane switches to install pre-computed backup trees. An optimized version of each installation algorithm is designed that finds a near minimum set of forwarding rules by sharing forwarding rules across multicast groups. This optimization reduces backup tree install time and associated control state. We implement these algorithms using the POX open-source OpenFlow controller and evaluate them using the Mininet emulator, quantifying control plane signaling and installation time.