# Improving System Reliability for Cyber-Physical Systems

*Thesis proposal*

**Leon Wu**
Department of Computer Science
Columbia University
1214 Amsterdam Avenue
Mailcode 0401
New York, NY 10027
leon@cs.columbia.edu

Advisor: Prof. Gail E. Kaiser

May 22, 2011

## Abstract

System reliability is a fundamental requirement of *Cyber-Physical System*, *i.e.*, a system featuring a tight combination of, and coordination between, the systems computational and physical elements. Cyber-physical system includes systems ranging from the critical infrastructure such as power grid and transportation system to the health and biomedical devices. An unreliable system often leads to disruption of service, financial cost and even loss of human life. This thesis aims to improve system reliability for cyber-physical systems that meet following criteria: processing large amount of data; employing software as a system component; running online continuously; having operator-in-the-loop because of human judgment and accountability requirement for safety critical systems. The reason that I limit the system scope to this type of cyber-physical system is that this type of cyber-physical systems are important and becoming more prevalent.

To improve system reliability for this type of cyber-physical systems, I propose a system evaluation approach named *automated online evaluation*. It works in parallel with the cyber-physical system to conduct automated evaluation at the multiple stages along the workflow of the system continuously and provide operator-in-the-loop feedback on reliability improvement. It is an approach whereby data from cyber-physical system is evaluated. For example, abnormal input and output data can be detected and flagged through data quality analysis. As a result, alerts can be sent to the operator-in-the-loop. The operator can then take actions and make changes to the system based on the alerts in order to achieve minimal system downtime and higher system reliability. To implement the proposed approach, I further propose a system architecture named *ARIS (Autonomic Reliability Improvement System)*.

One technique used by the approach is *data quality analysis using computational intelligence* that applies computational intelligence in evaluating data quality in some automated and efficient way to ensure data quality and make sure the running system to perform as expected reliably. The computational intelligence is enabled by machine learning, data mining, statistical and probabilistic analysis, and other intelligent techniques. In a cyber-physical system, the data collected from the system, *e.g.*, software bug reports, system status logs and error reports, are stored in some databases. In my approach, these data are analyzed via data mining and other intelligent techniques so that useful information on system reliability including erroneous data and abnormal system state can be concluded. These reliability related information are directed to operators so that proper actions can be taken, sometimes proactively based on the predictive results, to ensure the proper and reliable execution of the system.

Another technique used by the approach is *self-tuning* that automatically self-manages and self-configures the evaluation system to ensure it adapts itself based on the changes in the system and feedback from the operator. The self-tuning adapts the evaluation system to ensure its proper functioning, which leads to a more robust evaluation system and improved system reliability.

For feasibility study of the proposed approach, I first present *NOVA (Neutral Online Visualization-aided Autonomic)* system, a data quality analysis system for improving system reliability for power grid cyber-physical system. I then present a feasibility study on effectiveness of some self-tuning techniques, including data classification, redundancy checking and trend detection. The self-tuning leads to an adaptive evaluation system that works better under system changes and operator feedback, which will lead to improved system reliability.

The contribution of the work is an automated online evaluation approach that is able to improve system reliability for cyber-physical systems in the domain of interest as indicated above. It enables online reliability assurance of the deployed systems that are not possible to perform robust tests prior to actual deployment.

# Contents

# 1 Introduction

*Cyber-Physical System* (CPS) is a system featuring a tight combination of, and coordination between, the system's computational and physical elements [48]. "Applications of CPS arguably have the potential to dwarf the 20-th century IT revolution. They include high confidence medical devices and systems, assisted living, traffic control and safety, advanced automotive systems, process control, energy conservation, environmental control, avionics, instrumentation, critical infrastructure control (electric power, water resources, and communications systems for example), distributed robotics (telepresence, telemedicine), defense systems, manufacturing, and smart structures. It is easy to envision new capabilities, such as distributed micro power generation coupled into the power grid, where timing precision and security issues loom large. Transportation systems could benefit considerably from better embedded intelligence in automobiles, which could improve safety and efficiency. Networked autonomous vehicles could dramatically enhance the effectiveness of our military and could offer substantially more effective disaster recovery techniques. Networked building control systems (such as HVAC and lighting) could significantly improve energy efficiency and demand variability, reducing our dependence on fossil fuels and our greenhouse gas emissions [28]."

System reliability is a fundamental requirement of cyber-physical systems. An unreliable system often leads to disruption of service, financial cost and even loss of human life [1]. More importantly, cyber-physical system may not be deployed into some mission critical applications such as traffic control, automotive safety and health care without improved reliability and predictability [28].

This thesis aims to improve system reliability for cyber-physical systems that meet following criteria: processing large amount of data; employing software as a system component; running online continuously; having operator-in-the-loop because of human judgment and accountability requirement for safety critical systems. The reason that I limit the system scope to this type of cyber-physical system is that this type of cyber-physical systems are important and becoming more prevalent [20]. Systems that meet these criteria include power grid and energy system, highway transportation system, defense system, factory automation, and cloud computing data center. These systems will not be operating in a controlled environment, and must be robust to unexpected conditions and adaptable to subsystem failures [20]. It is often not possible to perform robust tests on a cyber-physical system prior to actual deployment because the physical devices are so expensive that they cannot be replicated in the testing lab, at least not for large scale. Thus it is imperative to have an online quality assurance process to continuously evaluate the live system in the field to ensure it is running reliably as expected.

To improve system reliability for cyber-physical systems in the domain of interest as indicated above, I propose a system evaluation approach named *automated online evaluation*. It works in parallel with the cyber-physical system to conduct automated evaluation at the multiple stages along the workflow of the system continuously and provide operator-in-the-loop feedback on reliability improvement. It is an approach whereby data from cyber-physical system is evaluated. For example, abnormal input and output data can be detected and flagged through data quality analysis. As a result, alerts can be sent to the operator-in-the-loop. The operator can then take actions and make changes to the system based on the alerts in order to achieve minimal system downtime and higher system reliability. To implement the proposed approach, I further propose a system architecture named *ARIS (Autonomic Reliability Improvement System)*.

One technique used by the approach is *data quality analysis using computational intelligence* that

applies computational intelligence in evaluating data quality in some automated and efficient way to ensure data quality and make sure the running system to perform as expected reliably. The computational intelligence is enabled by machine learning, data mining, statistical and probabilistic analysis, and other intelligent techniques. In a cyber-physical system, the data collected from the system, *e.g.*, software bug reports, system status logs and error reports, are stored in some databases. In my approach, these data are analyzed via data mining and other intelligent techniques so that useful information on system reliability including erroneous data and abnormal system state can be concluded. These reliability related information are directed to operators so that proper actions can be taken, sometimes proactively based on the predictive results, to ensure the proper and reliable execution of the system.

Another technique used by the approach is *self-tuning* that automatically self-manages and self-configures the evaluation system to ensure it adapts itself based on the changes in the system and feedback from the operator. The self-tuning adapts the evaluation system to ensure its proper functioning, which leads to a more robust evaluation system and improved system reliability.

For feasibility study of the proposed approach, I first present *NOVA (Neutral Online Visualization-aided Autonomic)* system for improving system reliability of power grid cyber-physical system. NOVA is a data quality analysis system that is able to provide objective evaluation of the machine learning and data mining software to ensure they are running as expected, the quality of the data input and output, and the consequential benefits, *i.e.*, physical system improvements, after the actions recommended by the machine learning and data mining systems have been taken.

I then present a feasibility study on effectiveness of some self-tuning techniques, including data classification, automatic redundancy checking and trend detection. The self-tuning leads to an adaptive evaluation system that works better under system changes and operator feedback, which will lead to improved system reliability.

In the following section, I will give definitions of terms and problem statement, along with requirements of the prospect solution. In section 3, I will describe proposed automated online evaluation approach and hypotheses. In section 4, I will describe proposed ARIS system architecture. In section 5, I will describe feasibility study, including section 5.1 NOVA system for data quality analysis and section 5.2 self-tuning. Section 6 compares some related work, followed by my research plan and schedule in Section 7. Section 8 lists some expected contributions. Section 9 is conclusion, followed by future work in Section 10 and acknowledgements.

# 2 Problem Statement

## 2.1 Definitions

This section formalizes some of the terms used throughout this proposal.

- *Cyber-physical system (CPS)* is a system featuring a tight combination of, and coordination between, the system's computational and physical elements [48]. The applicable domains of cyber-physical system include critical infrastructure such as power grid and highway transportation system, health and biomedical system, energy and industrial automation system, automated defense and combat system, and agricultural automation system [20].
- *System reliability* is the ability of a system or component to perform its required functions under stated conditions for a specified period of time [48]. It includes all parts of the system, including hardware, software, supporting infrastructure, operators and procedures. It is often reported as a probability.
- *Software reliability* is the probability of failure-free software operation for a specified period of time in a specified environment [3]. For cyber-physical system, software reliability is a part of system reliability.
- *Data quality* is an assessment of data's fitness to serve its purpose in a given context. Some aspects of data quality include: accuracy, completeness, update status, relevance, consistency across data sources, reliability, appropriate presentation, and accessibility [42].
- *Software intelligence* refers to a set of skills, technologies, applications and practices, used by an organization, to acquire a better understanding of its software assets and software projects. It offers software practitioners up-to-date and pertinent information to support their decision-making processes during the different stages of the software development life cycle [21]. Data quality analysis using computational intelligence has different goal and scope than the software intelligence, although they may use some similar techniques such as data mining.
- *Autonomic* is a system characteristic that means being able to control its internal functions and operations, being able to change its operation (*i.e.*, its configuration, state and functions), and being able to monitor (sense) its operational context as well as its internal state in order to be able to assess if its current operation serves its purpose [48].
- *Fault* is an incorrect step, process, or data definition in a program [3]. It is a programming error that leads to an erroneous result in some programs during execution. A *software bug* is the common term used to describe a fault in a program that produces an incorrect or unexpected result, or causes it to behave in unintended ways. *Fault density* is the number of software faults, usually expressed as faults per thousand lines of code. It is a common software reliability metric.
- *Failure* is the inability of a system or component to perform its required function within the specified performance requirement [3]. *Error* is the difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.

## 2.2 Problem Statement

It is difficult to make cyber-physical system reliable. First of all, it is difficult to make all parts of the system, including hardware, software, supporting infrastructure, operators and procedures, work together reliably, especially for those online systems that require continuous 24x7 uptime. Secondly, cyber-physical systems deployed in the field have to run in the live environments, which are not controlled and often unpredictable. Furthermore, for those cyber-physical systems that process large amount of data on the fly, it is difficult to ensure the data quality reliably. The erroneous input data and abnormal software results may cause system malfunction and service disruption.

This thesis aims to improve system reliability for cyber-physical systems that meet following criteria:

- Processing large amount of data
- Employing software as a system component
- Running online continuously
- Having operator-in-the-loop because of human judgment and accountability requirement for safety critical systems

The reason that I limit the system scope to this type of cyber-physical system is that this kind of cyber-physical systems are important and becoming more prevalent [20]. Systems that meet these criteria include power grid and energy system, highway transportation system, defense system, factory automation, and cloud computing data center. A typical example is energy control systems, whereas the sensors and actuators physically monitor and control the energy processes; the computer-based systems analyze and store data; and the communication networks interconnect the process and computer systems [32]. Another example is defense systems that will be more attuned to their environments, receiving and processing massive amounts of data, to determine courses of action [32]. This thesis is limited to cyber-physical systems in these domains.

## 2.3 Requirements

A solution to this problem must meet the following requirements:

- The approach should be able to improve system reliability for cyber-physical systems.
- The system reliability improvement brought by the approach should be able to be measured and verified quantitatively.
- The approach should make system reliability analysis and assurance more effective and efficient.
- The approach should be able to ensure online system executing as expected reliably and deal with erroneous data input and abnormal software results.
- The approach should be able to process large amount of available system data and derive useful information from them intelligently for reliability analysis.
- The approach should be able to reduce manual work, thus reducing human labor cost.

# 3 Proposed Approach and Hypotheses

## 3.1 Proposed Approach

To solve the problems mentioned above, I propose a system evaluation approach named *automated online evaluation* that is able to improve reliability for cyber-physical systems in the domain of interest as indicated above objectively, effectively, and efficiently. As illustrated in Figure 1, it works in parallel with the cyber-physical system to conduct automated evaluation at the multiple stages along the workflow of the system continuously and provide operator-in-the-loop feedback on reliability improvement. It is an approach whereby data from cyber-physical system is evaluated. For example, abnormal input and output data can be detected and flagged through data quality analysis. As a result, alerts can be sent to the operator-in-the-loop. The operator can then take actions and make changes to the system based on the alerts in order to achieve minimal system downtime and higher system reliability. The self-tuning component automatically self-manage and self-configure the evaluation system adaptively to ensure its proper functioning.

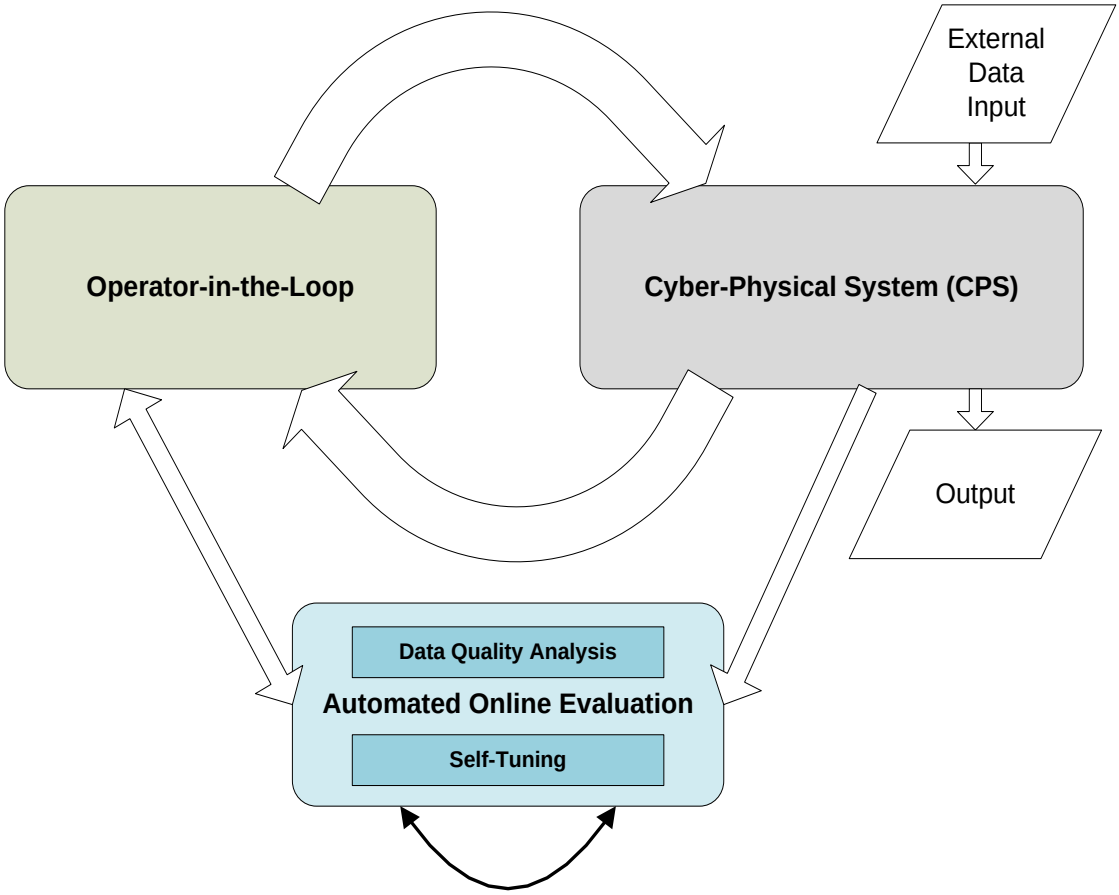


Figure 1: Proposed approach.

One technique used by the approach is *data quality analysis using computational intelligence* that applies computational intelligence in evaluating data quality in some automated and efficient way to ensure data quality and make sure the running system to perform as expected reliably. The computational intelligence is enabled by machine learning, data mining, statistical and probabilistic analysis, and other intelligent techniques. In a cyber-physical system, the data collected from the system, *e.g.*, software bug reports, system status logs and error reports, are stored in some databases. In my approach, these data are analyzed via data mining and other intelligent techniques so that useful information on system reliability including erroneous data and abnormal system state can be concluded. These reliability related information are directed to operators so that proper actions can be taken, sometimes proactively based on the predictive results, to ensure the proper and reliable execution of the system.

Another technique used by the approach is *self-tuning* that automatically self-manages and self-configures the evaluation system to ensure it adapts itself based on the changes in the system and feedback from the operator. One example of the types of self-tuning is to self-configure the evaluation system to adapt to the changes in the software models or thresholds, which may lead to different expected values and ranges for the data.

The evaluation is online, which differs from many statically analyzed systems that often employ a pre-deployment or postmortem evaluation and analysis. The evaluation is also autonomic because it works in parallel with the cyber-physical system to automatically alert the operator when abnormal events happen and it is able to self-tuning the evaluation system adaptively.

The approach does not aim to address software reliability, but it addresses software reliability through evaluating the data used by the software.

## 3.2 Hypotheses

I will prove following hypotheses in my thesis work.

The foremost hypothesis is that the automated online evaluation empowered by data quality analysis using computational intelligence can work effectively to improve system reliability for cyber-physical systems in the domain of interest as indicated above. In order to prove this hypothesis, a prototype system needs to be developed and deployed in some complex cyber-physical systems for measurement of the system reliability improvement the approach brings to the system.

The second hypothesis is that the self-tuning can effectively self-manage and self-configure the evaluation system based on the changes in the system and feedback from the operator-in-the-loop to improve system reliability.

The third hypothesis is that the approach should advance the state-of-the-art research in system reliability for cyber-physical system not only in its novel architectural design and capability in improving system reliability, but also in the new techniques developed and employed.

The fourth hypothesis is that the approach is efficient. It should not have large impact on the overall system performance and only introduce minimal extra overhead to the cyber-physical system.

# 4 Proposed Architecture

To implement the proposed approach, I propose a system named *ARIS (Autonomic Reliability Improvement System)*. It works in parallel with the software running as a part of the cyber-physical system and conducts automated and integrated evaluation at multiple stages along the workflow of the system.

As illustrated in Figure 2, to evaluate the system, it uses three stages of data quality analysis (*i.e.*, step 1, 2 and 3): first, evaluation of the input data; second, evaluation of the data output; third, evaluation of the feedback from the cyber-physical system.

The input data evaluation checks to see if the input data meets the data quality specifications predefined by the application developer and the system operator. Examples of data quality specification include data existence, up-to-date, conforming to certain distribution, time-synchronization across different sources, variation and pattern.

The output data evaluation checks the quality of the results of the application. For example, for a machine learning-based prediction system, the quality of the data output relates to the accuracy or confidence level of the prediction. For a non machine learning-based system, such as a building energy management system, the quality of the data output relates to the optimal results that can be used for subsequent actions, *e.g.*, building energy use adjustment.

The evaluation of the feedback from the cyber-physical system checks the outcome brought to the cyber-physical system by the prior steps. This evaluation is important to ensure that the data output in fact leads to the desired system outcome.

As shown as step 4 in Figure 2, the evaluation results of the data quality analysis are eventually directed to an user interface for system operators, who may take control or recovery actions when abnormal and erroneous situation happens. These actions ensure the proper execution of the system and lead to improved system reliability.

At step 5 and 6, as illustrated in Figure 2, the self-tuning component receives feedback from operator-in-the-loop and changes in the system.

At step 7, as illustrated in Figure 2, the self-tuning component self-manages and self-configures the evaluation system based on the feedback from the operator and the changes in the system. The self-tuning adapts the evaluation system to ensure its proper functioning, which leads to a more robust evaluation system and improved system reliability.

## 4.1 Use Case

To further illustrate the proposed architecture, I will describe an example use case engaging multiple steps and actions using ARIS. A *Building Management System (BMS)* is a computer-based control system installed in buildings that controls and monitors the building's mechanical and electrical equipment such as ventilation, lighting, power systems, fire systems, and security systems [48]. BMS is a type of cyber-physical systems consisting of software and hardware. Among all the functions of the BMS, the building energy control system is an important component that reads data feeds representing internal and exogenous conditions (*e.g.*, temperature, humidity, electricity load, peak load, fluctuating electricity pricing, and building work schedule) and take control actions (*e.g.*, adjusting lighting,

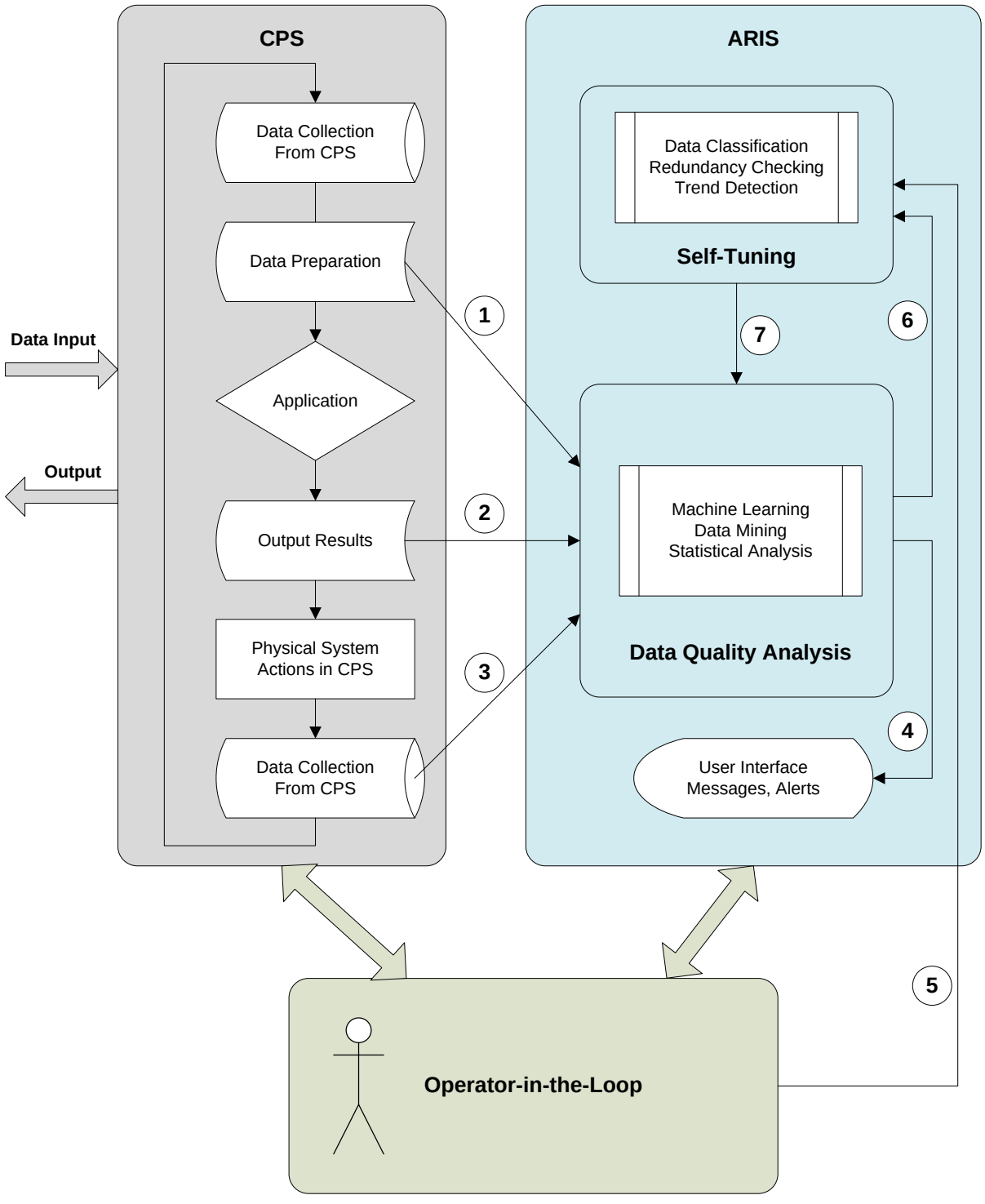


Figure 2: ARIS system architecture.

turning on/off the AC, and shutting off partial elevators) accordingly. The building's operators usually have the ability to change or override control actions taken by the BMS to accommodate some special situations such as severe weather condition or change in the building's work schedule.

To ensure the building energy control system to work reliably 24x7, the input data, output data (*i.e.*, control actions), and the result of the actions are to be evaluated using ARIS. In one example scenario, a malfunction of the digital thermostat leads to a temperature reading to stay at previous reading level and unchanged for a long time. The building energy control system has been designed to accept any value within certain temperature range. It would not be able to handle this input data error, *i.e.*, constant temperature. While, ARIS's intelligent data quality analysis component can quickly detect this type of input data error (*i.e.*, step 1 in the Figure 2), and give feedback to operator (*i.e.*, step 4 in the Figure 2). After receiving the automated notification from ARIS, the building's operator can then take action accordingly.

In another example scenario, the building's operator gets notice from the management that requires fully functioning building for a special one time only event during the coming weekend. The operator then notifies the ARIS about the abrupt change (*i.e.*, step 5 in the Figure 2). The self-tuning component of the ARIS takes this signal and transforms it for adjusting the data quality analysis (*i.e.*, step 7 and 6 in the Figure 2), thus avoiding possible false positive results of system warning due to the abnormal energy use data during this specific weekend.

# 5 Feasibility Study

In Section 5.1, I present a prototype of ARIS system that uses data quality analysis. In Section 5.2, I present some findings on the effectiveness of a self-tuning evaluation system.

## 5.1 Data Quality Analysis

In the following subsections, I present a feasibility study on improving power grid cyber-physical system reliability using data quality analysis. First, I will describe some background information on power grid and its system reliability. Then I will describe the NOVA system, a prototype implementation of ARIS system, followed by experimental results and analysis.

### 5.1.1 Power Grid as a Cyber-Physical System

As a type of critical infrastructure, power grid, *i.e.*, the electricity distribution and transmission system, is a typical continuously running cyber-physical system that processes large amount of data, uses software as a system component, and has operator-in-the-loop. In the past few years, power grid has been transitioning to *smart grid*, which is an automated electric power system that monitors and controls grid activities, ensuring the two-way flow of electricity and information between power plants and consumers—and all points in between [14]. Without the smart grid, many emerging clean energy technologies such as electric vehicles and solar, wind or cogeneration power cannot be adopted on a large scale [2].

### 5.1.2 System Reliability for Power Grid

It is a critical challenge to ensure power grid reliability. In fact, the power grid has become less reliable and more outage-prone in the past years. According to two data sets, one from the U.S. Department of Energy and the other one from the North American Electric Reliability Corp., the number of power outages greater than 100 Megawatts or affecting more than 50,000 customers in the U.S. almost doubled every five years in the past fifteen years, resulting in about $49 billion outage costs per year [1].

One of the main causes of the power grid failure is electrical component failure. The smart grid of the future will have to operate efficiently to satisfy the increasing capacity demand, and should use the current legacy grid as much as possible to keep costs lower. The legacy grid often contains old and unreliable electrical components. The electrical component failures may even lead to catastrophic cascading system failures. In 2004, the U.S.-Canada Power System Outage Task Force released their final report on the 2003 U.S. Northeast blackout placing the main cause of the blackout on some strained high-voltage power lines in Ohio that later went out of service, which led to the cascading effect that ultimately forced the shutdown of more than 100 power plants [15].

To tackle this electrical component failure problem, researchers at Columbia University have collaborated with the Consolidated Edison of New York, the main power utility provider of New York City, and developed several machine learning and data mining systems to rank some types of electrical components such as feeders, *i.e.*, transmission lines with radial circuit of intermediate voltage, by their susceptibility to impending failure. The rankings can then be used for planning of fieldwork

aimed at preventive maintenance, where the components should be proactively inspected and/or repaired in order of their estimated susceptibility to failure [37, 36, 19]. The preventive maintenance improves power grid system reliability.

MartaRank [6, 29] and ODDS [19] are two online machine learning and data mining-based feeder-ranking systems for preventive maintenance. MartaRank employs Support Vector Machines (*SVM*), RankBoost, Martingale Boosting and an ensemble-based wrapper. The ODDS ranking system uses ranked lists obtained from a linear SVM.

#### 5.1.3 NOVA System for Improving Power Grid System Reliability

To improve power grid system reliability, it requires objective evaluation of the machine learning and data mining software to ensure they are running as expected, the quality of the data input and output, and the consequential benefits, *i.e.*, physical system improvements, after the actions recommended by the machine learning and data mining systems have been taken. For this purpose, I have developed *NOVA (Neutral Online Visualization-aided Autonomic)* system, a prototype ARIS system for data quality analysis, that is able to provide such an evaluation objectively, effectively, and efficiently [52, 53]. Note that NOVA is not self-tuning.

NOVA conducts an automated and integrated evaluation at multiple stages along the workflow of the cyber-physical system. There are three steps provided through a unified user interface, as illustrated in Figure 3: first, evaluation of the input data; second, evaluation of the machine learning and data mining output; third, evaluation of the system's performance improvement. The results from Step 1, 2 and 3 are eventually directed to a centralized software dashboard for operator-in-the-loop to take actions. When abnormal results trigger pre-defined thresholds at any step, warning messages are dispatched automatically. I implemented NOVA in evaluating MartaRank and ODDS feeder-ranking systems and analyzed the experimental results.

In the following subsections, I will describe the details of each evaluation stage and demonstrate useful summarization charts for each step.

##### 5.1.3.1 Evaluation of Input Data Quality

In order for a system to perform as expected, the input data sets have to meet the pre-defined quality specifications. The evaluation process first uses *data constraints and checks* to see whether the required data exist and are up to date. Then the evaluation process conducts some more fine-grained checks, for example by using a *sparkline graph*, which is a type of information graphic characterized by its small size and high data density [43]. These checks would help researchers to correlate the changes in the input data sets with the variations of machine learning and data mining results, so that further study may be done to improve machine learning and data mining accuracy, thus leading to better rankings/actions and improved system reliability. As illustrated in Figure 4, in the sparkline time series graph, for the one-day period preceding an actual outage, among ten feeder attributes—maximum scaled voltage, number of joints, number of cables, peak load, etc.— being plotted, some attributes show varied patterns (*e.g.*, Attribute 1, 2, 5, 6, 7, and 10), while others are constant (*e.g.*, Attribute 3, 4, 8, and 9). These patterns may be used to improve machine learning and data mining results. For example, it may be possible that the constant attributes can be avoided so that only varied attributes are used as input data, which simplifies and improves the processing of the machine learning and data mining.

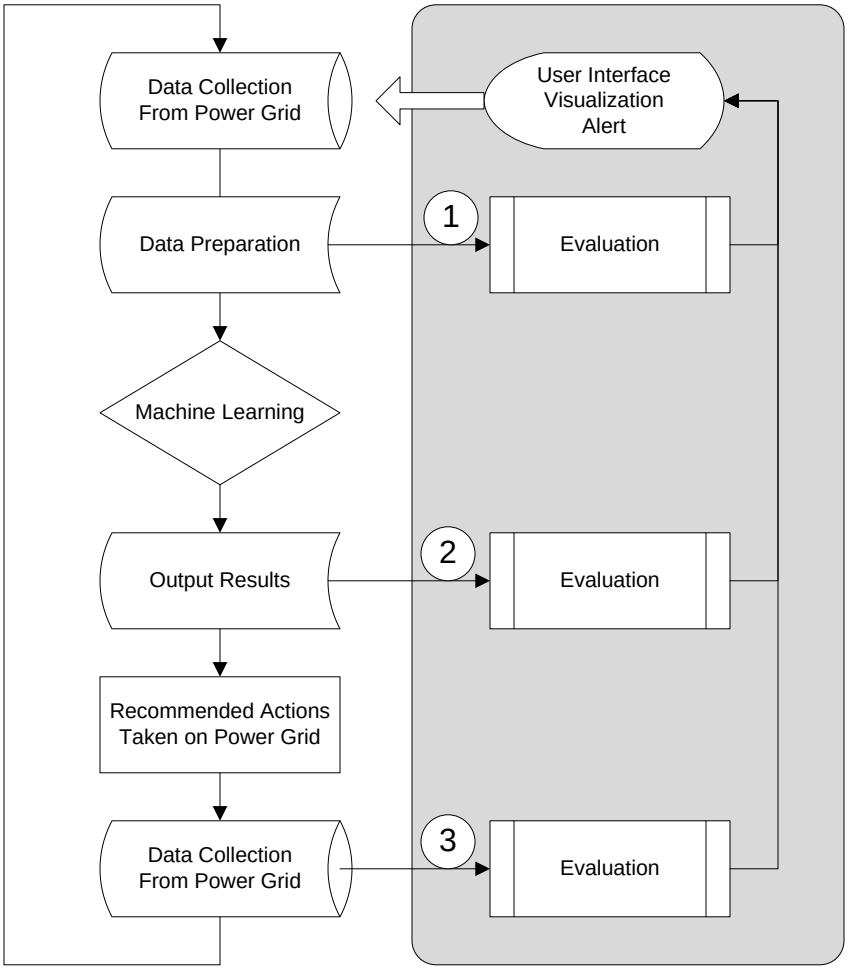


Figure 3: NOVA system design and workflow.

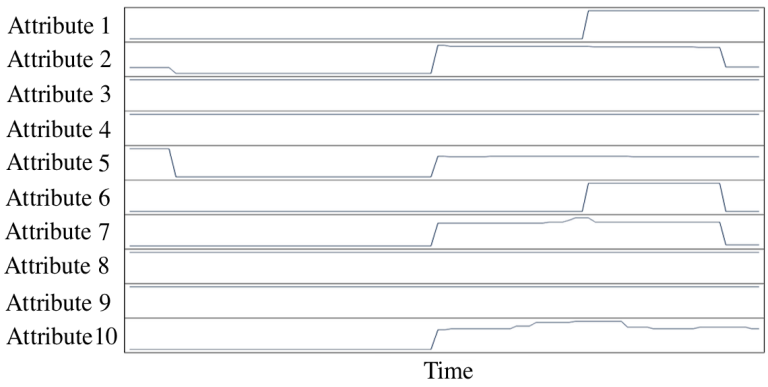


Figure 4: Sparkline graph for attributes data.

#### 5.1.3.2 Evaluation of Output Data Quality

The output is a ranked list of components ordered by their susceptibility to failures. To evaluate the output data quality, I use Receiver Operator Characteristic (*ROC*) curves, and accompanying rank statistics such as the Area Under the Curve (*AUC*). The AUC is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one [7, 12]. It is in the range of [0, 1], where an AUC of 0.5 represents a random ordering, and an AUC of close to 1.0 represents better ranking with the positive examples (*i.e.*, correctly predicted examples) at the top and the negative ones at the bottom. Figure 5 illustrates one typical ROC curve for a feeder-ranking with AUC equals 0.768. The description for each data point (*e.g.*, 17M96 (511)) stands for feeder name (*e.g.*, 17M96) and its ranking (*e.g.*, 511). When the AUC is bad, *i.e.* close to 0.5, the operator is informed that the output results are close to randomness so that the operator can use alternate factors for decision-making accordingly.

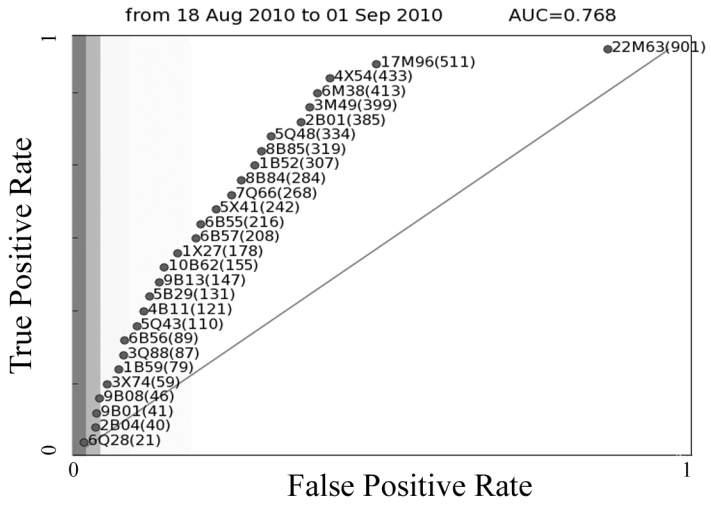


Figure 5: ROC Curve.

The ranking systems generate new models continuously, so the evaluation is presented as a time series of AUC values as shown in Figure 6. The black series in the figure shows the AUC time series of ODDS and the gray series shows the ones for MartaRank, both for the time period from May 2010 to November 2010. Our experiments show that MartaRank and ODDS feeder-ranking systems have comparable overall performance according to the AUC. The better the AUC results, the more accurate the component rankings are, which leads to better preventive maintenance results in improving system reliability.

#### 5.1.3.3 Evaluation of Reliability Improvement of the System

After the machine learning and data mining outputs ranking results, the feeders ranked with highest susceptibility to failure are usually treated with a higher priority. The final stage of the evaluation is to validate that the recommended actions are in fact leading to the expected power system improvement, *i.e.*, fewer outages and longer time between failures. For a longer time, a log(cumulative outages) versus log(time) chart is useful for seeing the changes in the time interval between failures. This graphical analysis is also called a *Duane plot*, which is a log-log plot of the cumulative number of failures versus time [16], shown in Figure 7. The changing slope of the regression lines of the

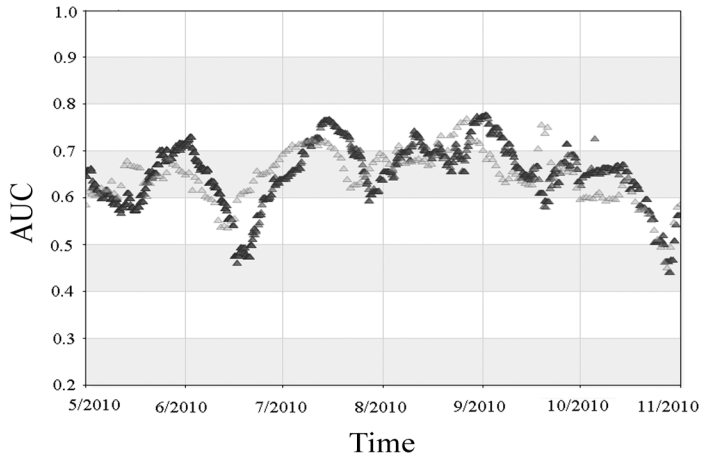


Figure 6: AUC cyclicity graph.

cumulative outages shows the improved rate of outages. If the failure rate had not changed, this log-log plot would show a straight line.

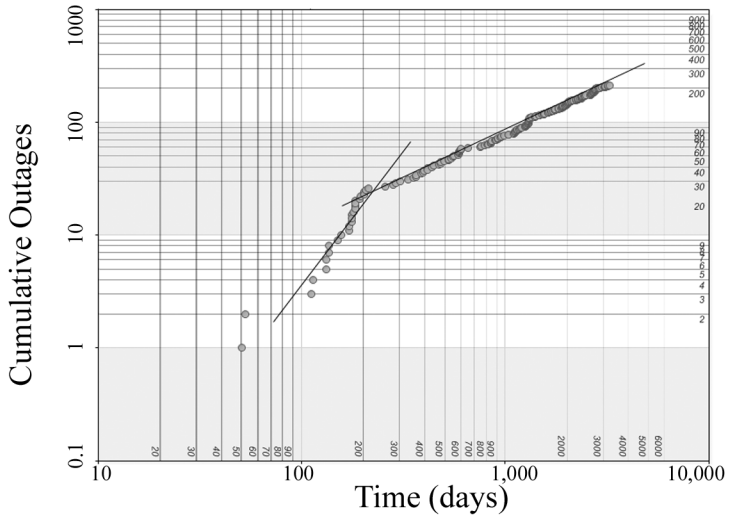


Figure 7: Cumulative outages versus time log-log chart.

To summarize the above key steps of the NOVA system as described above, Table 1 lists the evaluation targets and main techniques (*e.g.*, methods, metrics, charts) used at each evaluation stage.

### 5.1.4 Case Study

NOVA system has been implemented in evaluating two feeder-ranking systems in New York City's power grid since 2007. Some of its newer features were added from 2007 to 2010. New York City has over two thousand feeders. One experimental result I concluded from the evaluation using NOVA is the increasing MTBF (*Mean Time Between Failures*), *i.e.*, lower failure rate and better system

| Step | Evaluation target | Methods, metrics, charts |
|---|---|---|
| 1 | Input data | Sparkline graph, data checks and constraints |
| 2 | Machine learning and data mining results | ROC curve, AUC time series |
| 3 | Physical system improvements | Duane plot, MTBF, failure rate, linear regression |
| | Unified user interface | Dashboard, charts, triggers, warning messages, alert emails |

Table 1: Summary of techniques used in evaluation.

reliability, for most networks. *Mean Time Between Failures (MTBF)* is the predicted elapsed time between inherent failures of a system during operation [24]. Figure 8 illustrates MTBF time series for all the feeders in a specific network for the period from 2002 to 2009 and the linear regression. On average, the MTBF for feeders in this network are improving over time. The MTBF improvement after deployment of NOVA in 2007 was better than pre-deployment period, as the black regression line shown in the graph.

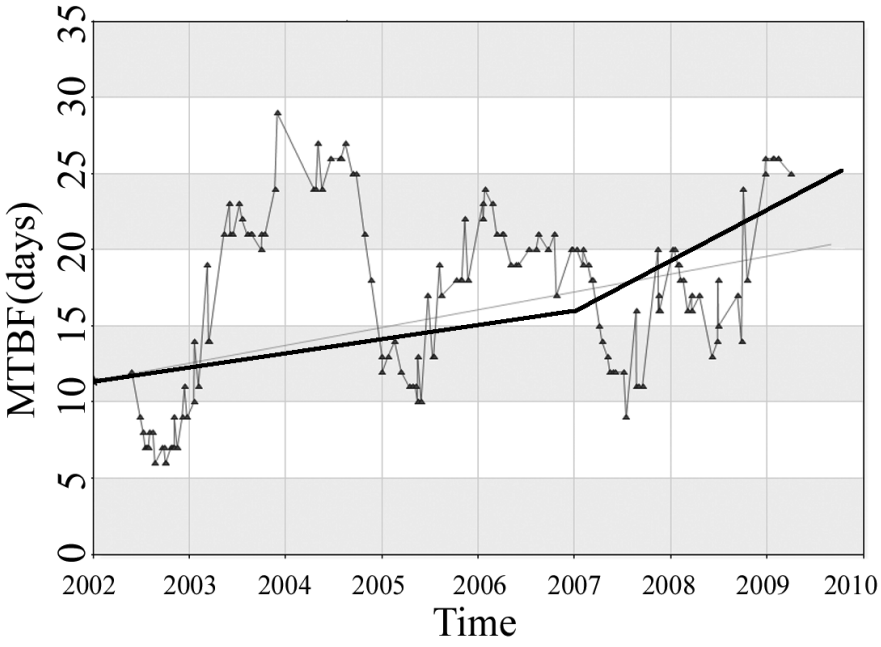


Figure 8: MTBF versus time and linear regression.

Figure 9 illustrates the MTBF differences between year 2009 and year 2002 for each network. The bars with values above zero indicate MTBF improvements. The majority of the networks saw significant increase of MTBF. More than ten percent of the approximately 2000 feeders in the city have been serviced or replaced according to their rankings. The preventive maintenance of these highly ranked error-prone feeders improved power grid system reliability.

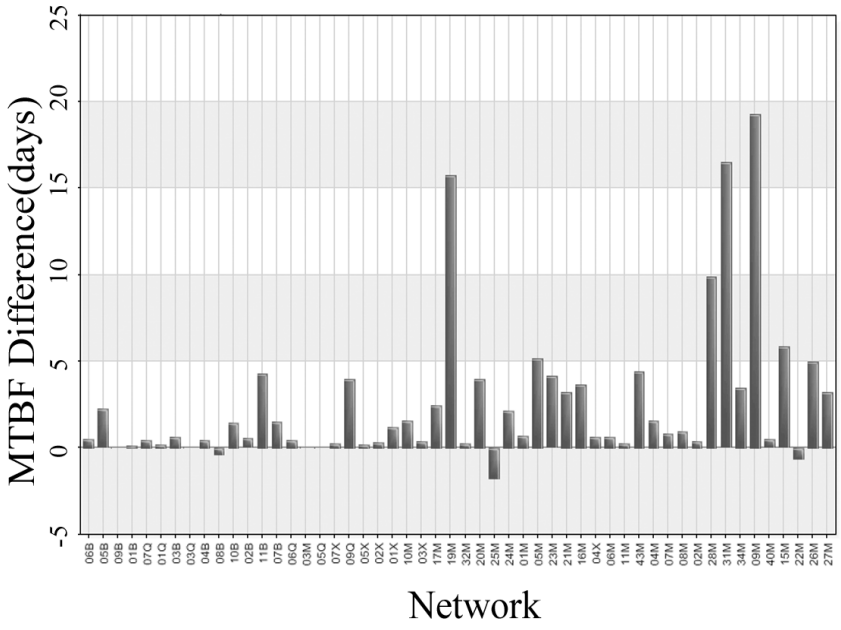


Figure 9: MTBF difference for each network.

| Year | Number of Feeder Failures |
|---|---|
| 2005 | 1612 |
| 2006 | 1547 |
| 2007 | 1431 |
| 2008 | 1239 |
| 2009 | 1009 |

Table 2: Number of feeder failures in the city.

Table 2 lists the total number of feeder failures in the city from year 2005 to year 2009. The decreasing number of feeder failures shows fewer outages of the power network.

In summary, it appears that the NOVA system contributes to the system reliability improvement of the cyber-physical system, *i.e.*, New York City's power grid, based on MTBF and number of feeder failures metrics.

## 5.2 Self-Tuning Evaluation System

The evaluation system should be able to adapt itself to the changes in the system and feedback from the operator in a live cyber-physical system through self-tuning. Now I will explain a feasibility study on self-tuning evaluation system.

### 5.2.1 Introduction

Self-tuning can be used to improve the automated online evaluation. One example is how to make the evaluation system automatically adapt to the anomaly of input data, such as the seasonality of the

temperature data. The self-tuning process is important to make sure the evaluation system can learn from the system changes or operator's feedback and self-manage the evaluation system.

### 5.2.2 Approach

A prototype approach to achieve self-tuning employs data classification, redundancy checking, and trend detection techniques. The data classification helps to determine or predict some unknown data based on historic data. The redundancy checking helps to determine if the data instance is a duplicate of some prior data. The trend detection helps to find the trend pattern for the data set. These techniques help to make self-tuning possible, which in turn self-manages the evaluation system to improve system reliability. The following subsections will describe the details for each of the techniques.

#### 5.2.2.1 Data Classification

In a live system, the input data sometimes may have missing values. To determine or predict these unknown data based on historic data helps automated online evaluation to work more accurately, which leads to better evaluation. The data classification can be solved as a supervised learning problem. By training a classification model on existing data, the missing values can be predicted. Support Vector Machines (SVM) can be used as the classifier [44, 10].

#### 5.2.2.2 Redundancy Checking

Redundant data often leads to duplicate processing and even skewed or abnormal results. It is helpful for self-tuning to effectively detect the data redundancy so that proper adjustment to evaluation system can be done accordingly, which leads to improved system reliability. For redundancy checking, I represent dataset in a vector space model (*i.e.*, term vector model), an algebraic model for representing text documents as vectors of identifiers, such as index terms [38]. I measure the similarity between two data instances based on Cosine similarity, *i.e.*, the Cosine of the angle between the two vectors that represent these two data instances, as shown in the following formula:

$$Distance_{COS}(a,b) = \frac{\sum_i a_i \times b_i}{\sqrt{\sum_i a_i^2} \times \sqrt{\sum_i b_i^2}},$$

where $a$ and $b$ represent two vectors. Its result equals 1 when the angle between two vectors is 0 (*i.e.*, two vectors are pointing in the same direction), and its result is less than 1 otherwise.

In addition to Cosine similarity, I rank all prior data instances based on their relevance to the new instance using probability distribution. Kullback-Leibler (*i.e.*, KL) divergence [11, 30] is an effective relevance metric that assumes each data instance in a high dimensional feature space is characterized by a probability distribution. KL divergence measures the dissimilarity between two probability distributions, as shown in the following formula:

$$D_{KL}(a||b) = \sum_{t \in V} P(t|M_a) log \frac{P(t|M_a)}{P(t|M_b)},$$

where $M_a$ and $M_b$ represent the probability distributions for vector $a$ and $b$ respectively. $V$ is the vocabulary of all terms and $t$ is a term in $V$. KL divergence measures how bad the probability distribution $M_a$ is at modeling $M_b$.

#### 5.2.2.3 Trend Detection

To detect data trend is important for self-tuning to adjust the evaluation system effectively, which leads to improved system reliability. For example, the change of the data trend curve may indicate overall system state change, which requires self-tuning to act on the evaluation system. One way to model the trend pattern is using Weibull distribution [35], which provides the basis for trend detection and analysis. First, historic data is used to fit the Weibull function and derive the $\lambda$ and $k$ parameters. Then for any given time $t$, the number of instances that may happen during that $t$-th time period can be estimated using the Weibull's density function $f(t)$. Similarly, the instantaneous incidence rate can be estimated using the hazard function $h(t)$. Other semiparametric approach may be used to provide similar estimation [54].

### 5.2.3 Evaluation

I evaluated the effectiveness of these techniques through a prototype system implemented using Java, Weka [49] and MATLAB [18]. I experimented the system on a bug report dataset of Apache Tomcat [34, 55]. The dataset contains 1525 data instances with two product versions (*i.e.*, Tomcat 3 and Tomcat 7), 16 different operating systems, and 16 functional software components.

#### Data Classification

In data classification experiments, I train classification model on 80% of the data and do blind-test on the remaining 20% of the data. Table 3 lists the classification results for the Tomcat version. The accuracy of the classification on testing instances is 99.02%. This means the product version in this case can be determined by the classification highly accurately.

Table 3: Classification results of products

| TP Rate | FP Rate | Precision | Recall | F-Measure | ROC Area | Class |
|---|---|---|---|---|---|---|
| 0.991 | 0.014 | 0.996 | 0.991 | 0.993 | 0.989 | tomcat 3 |
| 0.986 | 0.009 | 0.973 | 0.986 | 0.98 | 0.989 | tomcat 7 |
| 0.99 | 0.012 | 0.99 | 0.99 | 0.99 | 0.989 | Weighted Avg. |

#### Redundancy Checking

For redundancy checking, I first transform the historic training data instances and the testing data instance to vectors using the vector space model. After the $csv$ and $kld$ value for each training instance are calculated, all the training instances are then sorted in an descending order based on the $csv$ value and in an ascending order based on the $kld$ value. The data instances at the top of the ranked lists are the most similar ones to the testing instance. Table 4 lists some sample results for a given data instance #393. From the results, the instance #393 is highly likely to be a duplicate of some data instances because there exists historic data instances with $csv \geq 0.9$ and $kld \leq 2.0$ (*i.e.*, #330 and #296).

Table 4: Similarity ranking results

| bug_id | $csv$ | $kld$ |
|---|---|---|
| 330 | 0.928 | 1.940 |
| 296 | 0.917 | 0.816 |
| 228 | 0.717 | 9.868 |

#### Trend Analysis

For trend analysis, I first aggregate the historic data to compute a vector of the time ($i$-th week) and the number of data instances whose first reporting date falls in the $i$-th week. Then a result vector returns the 95% confidence intervals for the estimates of the parameters of the Weibull distribution given the historic vector data. The two-element row vector estimates the Weibull parameter $\lambda$ and $k$. The first row of the 2-by-2 matrix contains the lower bounds of the confidence intervals for the parameters, and the second row contains the upper bounds of the confidence intervals.

Table 5: Weibull parameter estimates

| Software | $\lambda$ | $\lambda_{low}$ | $\lambda_{high}$ | $k$ | $k_{low}$ | $k_{high}$ |
|---|---|---|---|---|---|---|
| Tomcat 3 | 0.3885 | 0.2280 | 0.6621 | 0.2241 | 0.2041 | 0.2461 |

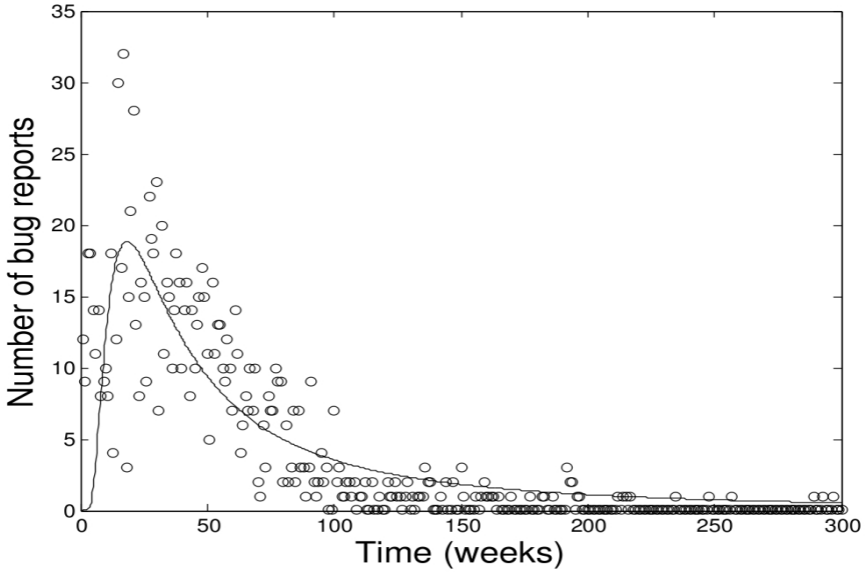


Figure 10: Weibull fit for Tomcat 3

Table 5 shows the estimates of the Weibull parameters for Apache Tomcat 3. The value of $k$ is less than 1, which indicates that the incidence rate decreases over time. The related curve fit is illustrated in Figure 10. The starting time, (*i.e.*, the 0 on the $x$-axis) is the week of August 25, 2000. The curve fit shows that the Weibull distribution closely resembles the actual data incidence trend.

In summary, my experiments show that the self-tuning techniques described above, including data classification, automatic redundancy checking and trend detection, are effective. The proposed ARIS system can employ these techniques to perform self-tuning. This self-tuning will lead to an adaptive evaluation system that works better under system changes and operator feedback, which will lead to improved system reliability. An example that these self-tuning techniques can lead to better automated online evaluation for cyber-physical systems is their application in a smart building energy management system. The data classification can be used to predict temperature and electricity load based on historic data; the redundancy checking can be used to simplify duplicate readings from smart sensors; and trend detection can be used to model the daily energy usage pattern and the seasonal energy usage shift. These self-tuning techniques would help the automated online evaluation to work better, which leads to the improved system reliability of the cyber-physical system.

# 6 Related Work

## 6.1 Cyber-Physical System Reliability

Reliability has been recognized as a critical requirement for cyber-physical systems. In Lee's paper "Cyber Physical Systems: Design Challenges", he pointed out that the expectation of reliability in cyber-physical system will only increase, and cyber-physical system will not be deployed into some mission critical applications as traffic control, automotive safety, and health care without improved reliability and predictability [28]. CPS steering group stated in its executive summary that architectures and tools are needed in order to build reliable and resilient cyber-physical systems [20]. The report also described software reliability affects the overall system reliability because replicated software can cause systematic failures that are not common in purely physical systems and todays computer systems do not allow us to distribute computer-based control in ways that preserve reliability.

For cyber-physical transportation systems, Clarke *et al.* proposed some demanding challenges of applying formal analysis technique on autonomous transportation control for cars, trains, and aircraft. Their paper listed scalable analysis with respect to complexity and dimensionality, large-scale verification architectures, dynamic networks, probabilistic effects in cyber-physical transportation as some of the main challenges [9]. Our approach does not use formal analysis technique and does not have the scalability limitation due to state-space exploration.

For reliability of electric power grid systems, Singh *et al.* concluded that the current techniques for power system reliability are insufficient because they focused mainly on the current carrying part of the power grid with some work done in the inclusion of protection systems. The paper also pointed out that the literature on the reliability of the cyber part is practically non-existent and the analysis of the power system as a cyber-physical system appears to be a challenging task because of the dimensionality and complexity issues [40]. Faza *et al.* described the use of software fault injection combined with physical failures in identifying integrated cyber-physical failure scenarios for the Smart Grid [13].

For architectural design of reliable cyber-physical system, Sha *et al.* proposed a hybrid approach that combines fault-tolerant architectures with formal verification to support the design of safe and robust cyber-physical systems [39]. La *et al.* proposed a service-based cyber-physical system based on service-oriented architecture (SOA) and mobile Internet device to achieve dynamic composition, dynamic adaptation, and high confidence [26]. Our approach does not aim to design a cyber-physical system; instead, I try to use some new techniques to ensure the reliability of cyber-physical system. One real-world constraint is that many cyber-physical systems such as power grid have expansive infrastructure already built and these legacy systems are often too hard and expensive to replace. Improving reliability of these systems entails working with the software, hardware and physical devices that have already been deployed.

Security for cyber-physical systems has also been an important research topic in the past years. [45] described some vulnerabilities and countermeasures for sensor network, a type of cyber-physical system. [46] gave a general overview on wireless sensor network security: obstacles, requirements, attacks, and defenses. An unreliable system certainly may pose more security vulnerabilities that can be exploited by malicious attackers. "A system can't be reliable if it's not secure, and to some degree, if it's not reliable, at least in a security context, it can't be secure, either [5]." While, the focus of this

thesis is not to target potential security issues and defend the possible malicious attack, instead I try to improve system reliability for the cyber-physical system so that the system can run properly with or without security attack.

## 6.2 Automated Online Evaluation

For NOVA system, I have given examples each of the three steps of online evaluation, using NYC power grid data. Depending on specific data and operational goals, there may be many ways to perform one of the three evaluations; the key point is that all of these three types of evaluation must be present. In machine learning and data mining, only the second type of evaluation is typically considered (step 2), and even that evaluation is mainly considered in static settings (without the element of time).

Langley's seminal paper "Machine Learning as an Experimental Science" made empirical study an indispensable aspect of machine learning research [27]. Since that time, many challenges in experimental machine learning have been identified. For instance, a more recent survey of Japkowicz reviewed shortcomings in current evaluation methods [23]. Through using NOVA on the New York City power grid, I have also been able to identify new challenges (*e.g.*, the AUC cyclicity challenge). In machine learning, the goal is often to optimize the criteria used for evaluation. NOVA suggests a much more ambitious set of evaluations than what are usually performed in machine learning and data mining experiments, potentially leading to a much broader way to consider and design machine learning systems, and hopefully leading to improvements in power grid operations.

Murphy et al. have done research on verification of machine learning programs from software testing perspective [31]. Our approach does not verify the internal correctness of the machine learning and data mining component. NOVA treats the machine learning and data mining process as a black-box module and conducts evaluation according to its external specifications. This leaves the quality assurance of the machine learning and data mining software module to the machine learning researchers and software developers or testers.

## 6.3 Data Quality Analysis Techniques

Data mining finds its increased adoption and application in software engineering in recent years. [21] described the concept of software intelligence and the future of mining software engineering data. [56] presented a general overview of data mining for software engineering and described an example of duplicate bug detection using vector space-based similarity. [47] also described an approach to detect duplicate bug reports using both natural language and execution information. Our redundancy checking engine uses both probability distribution-based KL divergence and vector space-based Cosine similarity ranking, instead of only vector space-based similarity. Furthermore, our approach provides a similarity ranking list that can be used for search, instead of only Yes and No on duplication check. [17] presented text mining of bug reports to identify security issues. Their work aims to identify security problems such as buffer overflow through mining the bug reports. Their purpose and techniques are different from our approach.

## 6.4 Self-Tuning and Autonomic Computing

Some prior research has been done on self-tuning. Sullivan demonstrated in his Ph.D. thesis that probabilistic reasoning and decision-making techniques can be used as the foundation of an effective, automated approach to software tuning [41]. Self-tuning is an aspect of autonomic computing, which is an approach to self-managed computing systems with a minimum of human interference [22] and refers to the self-managing characteristics of distributed computing resources, adapting to unpredictable changes whilst hiding intrinsic complexity to operators and users [48]. Kaiser *et al.* have retrofitted autonomic computing onto legacy systems, externally, without any need to understand or modify the code, and in many cases even when it is impossible to recompile [25, 33].

# 7 Research Plan and Schedule

## 7.1 Development Tasks

The main development task is to develop ARIS system [52, 53] based on the NOVA system described in 5.1 and the self-tuning techniques described in 5.2. NOVA has been proved to be effective and useful in evaluating online machine learning and data mining systems used in New York City's power grid. Although it is a good example of automated online evaluation approach, it was custom designed to evaluate a specific type of software, *i.e.*, machine learning software, used in cyber-physical system, *i.e.*, power grid. Based on the work on NOVA system and the self-tuning techniques, I will further explore a general-purpose ARIS system for automated online evaluation, and use it to improve system reliability for cyber-physical system.

## 7.2 Experiments and Methodology

The experiments of ARIS system will be conducted not only in a lab environment, but also in some real-world cyber-physical systems. Some quality assurance measurements and metrics I will use to quantitatively measure the system reliability improvement include:

- *Mean Time Between Failures (MTBF)* is the predicted elapsed time between inherent failures of a system during operation [24].
- *Availability* is the measurement of the fraction of time system is really available for use. It takes repair and restart times into account and is relevant for non-stop continuously running systems.
- *Rate of Fault Occurrence* reflects failure rate in the system. It is useful when system has to process a large number of similar requests that are relatively frequent.
- *Probability of Failure on Demand* is the probability system will fail when a service request is made. It is useful when requests are made on an intermittent or infrequent basis.
- *Power-On Hours (POH)* is the length of time (in hours), which electrical power is applied to a device.

### 7.2.1 Controlled Experiment

The controlled experiments will be based on lab or benchmark environment. The experimental data will be the data that are available to general public via Internet download and the supporting software for the environment will be the ones that are commonly used. These controlled experiments are important for proving the hypotheses. They are to prove first, second and third hypotheses described in 3.2. The initial experiments in the lab are good for initial proof-of-concept prior to real-world deployment and experiments.

In the controlled experiment, the input and output data anomaly can be simulated using fault injection, a software testing technique for improving the coverage of a test by introducing faults to test code paths [4, 8, 51, 50]. Some data randomization can be used to simulate uncontrolled data in the real-world environment. The effect of the output results on the cyber-physical system can be simulated, possibly using some emulator that can represent the physical part of the cyber-physical system.

In any experiment, two independent cyber-physical systems will be used in parallel: one with the ARIS implemented and the other one without ARIS. Both systems will be supplied faulty input data. And then measurement and validation will be done to compare both systems' reliability to see if they can continue reliable execution without problem. The faulty conditions in output data and physical system effect will also be simulated for evaluating the system reliability of the two cyber-physical systems.

For controlled experiments on the internal functions of the ARIS (*i.e.*, data quality analysis using computational intelligence and self-tuning, as shown in Figure 2), the ARIS is compared with some alternate methods. For example, to evaluate input data quality in terms of data range and distribution, ARIS can be compared with the rule-based system to see which one can produce better analysis results. This type of comparative studies may also show the advancement of the state of the art research by the proposed approach.

### 7.2.2 Real-World Experiment

Another important set of experiments will be conducted in some real-world environments where unpredictable conditions may happen. This requirement is important and necessary in supporting the proposed approach workable because cyber-physical systems will not be operating in a controlled environment, and must be robust to unexpected conditions and adaptable to subsystem failures [20]. The real-world experiments can provide a more realistic evaluation and analysis of the new techniques and the approach on their effectiveness and efficiency in improving system reliability for cyber-physical systems. The real-world experiments are to support the fourth efficiency hypothesis described in 3.2.

For the real-world experiments, I will deploy and experiment the prototype systems into two major cyber-physical systems: electric power grid and building energy control system.

- Electric power grid is a type of critical infrastructure, which is migrating to smart grid with more and more computing and communication components. I will apply the new ARIS system on some power grid systems to prove that the proposed approach can in fact improve system reliability for cyber-physical systems and it does not incur too much extra cost.
- Building energy control system is another important type of cyber-physical systems that are becoming increasingly smart and complex. The building energy control systems normally collect and use sensor and other utilities usage data with the help of *Building Management System (BMS)* software. I will try to deploy and experiment the new ARIS system on some smart building cyber-physical systems and prove the proposal techniques and approach are effective and efficiency in improving system reliability.

## 7.3 Schedule

Table 6 shows my plan for completion of the research.

| Completion Date | Work | Status |
|---|---|---|
| Sep. 2009 | Conduct literature review on concurrency testing | completed |
| Nov. 2009 | Conduct studies on common concurrency bug patterns | completed |
| Jan. 2010 | Conduct literature review on mutation testing | completed |
| Feb. 2010 | Conduct feasibility study on second-order concurrency mutants | completed |
| Mar. 2010 | Complete implementation and demo of BUGGEN | completed |
| Apr. 2010 | **Complete CS tech report on concurrency mutation operators** | deposited |
| May. 2010 | Write and submit BUGGEN paper to ISSRE | rejected |
| Aug. 2010 | Conduct literature review on evaluation of machine learning | completed |
| Sep. 2010 | Determine scope of NOVA system development | completed |
| Oct. 2010 | **Co-author TPAMI paper "ML in New York City Power Grid"** | accepted |
| Nov. 2010 | Research and experiment on statistical reliability estimation | completed |
| Dec. 2010 | Complete NOVA implementation and empirical study | completed |
| Jan. 2011 | Write and submit NOVA paper to AAAI | rejected |
| Feb. 2011 | **Write and submit paper on reliability estimation paper to IEEE** | accepted |
| Feb. 2011 | Conduct feasibility study and experiments on BUGMINER | completed |
| Mar. 2011 | **Rewrite and submit BUGGEN paper to SEKE** | accepted |
| Mar. 2011 | **Write and submit BUGMINER paper to SEKE** | accepted |
| Apr. 2011 | **Rewrite and submit NOVA paper to ICML workshop** | accepted |
| May. 2011 | **Write and submit extended NOVA paper to KDD workshop** | accepted |
| May. 2011 | Write thesis proposal draft | completed |
| Jun. 2011 | Revise thesis proposal and prepare for presentation | completed |
| Jul. 2011 | Revise thesis proposal and present the proposal | completed |
| Aug. 2011 | Revise and complete thesis proposal | |
| Aug. 2011 | Extend data quality analysis techniques | |
| Sep. 2011 | Research and develop ARIS system | |
| Sep. 2011 | **Write and submit comprehensive paper to appropriate conference** | |
| Oct. 2011 | Further development and experiment of ARIS system | |
| Oct. 2011 | **Write and submit additional paper to appropriate conference** | |
| Nov. 2011 | Implementation of new techniques on real-world systems | |
| Dec. 2011 | Present the recent research progress | |
| Jan. 2012 | **Write and submit empirical study paper to appropriate conference** | |
| Feb. 2011 | Demonstrate applicability to other domains | |
| Mar. 2012 | Extend evaluation techniques for large-scale real-world systems | |
| Mar. 2012 | **Write and submit paper to appropriate venue** | |
| Apr. 2012 | Write thesis | |
| May. 2012 | Write thesis | |
| June. 2012 | Write thesis | |
| July. 2012 | Complete thesis and schedule thesis defense | |
| Aug. 2012 | Defend thesis | |
| | ** bold fonts indicate paper or tech report submission* | |

Table 6: Plan for completion of research

# 8 Expected Contributions

The expected contributions are listed as follows:

- A system evaluation approach named automated online evaluation that is able to improve system reliability for cyber-physical systems in the domain of interest as described in Section 2. The approach employs data quality analysis and self-tuning. It enables online reliability assurance of the deployed systems that are not possible to perform robust tests prior to actual deployment because of physical and cost constraints.
- A prototype implementation of the approach, *i.e.*, ARIS system, and experimental demonstration of the approach using ARIS in some controlled experiments as well as some real-world environment.
- A new technique of data quality analysis using computational intelligence and its application in this type of evaluation system for cyber-physical system.
- A new demonstration of applying self-tuning in this type of evaluation system for cyber-physical system.
- A preliminary study on applicability of the approach on other domains in order to show that the approach can be potentially adapted and extended for use in improving system reliability for much broader range of large-scale real-world online cyber-physical systems.

# 9 Conclusion

In this proposal, I presented a system evaluation approach named automated online evaluation that employs data quality analysis using computational intelligence and self-tuning techniques for improving system reliability for cyber-physical systems that meet following criteria: processing large amount of data; employing software as a system component; running online continuously; having operator-in-the-loop. I then presented ARIS system architecture for implementation of such an approach. I further described some feasibility studies: first, a date quality analysis system named NOVA that is able to evaluate online machine learning and data mining applied in power grid cyber-physical system to improve system reliability; second, a feasibility study on the effectiveness of some self-tuning techniques, including data classification, redundancy checking and trend detection. I further laid out my research and development plan, and described experiments of the proposed approach for proving the stated hypotheses.

# 10 Future Work

One potential future work is to extend and apply automated online evaluation, data quality analysis using computational intelligence, and self-tuning techniques onto other complex systems such as cloud computing systems to validate the approach's applicability and effectiveness.

Another potential future work is to further offload the work by human system operators, thus closing the feedback loop, and employ some automated software processes or robots that can take actions as human operators on the cyber-physical systems. In this way, the whole cyber-physical system can be fully autonomic with self-managing and self-configuring.

# 11 Acknowledgments

I would like to thank my advisor Prof. Gail Kaiser for her support and guidance along my PhD study. I also want to thank my thesis committee (Prof. Gail Kaiser, Dr. Roger Anderson, and Dr. Christian Murphy) for their valuable suggestions and help. Furthermore, I would like to thank Phil Gross, Cynthia Rudin and the members of the Programming Systems Laboratory and the Center for Computational Learning Systems at Columbia University for their assistance in my research.

# References


[1] S. M. Amin. U.s. electrical grid gets less reliable. *IEEE Spectrum*, page 80, January 2011.

[2] R. N. Anderson. Building the energy internet. *Economist*, March 11th 2004.

[3] ANSI/IEEE. *Standard Glossary of Software Engineering Terminology*. ANSI/IEEE, 1991.

[4] J. Arlat, M. Aguera, L. Amat, Y. Crouzet, J.-C. Fabre, J.-C. Laprie, E. Martins, and D. Powell. Fault injection for dependability validation: a methodology and some applications. *Software Engineering, IEEE Transactions on*, 16(2):166 –182, feb 1990.

[5] S. Barnum, S. Sastry, and J. A. Stankovic. Roundtable: Reliability of embedded and cyber-physical systems. *Security Privacy, IEEE*, 8(5):27 –32, sept.-oct. 2010.

[6] H. Becker and M. Arias. Real-time ranking with concept drift using expert advice. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 86–94, New York, NY, USA, 2007. ACM.

[7] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, July 1997.

[8] J. V. Carreira, D. Costa, and J. G. Silva. Fault injection spot-checks computer system dependability. *Spectrum, IEEE*, 36(8):50 –55, aug 1999.

[9] E. M. Clarke, B. Krogh, A. Platzer, and R. Rajkumar. Analysis and verification challenges for cyber-physical transportation systems. In *National Workshop for Research on High-confidence Transportation Cyber-Physical Systems: Automotive, Aviation & Rail*, Washington, DC, November 2008.

[10] C. Cortes and V. Vapnik. Support-vector networks. In *Machine Learning*, page 20. Springer, 1995.

[11] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[12] T. Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27:861–874, 2006.

[13] A. Faza, S. Sedigh, and B. McMillin. Integrated cyber-physical fault injection for reliability analysis of the smart grid. In Erwin Schoitsch, editor, *Computer Safety, Reliability, and Security*, volume 6351 of *Lecture Notes in Computer Science*, pages 277–290. Springer Berlin / Heidelberg, 2010.

[14] Federal Smart Grid Task Force. Smart grid basics, 2010. Available at http://www.smartgrid.gov/basics.

[15] U.S.-Canada Power System Outage Task Force. Interim report: Causes of the august 14th blackout in the united states and canada, 2003.

[16] O. Gaudoin, B. Yang, and M. Xie. A simple goodness-of-fit test for the power-law process, based on the duane plot. *IEEE Transactions on Reliability*, 52(1):69–74, March 2003.

[17] M. Gegick, P. Rotella, and T. Xie. Identifying security bug reports via text mining: An industrial case study. In *Proceedings of the 7th IEEE Working Conference on Mining Software Repositories (MSR)*, pages 11–20, Cape Town, May 2010.

[18] A. Gilat. *MATLAB: An Introduction with Applications 2nd Edition*. John Wiley & Sons., July 2004.

[19] P. Gross, A. Salleb-Aouissi, H. Dutta, and A. Boulanger. Ranking electrical feeders of the new york power grid. In *Proceedings of the International Conference on Machine Learning and Applications (ICMLA)*, pages 725–730, 2009.

[20] CPS Steering Group. Cyber-physical systems executive summary, March 2008.

[21] A. E. Hassan and T. Xie. Software intelligence: Future of mining software engineering data. In *Proceedings of the FSE/SDP Workshop on the Future of Software Engineering Research (FoSER 2010)*, pages 161–166, Santa Fe, NM, November 2010.

[22] IBM. Autonomic computing, 2011. available at http://www.research.ibm.com/autonomic/.

[23] N. Japkowicz. Why question machine learning evaluation methods (an illustrative review of the shortcomings of current methods). In *2006 AAAI Workshop Evaluation Method for Machine Learning*. AAAI, 2006.

[24] J. V. Jones. *Integrated Logistics Support Handbook, 2nd Edition*. McGraw-Hill Professional, 1998.

[25] G. Kaiser. Autonomizing legacy systems. In *2002 IBM Almaden Institute Symposium on Autonomic Computing*, April 2001.

[26] H. J. La and S. D. Kim. A service-based approach to designing cyber physical systems. In *Proceedings of the 2010 IEEE/ACIS 9th International Conference on Computer and Information Science*, ICIS '10, pages 895–900, Washington, DC, USA, 2010. IEEE Computer Society.

[27] P. Langley. Machine learning as an experimental science. *Machine Learning*, 3(1):5–8, 1988.

[28] Edward A. Lee. Cyber physical systems: Design challenges. In *International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC)*, May 2008. Invited Paper.

[29] P. M. Long and R. A. Servedio. Martingale boosting. In *Eighteenth Annual Conference on Computational Learning Theory (COLT)*, pages 79–94, 2005.

[30] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[31] C. Murphy, G. E. Kaiser, L. Hu, and L. Wu. Properties of machine learning applications for use in metamorphic testing. In *Proceedings of the 20th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, pages 867–872, July 2008.

[32] NITRD. Winning the future with science and technology for 21st century smart systems. Technical report, Networking and Information Technology Research and Development (NITRD) Program, 2010.

[33] J. Parekh, G. Kaiser, P. Gross, and G. Valetto. Retrofitting autonomic capabilities onto legacy systems. *Journal of Cluster Computing*, 9(2):141–159, April 2006.

[34] Apache Project. http://issues.apache.org/bugzilla/, 2011.

[35] S. E. Rigdon and A. P. Basu. Estimating the intensity function of a weibull process at the current time: Failure truncated case. In *Journal of Statistical Computation and Simulation (JSCS)*, volume 30, pages 17–38, 1988.

[36] C. Rudin, R J. Passonneau, A. Radeva, H. Dutta, S. Ierome, and D. Isaac. A process for predicting manhole events in manhattan. *Machine Learning*, 80(1):1–31, 2010.

[37] C. Rudin, D. Waltz, R. N. Anderson, A. Boulanger, A. Salleb-Aouissi, M. Chow, H. Dutta, P. Gross, B. Huang, S. Ierome, D. Isaac, A. Kressner, R. J. Passonneau, A. Radeva, and L. Wu. Machine learning for the new york city power grid. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, May 2011.

[38] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[39] L. Sha and J. Meseguer. Design of complex cyber physical systems with formalized architectural patterns. In Martin Wirsing, Jean-Pierre Banâtre, Matthias Hölzl, and Axel Rauschmayer, editors, *Software-Intensive Systems and New Computing Paradigms*, pages 92–100. Springer-Verlag, Berlin, Heidelberg, 2008.

[40] C. Singh and A. Sprintson. Reliability assurance of cyber-physical power systems. In *2010 IEEE Power and Energy Society General Meeting*, pages 1 –6, july 2010.

[41] D. G. Sullivan. Using probabilistic reasoning to automate software tuning. Technical report, Harvard University, September 2003.

[42] TechTarget. http://www.searchdatamanagement.com, 2011.

[43] E. Tufte. *Beautiful Evidence*. Graphics Press, 2006.

[44] V. N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, New York, 1995.

[45] A. Vaseashta and S. Vaseashta. A survey of sensor network security. *Sensors and Transducers*, 94(7):91–102, July 2008.

[46] J. P. Walters, Z. Liang, W. Shi, and V. Chaudhary. Wireless sensor network security: A survey, in book chapter of security. In *in Distributed, Grid, and Pervasive Computing, Yang Xiao (Eds*, pages 0–849. CRC Press, 2007.

[47] X. Wang, L. Zhang, T. Xie, J. Anvik, and J. Sun. An approach to detecting duplicate bug reports using natural language and execution information. In *Proceedings of the 30th International Conference on Software Engineering (ICSE)*, pages 461–470. ACM Press, 2008.

[48] Wikipedia. http://www.wikipedia.org, 2011.

[49] I. H. Witten, E. Frank, L. Trigg, M. Hall, G. Holmes, and S. J. Cunningham. Weka: Practical machine learning tools and techniques with java implementations. In *Proceedings of the ICONIP/ANZIIS/ANNES'99 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems*, pages 192–196, 1999.

[50] L. Wu and G. Kaiser. Empirical study of concurrency mutation operators for java. Technical report, Department of Computer Science, Columbia University, 2010.

[51] L. Wu and G. Kaiser. Constructing subtle concurrency bugs using synchronization-centric second-order mutation operators. In *Proceedings of the 23th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, July 2011.

[52] L. Wu, G. Kaiser, C. Rudin, and R. Anderson. Data quality assurance and performance measurement of data mining for preventive maintenance of power grid. In *Proceedings of the 17th ACM SIGKDD Workshop on Data Mining for Service and Maintenance*, August 2011.

[53] L. Wu, G. Kaiser, C. Rudin, D. Waltz, R. Anderson, A. Boulanger, A. Salleb-Aouissi, H. Dutta, and M. Pooleery. Evaluating machine learning for improving power grid reliability. In *ICML 2011 Workshop on Machine Learning for Global Challenges*, July 2011.

[54] L. Wu, T. Teräväinen, G. Kaiser, R. Anderson, A. Boulanger, and C. Rudin. Estimation of system reliability using a semiparametric model. In *Proceedings of the IEEE EnergyTech 2011 (EnergyTech'11)*, May 2011.

[55] L. Wu, B. Xie, G. Kaiser, and R. Passonneau. Bugminer: Software reliability analysis via data mining of bug reports. In *Proceedings of the 23th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, July 2011.

[56] T. Xie, S. Thummalapenta, D. Lo, and C. Liu. Data mining for software engineering. *Computer*, 42(8):55–62, 2009.