학위 논문 제목	A Power Efficient Voltage Up-Converter for Embedded EEPROM Application
학과	전자전기컴퓨터공학과
학번	
이름	
지도교수	

1. 논문 요약 (Abstract)

본 논문에서는 극 저전력을 필요로 하는 모바일 기기나 RFID tag등에서 적용될 수 있는 power efficient voltage up-converter를 제안한다. Charge pump, VPP detector, Level shifter로 구성된 voltage up-converter는 그 효율이 매우 중요하기 때문에 본 논문은 효율이 좋지 않은 기존의 회로들이 가지는 문제점과 그에 따른 해결책을 제시한다. VPP 전압이 올라갈수록 효율이 떨어지는 NMOS type Dickson charge pump의 사용을 배제하고 CMOS type의 charge pump를 적용, stage, pumping capacitor, IPP등을 고려하여 최적화 하였다. 또한 기존의 저항 VPP divider를 가지는 VPP detector 대신 RC coupled VPP divider를 가지는 VPP detector를 제안하고, 별도의 전력소모 없이 coupling capacitor 이용하여 히스테리시스를 가지게 하였다. VPP 전압이 올라갈수록 많은 양의 short circuit current를 발생하는 level shifter는 bootstrapped PMOS를 이용하여 level shifter의 전력소모 뿐만 아니라 delay 특성을 개선시켜, VPP 12V에서 제안된 level shifter는 기존의 level shifter보다 PDP 특성이 3.7배 개선된 것으로 나타났다. 본 논문에서 제안하는 power efficient voltage up-converter는 0.18um CMOS 공정에서 설계되었으며, RFID tag에 쓰이는 EEPROM IP에 적용되었다.

주제어 : voltage up-converter , charge pump , VPP detector ,level shifter , low power

2. 논문 기여도 (Contribution)

기존 연구	학위 논문과의 차이점
[1]	M. M. Ahmadi가 제안하는 논문은 기존의 NMOS type Dickson's charge pump가 가지고 있는 문제점들을 개선하여 CMOS tpye의 charge pump를 소개하고 있다. 그러나 극 저전력을 필요로 하는 RFID tag 등에 그대로 적용하기 힘들기 때문에 IPP, pumping capacitor, pumping stage등을 고려, 분석하여 charge pump의 전력 효율을 최적화 하였다.
[2]	Qadeer A. Khan이 제안하는 multi-voltage level shifter는 VPP 전압이 올라갈수록 delay, 전력소모 측면에서 불리한 점을 안고 있다. 본 논문에서는 bootstrapped PMOS를 이용하여 delay를 줄이고 전력소모를 크게 낮춘 level shifter를 소개 하였다.

[1] M. M. Ahmadi and G. Jullien, "A New CMOS Charge Pump for Low Voltage Applications," *IEEE International Symposium on Circuits and Systems, vol. 5, pp. 426*1-4264, 23-26, May 2005.
[2] Khan, Q.A., Wadhwa, S.K., Misri, K., "A single supply level shifter for multi-voltage systems", VLSI Design, 2006.

학술적/기술적 기여도	
정량적 측면	본 논문에서 제안하고 있는 voltage-up converter는 RFID tag용 EEPROM IP에 적용 되었다. 가장 전력소모가 많은 charge pump 블록에서는 전력 효율을 최적화 하여 level shifter 블록에서 발생하는 short circuit current를 제거하여 memory의 write power consumption을 20.2uW로 줄였다.
정성적 측면	Voltage-up converter를 이루고 있는 세 블록중 charge pump 블록에서는 부하 전류(IPP)에 따라 pumping cap size, pumping stage를 최적화 하여 효율을 높였다. VPP detector블록 에서는 R-C coupled divider를 이용하여 standby 전류를 줄이고 안정적이고 빠른 VPP regulating이 일어나도록 하였다. Level shifter 기존의 level shifter가 안고있는 short circuit current를 제거함으로서 그 성능을 크게 향상 시켜 그 내용을 IEEE trans. circuits and systems society로부터 인정받았다.

3. 논문 발표 실적 (Publications and Awards)

1) 국제 저널 실적

Jong-Min Baek, Jung-Hoon Chun and Kee-Won Kwon, "A Power Efficient Voltage-Up Converter for Embedded EEPROM Aplication", *IEEE trans. circuits and systems--II: Express Briefs, Volume 57, Issue 6,* 2010

2) 국내 저널 실적

3) 국제 학회 실적

Won-Ji Lee, Kyoung-Su Lee, Jong-Min Baek, Jung-Hyun Song, Jae-Chul Park and Kee-Won Kwon, "A Power-Efficient Voltage Up-Converting Circuit System", The 23rd *International Technical Conferenceon Circuits/Systems, Computersand Communications* (ITC-CSCC2008) July2008, Shimonoseki, Japan.

Jung-Hyun Song, Kyoung-Su Lee, Won-Ji Lee, Jong-Min Baek, Kee-Won Kwon, "A Highly Reliable, Low Power Non-Volatile Memory Cell Utilizing Localized Voltage Boosting" *2008International Conferenceon Electronics, Information,and Communication*, June 2008, Tashkent, Uzbekistan

4) 국내 학회 실적

Kyoung-Su Lee, Won-Ji Lee, Jong-Min Baek, Jung-Hyun Song, Ji-Hong Kim, and Kee-Won Kwon,, "Low Voltage Floating Gate Type Memory IP Design" *SoC Conference*, May 2008, Seoul Korea, pp.192-195

5) 특허 및 수상 실적

Kyoung-Su Lee, Won-Ji Lee, Jong-Min Baek, Jung-Hyun Song, Ji-Hong Kim, and Kee-Won Kwon,, "Low Voltage Floating Gate Type Memory IP Design" *SoC Conference*, May2008, Seoul Korea, pp.192-195(Outstanding Paper Award)

6) 기타 실적

Master’s Thesis

Fault Diagnosis and Prognosis Model for Self-Healing

Gilsan Jang

Department of Electrical and Computer Engineering

The Graduate School

Sungkyunkwan University

Master’ s Thesis

Fault Diagnosis and Prognosis Model for Self-Healing

Gilsan Jang

Department of Electrical and Computer Engineering
The Graduate School
Sungkyunkwan University

Fault Diagnosis and Prognosis Model for Self-Healing

Gilsan Jang

A Master's Thesis Submitted to the Department of Electrical and Computer Engineering and the Graduate School of Sungkyunkwan University in partial fulfillment of the requirements for the degree of Master of Science in Engineering

October 2020

Supervised by
Gil dong Hong
Major Advisor

This certifies that the Master's Thesis
of Gilsan Jang is approved.

심사위원장 signature

Committee Chairman : Bu– jang Kim

심사위원 signature

Committee Member : Cha– jang Lee

지도교수 signature

Major Advisor : Gil–dong Hong

The Graduate School
Sungkyunkwan University
December 2020

Table of Contents

List of Tables

List of Figures

Abstract

Fault Diagnosis and Prognosis Model for Self-Healing

The increasing complexity of distributed computing systems which involve a great number of computing devices, has made a challenge of managing and controlling such systems in an automated way. One obvious characteristic of autonomic computing called self-healing is to accurately recognize system real-time behavioral information and external environment knowledge and reflect to abnormities via corrective repair strategies, which reduces human intervention. Many Artificial Intelligent techniques including machine learning are widely used in field of fault management which supports problem determination. In this paper, we propose an approach to fault management for self-healing based on probabilistic dependency analysis to provide not only fault diagnosis but also fault prognosis functions. We use Bayesian network algorithm to transform a complex system into a compact probabilistic dependency model with various relative factors, providing inductive and deductive inferences on the network. We also propose an approach to extracting relevant nodes as an ordering node list in support of learning a network, which enhances learning efficiency and reduces learning time. Using the proposed approach enables us quickly and accurately to localize root cause of faults and to predict potential problems based on given observations. In order to estimate the efficiency and accuracy, we give an experimental demonstration based on analyzing the raw system health measurements and evaluations on various comparisons, focusing on system reliability.

Keywords: Fault Diagnosis, Prognosis, Preprocessing, Probabilistic Dependency Analysis, Self-Healing

1. Introduction

With the inherent nature of the Ubiquitous computing system, complexity appears frequently in all places as the increasing growth in size and flexibility. More requirements in distributed system make more complexities created [1], which brings much more burdens and hardness for administrators to handle abnormities and maintain high system reliability. It is very important to system manager for managing the computer system and to users for running their applications successfully. In such a system, for no obvious reason, even with reliable hardware and software there are always faults which have to be repaired otherwise computer systems sometimes crash and fail to deliver the services that have been requested. Thereby, the issue of fault management emerged, which led to the appearance of autonomic computing or self-managing systems, including self-healing, self-configuring, self-protecting, and self-optimizing [2]. One of the essential characteristics of self-healing is that the system can recover from faults on its own initiative instead of system administrators' direct handling, in order to provide higher quality of services without interruption, which is directly related to system capability and reliability. System failures are inevitable, but the disruption caused by failure can be minimized if the system can be repaired quickly. In order for that to happen, root cause analysis and proactive problem prediction based on probabilistic dependency an analysis are required in large distributed computer systems, which are challenging tasks that require rapid and accurate inferences from potentially huge data volumes [3].

From a fault management point of view, self-healing can be regarded as consisting of fault detection, fault diagnosis and fault-repair, a series of processes corresponding to monitoring, analysis, plan, and execution that perform self-healing capability. When the system fails to deliver correct services as specified, it is difficult to localize which part of the system is the source of the problem with uncertain and unobservable knowledge of environment in which the system is used. Existing techniques such as rule-based or case-based algorithms are not competent. In some cases, it is not popular in uncertain domain with missing information and inferring with low accuracy, and it becomes large size as increasing states [4]. Many fault management techniques rely on an explicit fault propagation model (FPM) [5] or cause-effect causality model representing either causal relationships among events or dependencies among communication entities, and use deterministic or probabilistic dependency analysis for inferences.

Problem localization is a process of deducing the exact root cause of problems based on a set of observed information. Clearly, fault diagnosis and prognosis, which are central aspects of fault management, are critical to designing an effective self-healing system, by which the system determines and solves problems automatically. In this paper, we propose an approach to fault management using Bayesian Network to transform uncertain and complex system to a compact model, which enable us both to localize the root cause of problems and predict potential problems under given observations in advance via probabilistic dependency analysis. However, the quality of modeling a Bayesian network is quite related to the results of inferences, especially in the case of that having more factors in a model. In order to model accurate and efficient Bayesian model with less degrading the quality of learning, ordering parameters are extracted from preprocessing course, which includes two phases: parameter selection and parameter ordering. The proposed approach is possible to conduct automated system management in complex distributed system and improve system reliability. We present a demonstration experimental evaluation of our work through illustration results of performance measurements. Various comparisons and evaluations prove that our proposed approach is effective on problem localization for self-healing and can achieve significant cost savings.

The rest of this paper is organized as follows. First, we first present the overview of autonomic computing and existing fault management techniques on which our approach to problem localization is based. Second, we give a detail description of the proposed approach to fault management for self-healing, mainly focusing on the improved process and method of modeling Bayesian network. Third, we examine a straightforward application of learning network and discuss how to implement problem localization under the proposed approach. In the last section, we conclude this paper and provide directions for future research.

2. Related works

Self-managing system tasks in Ubiquitous environment such as real-time fault localization and problem diagnosis, call for higher levels of automation.

Although there are already some research efforts on real-time fault management fields, which conducts large scope of Artificial Intelligent techniques in self-healing, the topics of root cause analysis and proactive prediction techniques remain an open research problem since the inherent variety and the increasing complexity. Many recent studies introduce various methods for automated system management [6], attempting to explore new approaches to improve self-healing capability.

IBM research on self-aware distributed systems aims at automating an increasingly complex and expensive task of real-time problem diagnosis in large-scale distributed system by using state-of-art machine learning, probabilistic reasoning and information theoretic approaches. It shows an architecture of diagnosis system called RAIL (Real-Time Active Inference and Learning), using a real-time event stream of various observations of the system's behavior and a Bayesian network dependency model [7].

The most current focus of the work is on:

. Active diagnosis: Adjusting the probe set dynamically to improve diagnosis;

. Extending local approximation techniques to incremental, real-time scenarios;

. Handling intermittent failures, dynamic routing, and other nonstationarity in the network state and behavior using on-line learning;

. Active learning using flexibility in probe selection

The RAIL system makes online inferences about the underlying faults and performance degradations of the system's components that might be difficult or expensive to measure directly. It also actively requests the most informative tests to improve the diagnosis and updates the dependency model using online learning.

The Sun Fire X4500 server features the latest fault management technologies. This technology is incorporated into both the hardware and software of the server. Predictive Self Healing introduces a new software architecture and methodology for fault detection, diagnostics, logging, and system service management across Sun's product line. There are two major components in Predictive Self Healing [8]: Fault Management Architecture (FMA) and Service Management Facility (SMF).

Predictive self healing addresses two problems of commercial IT:

1) Fix problems before they occur

2) Circumvent operational problems with services

Problem localization techniques based on the results of various measurements are widely used. Two approaches are commonly used in dependency based methods for fault reasoning and localization: deterministic approach and probabilistic symptom-fault causality model. One approach utilizes two concepts, including dependency graphs and dynamic runtime performance characteristics of resources that comprise I/T environment, to design algorithms for rapid root cause identification in case of problems [9]. A novel technique called Action Integrated Fault Reasoning or AIR is presented to seamlessly integrate passive and active fault reasoning in order to reduce fault detection time as well as improve the accuracy of fault diagnosis [10]. The technique proposed in [5] isolates the most probable set of faults through incremental updating of the symptom explanation hypothesis. It uses a probabilistic model, which makes the technique applicable to systems with a high degree of non-determinism. A system architecture that adapts to changing dependencies in a dynamically changing environment of mobile ad hoc networks is designed, using a dynamic dependency model and hypothesis search space to incorporate the observed changes [11]. A technique for fault detection in the perception mechanism of a context-aware ubiquitous system using Bayesian network is defined in [12], which facilitates the correct context formation based on perceived data, hence improving overall system performance. A health engine, the central component of 'Self-Awareness and Control' architecture, combines domain independent statistical analysis and probabilistic reasoning technology - Bayesian network with domain dependent measurement collection and evaluation methods [13]. An automated approach to performance problem localization in service-oriented systems focuses on causing services that are the most important causes of slow end-to-end response time [14]. A lightweight Bayesian network model is adopted to assess the response time degradation caused by services even in the event of missing data. In [15], an application of Bayesian reasoning using belief networks to non-deterministic fault diagnosis in complex communication systems is investigated. It applies two Bayesian inference algorithms that calculate belief-updating and most-probable-explanation queries in singly connected belief networks to perform fault localization in belief networks with loops. As structure learning is the main issue when using Bayesian network method, more researches induce automatically learning

Bayesian network from data despite instead of manual designed model based on domain knowledge [16], which may be disputed as it is unalterable and unable to reflect to the real−time changes of data.

Actually, the existing fault management techniques can be classified into two categories: dependency based methods and non−dependency based methods. However, predictive self−healing using non−dependency based methods such as rule−based or case−based inferences will bring problems because most of them rarely consider relationships between collected information, which are inefficient in the case of uncertainty. 1) The larger the numbers of levels of considering components, the more generated rules are required, which makes the system experience high overload and low efficiency. 2) All rule or case generations should be user−defined in advanced. 3) All created rules or cases are impossible to be comprehensive, which implies that one event occurred may not be included in the existing aggregation. Thereby, in the case of such an uncertainty, with the fact that there are somewhat interrelated relationships between various system metrics, we can consider fault management starting with representing a probabilistic dependency model among system elements rather than considering them mutually independent in large scale distributed application domains.

3. Fault Management for Self−Healing

Autonomic computing implies that it has ability to be aware of the environment and have the ability to learn from previous experiences to recover from faults with less human intervention, namely, the capability of self-awareness. Fault diagnosis and prognosis based on real-time streams of computer events contribute to self-healing for the purpose of determining root causes of problems i.e. fault localization and predicting future situations such as potential problems that going to occur in large scale distributed computing system, especially in Ubiquitous environment.

In a distributed computing system, real-time status information which represents real-time system performance events is abstracted and filtered. Furthermore, fault detection is executed at the same time. After precondition including parameter selection and ordering, a dependency model using ordering parameters is constructed through learning from the collected information. Based on the created model and the symptom information collected, root cause locations and predicted problems are inferred via probabilistic dependency analysis after probability propagation throughout the network. Hence, proper repairs are applied into the target system to repair faults based on the results of inferences. The overview of fault management for self-healing is described in Fig. 1.

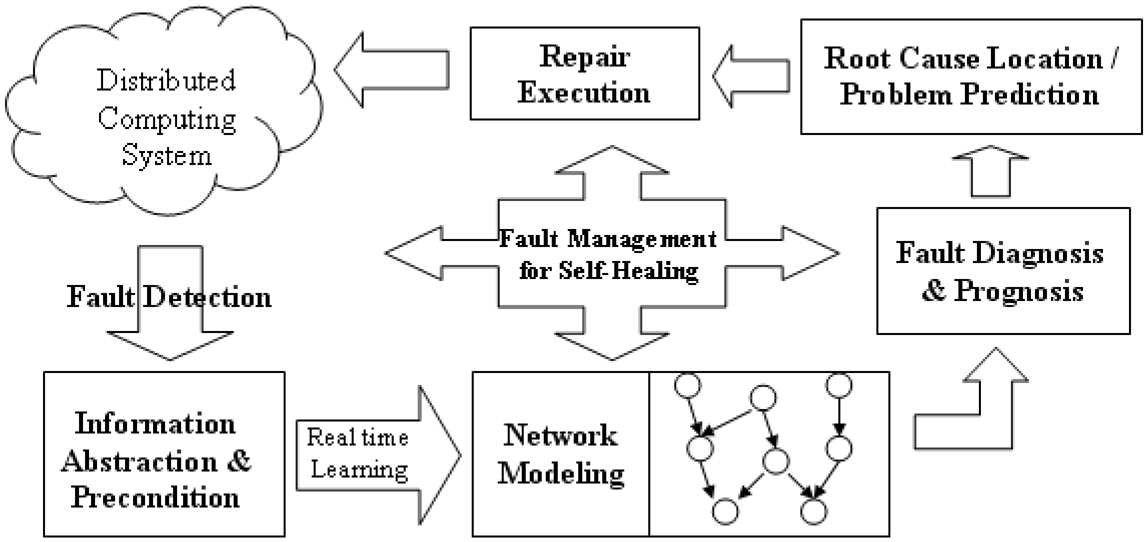


Fig. 1. Fault Management for Self−Healing

In this paper, we use probabilistic machine learning method, which is mainly used as a modeling tool, to propose an inference model structure for fault diagnosis and prognosis in self−healing system. It inferences the likelihood that a factor is in one state which is dependent on other factors' states that reflect the degrees of confidence. In terms of accuracy and efficiency of diagnosing problems and forecasting potential problems, we can deal with the data in the raw beforehand then combine prior information for inference.

3.1 Fundamental and Characteristics of Bayesian Network

Bayesian network or Bayesian belief network is a graphical structure to represent and reason about an uncertain domain, including nodes represent random variables of interest in the domain and arcs represent direct influences i.e. conditional dependencies between variables. It emphasizes that a link between two nodes does not, and need not, always imply causality, i.e. the network is not always a causal structure. It only implies a direct influence of parent node over child node in the sense that the probability of child node is conditional on the value of parent node, and two nodes may have a link between them even if there is no direct cause [17]. The formula (1) expressed below is a simple representation of Bayes' rule.

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{\sum_i P(B \mid A_i) \cdot P(A_i)} \quad (1)$$

For more complex problems, it also has a mechanism that can propagate probabilities via extending Bayes' Rule throughout the whole network automatically. If a Bayesian network encodes the true independence assumptions of a distribution, we can use a factored representation for the distribution as follows:

$$\begin{aligned} P(x_1, \ldots, x_n) &= \prod_{i=1}^{n} P(x_i \mid x_{i+1}, \ldots, x_n) \\ &= \prod_{i=1}^{n} P(x_i \mid Pa(x_i)) \end{aligned} \quad (2)$$

Formula (2) shows that instead of the full joint distribution, we need only the conditional probabilities of a variable given its parents, which is based on Markov assumption. A distinct characteristic of Bayesian network is that it is especially useful in uncertainty domains with information about the past and/or the current situation being vague, incomplete, and conflicting. It's easy to explain how a system arrived at a particular recommendation, decision, or action as it can represent probabilistic relationships between nodes dynamically. Furthermore, Bayesian Network can be run in multiple directions, including bottom-up and top-down, which features of Bayesian Network are applied in this paper. Another feature is that it can post evidence to a Bayesian belief network to predict a result or to diagnose a cause based on analyzing current beliefs. The evidence is information about a current situation and beliefs are the probability that a variable will be in a certain state based on the addition of evidence in a current situation [18].

Learning a Bayesian network B=<G, P> from data consists of learning Bayesian structure G and learning probabilistic parameter P. Structure learning is to find the DAG structure G that is most probable to the training data D. However, there are too many possible numbers of DAGs which is the difficulty of Bayesian structure learning. The number of possible DAGs for n nodes is computed as follows:

(3)

Following above formula (3), there will be 3 possible structures for 2 nodes, 25 possible structures for 3 nodes, 29,281 possible structures for 5nodes, etc. It increases exponentially along with increasing number of nodes. It' s impossible for us to consider and measure all possible structures. Therefore, finding the most probable structure effectively must be taken into account when learning Bayesian networks.

3.2 Process of Modeling

Using Bayesian network algorithm to perform both root cause analysis and proactive problem prediction functions for self-healing system, we should emphasize the method of modeling a compact structure by following a defined process. Bayesian network structure can be created by hand [16] [19], which is expensive in terms of time and cost, and also manual designed model may be disputed as it is unalterable and unable to reflect to the real-time changes of data. Recently, many researchers have begun to investigate methods for learning Bayesian network from data automatically to search a structure that can capture a real-world distribution, and it can deal with missing data and hidden variables. A learned graph structure provides much insight into the domain whose knowledge information is collected for learning. Therefore, learning structure plays an important role in the whole course and has a direct effect on the final results of probabilistic references.

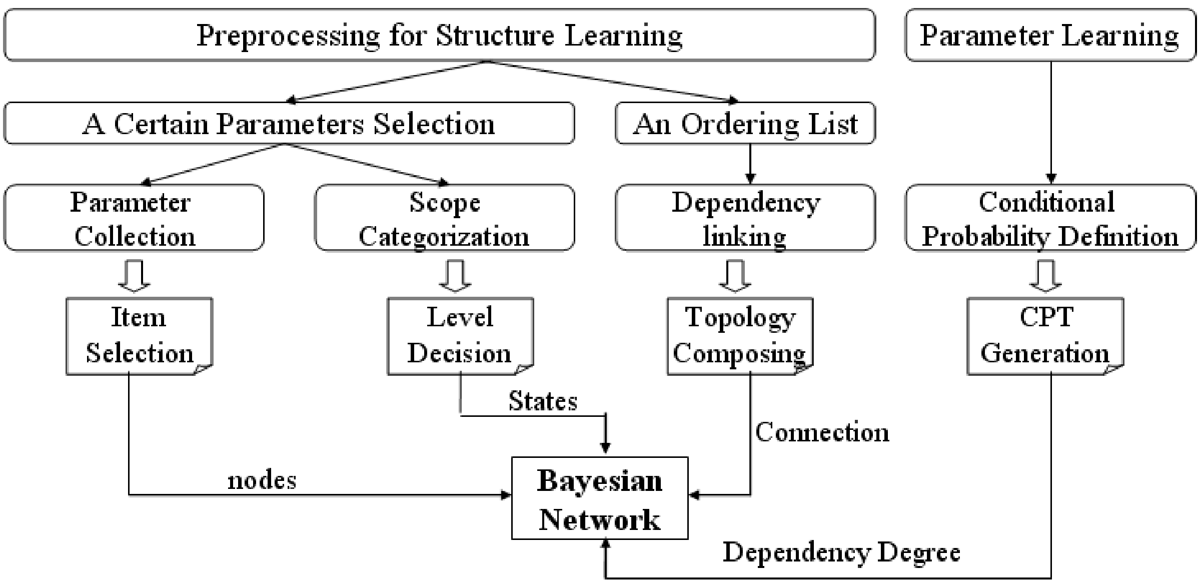


Fig. 2. Process of Bayesian Network Modeling

Obviously, there are mainly two learning phases that includes structure learning and parameter learning in process of modeling as described in Fig. 2. To create an efficient and accurate model, a preprocessing step which provides certain number of parameters and an ordering list of them is prepared for structure learning which is difficult in diverse domains especially with more parameters. The nodes and states are determined based on the selected parameters, and the topology composing is presented as dependency linking which is learned based on the ordering list. Given the built structure learning, parameter learning which defines Conditional Probability Table (CPT) for each node is carried out to form a complete Bayesian network.

Learning structure is more crucial part of the whole course and the final results are directly related to it. Recently many methods for structure learning have been developed, finding the structure that is most suitable to training data. The score based search method uses approximate search algorithms to construct candidates and measures them using scoring evaluation. The dependency analysis method starts with analyzing dependency relationships between nodes to construct a network. However, both methods are not suitable when there are larger data, which in this case brings overfitting which is one of the main issues in using machine learning. The overfitting phenomenon occurs when too many parameters are considered in a given domain. In building Bayesian network structure, it occurs when considering too many parameters in structure learning. So in order to solve such problems and make structure learning more efficient, we can provide preconditioning course previously.

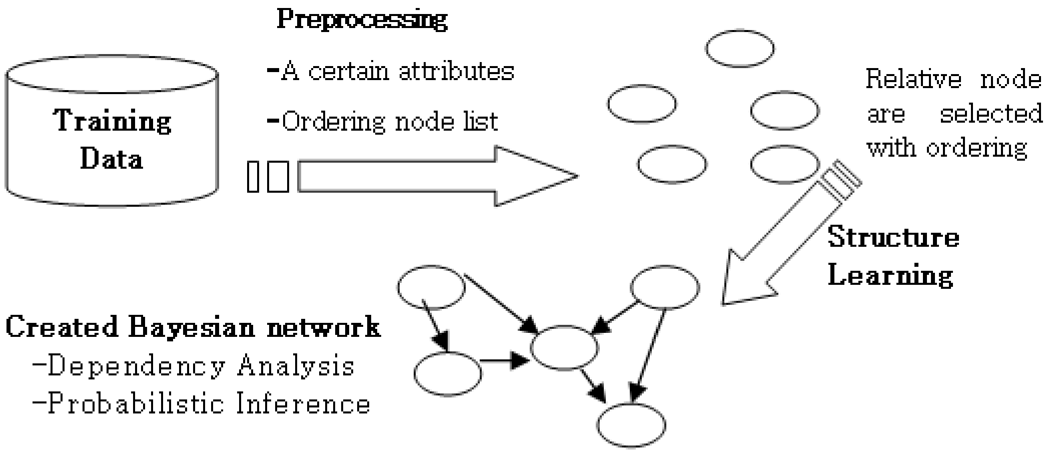


Fig.3. Preprocessing

Fig.3 depicts the preprocessing before Bayesian network structure learning. Given training data, it selects certain relative factors with ordering, and then enters into the step of structure learning, on which probabilistic dependency analysis are based.

There are two phases included in preprocessing. First, from the given large dataset with more parameters, it can only consider factors that are more relative with focusing problems, i.e. choose relative factors describing the domain. We can downsize the number of factors by using information theory method to analyze relationships or clustering or other approaches. After determining certain factors, it arranges them in a special order, which means anterior one has direct influence on the posterior one in the same direction of arrow, by analyzing information gain between pairs of observing data. However, it emphasizes the assumption is that problematic parameters are independent of each other when learning structure. All parameters in the ordering lists are able to have influence on each problematic parameter; thereby each problematic problem has the same ordering list only with each different problematic factor as the last one, which implies that all the factors in front of it could be a parent node of the last one. The pseudo code of the ordering algorithm is described as follows.

```
Input: Separate observing parameters from problematic parameters stored in set S= {V1 ...Vn, P1...Pm}.
Output: An ordering list with a certain number of parameters
1) Select a certain observing parameters with high relevance to problematic parameters.
   for each problematic parameter Pj ( j= 1 to m) do
       for i = 1 to n do
       compute information gain Gij = Gain(Vi,Pj) = H(Pj)-H(Vi,Pj) (H() means entropy)
       end for
       rank parameters with Gij from maximum to minimum and save them to list Lj
    end for
Combine all lists Lj (j =1 to m), select observing parameters with the mean information gain exceeds
defined threshold value.
Return the selected parameters and all problematic parameters. S'= {V1...Vk, P1...Pm} (k<n)
2) Make an ordering list for the selected observing parameters in set S'
   Initialize set S''={V1,...,Vk} except for problematic parameters; pair set P={empty}; list L={empty}
   Select two parameters Vx and Vy from the head of set S'' (x!=y)
   Compute Gain(Vx, Vy) and Gain(Vy, Vx) for each pair, put pair(Vx->Vy) with larger Gain into set P
   stop when there is close loop, run until all parameters in set S'' are considered.
   Sort the pairs in set S'' to an single ordering list L
   Return an ordering list L only with observing parameters
```

Applying ordering node list into the next step of learning, for score based search method, it can reduce the entire search space when adding link to construct network, as a node can be parent only of node which is behind it according to the ordering node list; for dependency analysis method, it can reduce computing complexity as the number of nodes is decreased and determine the direction between two nodes.

3.3 Method of Structure Learning

In this paper, an ordering node list with certain parameters is used as input to create a fine-grained model by analyzing conditional independency evaluation, which determines dependency relationships between all pairs of nodes. It should be stressed at this point that Bayesian network implies conditional independencies via showing conditional probability tables for leaf nodes having direct parent nodes.

Input: A certain parameters with special ordering
Output: A Bayesian network

Initialize a graph G (V, E) where V= {a certain parameters}, pair list $L = \{\Phi\}$. Edge $E = \{\Phi\}$.
For each $(x_i, x_j) \subset V$ (except for pairs of problematic parameters)

Compute mutual information. $I(x_i; x_j) = \sum_{x \in r_i} \sum_{x' \in r_j} P(x, x') \log_2 \left(\frac{P(x, x')}{P(x), P(x')} \right)$

For all the pair of nodes that have mutual information greater than a certain threshold ε, sort them and put these pairs of nodes into list L from large to small.

Get pair of nodes one by one in list L and remove them from it. Add the corresponding arcs to E with direction of arcs according to the ordering node list.

Repeat above procedure until list L is empty.
Return a Bayesian network based on arcs in set E.

Bayesian network structure learning from data presents an efficient algorithm based on the conditional independence (CI) test to measure dependency relationships [20]. In this paper, one of the structure learning mechanisms, which begin with the definition of Bayesian network, is based on computing mutual information introduced in the Information Theory for pairs of nodes to reflect different degrees of dependency relationships among them. A threshold is given to determine the existence of probabilistic dependency relationship between nodes.

4. Experiment and Evaluation

Following the rapid growing internet systems in the Ubiquitous computing era, violations of service level objectives [21] are related to reliability of system and quality of service. As automated management capability described in self-healing, when there are faults such as bottlenecks, violation of Service Level Objectives occurred, the system should find which factor is directly related to them and affect high level performance of system automatically, by analyzing observed parameters consisting of performances of individual servers or processes, capability of network, hardware and software, dynamic variation resource utilizations by different types of client requests, and temporary traffic situation. Thereby, they can be used to determine which part of the system is responsible for current fault of the system, then it is repaired appropriately; oppositely, the collected information can be used to forecast system potential problems, preventing them in advance.

4.1 Experimental Illustration

In our experiment, it collects and filters data of interest that can be used for analysis, including CPU, memory, disk utilization, count of client, package volume, bandwidth logged in a server and detects information such as threshold violation in response time and throughput, on which we rely to analyze and control system management for providing high quality of service and performance. After collecting sample data, each parameter should be categorized into corresponding classes according to given criteria, such as High, Medium, Low for performance parameters and Error, warning, normal for problematic parameters, as shown in Table 1.

Table 1. Classified parameters

CPU	RAM	Disk	Bandwidth	PacketVolume	ClientCount	Throughput	Responsetime
Medium	Low	Medium	Medium	High	Medium	Normal	Normal
Medium	Low	Low	Medium	High	Medium	Normal	Error
Low	High	High	Medium	High	High	Warning	Normal
Low	High	Medium	Medium	Medium	Medium	Normal	Error
High	Low	Medium	Medium	Medium	High	Normal	Normal
Medium	Low	High	High	Medium	High	Normal	Normal
Low	Low	Medium	High	High	Low	Warning	Normal
Low	Low	Medium	Medium	Medium	Low	Warning	Error
High	Low	Medium	Medium	Medium	High	Normal	Normal
Medium	Medium	Medium	Medium	Low	Low	Error	Normal
Medium	Low	High	Medium	Low	Medium	Normal	Warning
Medium	High	High	Medium	Low	Medium	Error	Normal

We take above parameters as input to create node ordering with certain number of parameters which are highly related to problematic parameters. After learning on the training data, the result of selecting relative parameters is:

Then the observing parameters are ranked by using the proposed approach to output a node ordering list without problematic parameters, as follows:

With the predefined assumption, the problematic parameters response time and throughput are independent of each other. From the above ordering list, it implies that the node orderings can be used when constructing a Bayesian network.

The flowchart of the preprocessing is shown in Fig. 4 as follows:

$S = \{V_{CPU}, V_{RAM}, V_{DISK}, V_{Bandwidth}, V_{client}, V_{filesize}, P_{response}, P_{throughput}\}$

For all P_i in problematic parameter set

Observing Parameters						Problematic Parameter
V_{CPU}	V_{RAM}	V_{DISK}	$V_{Bandwidth}$	V_{client}	$V_{filesize}$	$P_{response}$

Observing Parameters						Problematic Parameter
V_{CPU}	V_{RAM}	V_{DISK}	$V_{Bandwidth}$	V_{client}	$V_{filesize}$	$P_{throughput}$

1) A certain parameters are selected

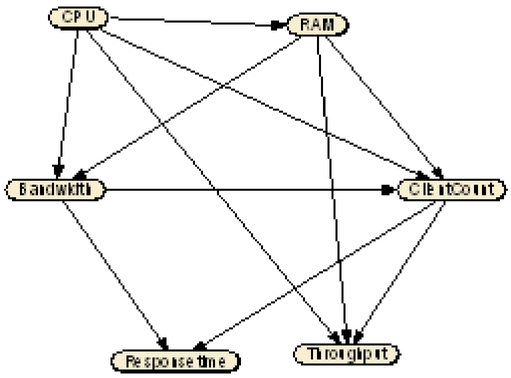
Observing Parameters				Problematic Parameter	
$V_{Bandwidth}$	V_{client}	V_{CPU}	V_{RAM}	$P_{response}$	$P_{throughput}$

2) An ordering list is generated

$$\boldsymbol{Orderinglist} = \{\boldsymbol{V}_{cpu} -> \boldsymbol{V}_{ram} -> \boldsymbol{V}_{bandwidth} -> \boldsymbol{V}_{client} -> \boldsymbol{P}_{response}, \boldsymbol{P}_{throughput}\}$$

Fig. 4. Flowchart of Preprocessing

From Fig. 5 we can see that the created structure is a compact hierarchy model after learning from certain parameters and ordering list. Comparing with traditional double-deck cause-effect structure of Naïve Bayesian network, it also discovers internal dependency relationships among causal parameters in the network structure, which makes the results of inferences more accurate. The next learning phase is parameter learning given structure and training data i.e. fixing conditional probabilities for each node. Fig.6 describes the complete Bayesian network after parameter learning.



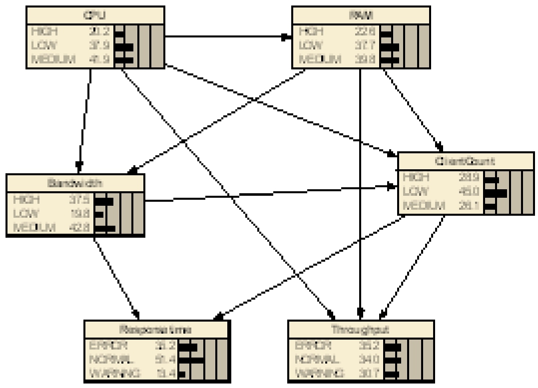


Fig.5. Structure Learning

Fig.6. Parameter Learning

Given the convinced states of several parameters, and it makes the known state with assured belief, which operation can change beliefs of all nodes that related to such one node after probabilities being propagated throughout the whole network. As mentioned above, the evidence is information about a current situation, and belief is the probability that a variable will be in a certain state. According to them, we can find the answer which we need by adjusting the beliefs of states of one node, and also can discover that how the nodes affect each other.

For instance, when a violation of response time is observed, which means that it makes response time be of 'error' state, we have the evidence of response time by changing the belief of 'error' state of response time with 100% and the other states with 0%. After then, the most probable impact factor can be decided by finding the 'low' state of one node with max probability comparing to the worse states of other nodes, namely 'low' state of bandwidth. Therefore, we can say that the root cause of response time is bandwidth, and the causal factors can be ranked from max probability to min probability of 'low' state. The propagation of probabilities is shown in Fig. 7.

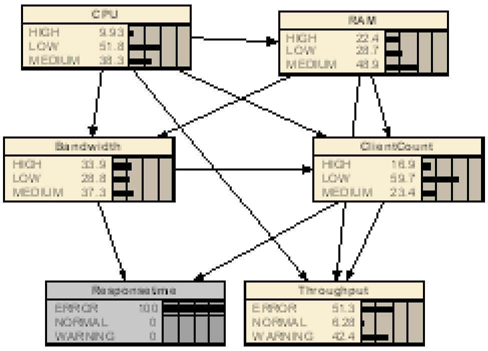


Fig. 7. Bottom-up Probability Inference for Diagnosis

On the other hand, when the utilization of CPU resource arrives over 95% which means that it belongs to 'high' state, so we adjust the believe of 'high' state of CPU to 100% evidence. Then we can see that the probability of 'error' state of throughput gets up to the max one, which stands for that there will be a fault of throughput appeared in coming time. The changed probabilities propagation is showed in Fig. 8.

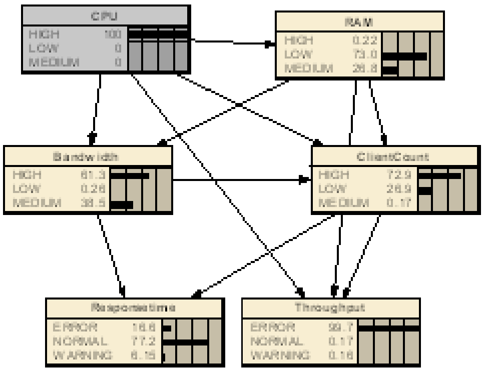


Fig. 8. Top−down Probability Inference for Prediction

Applying the testing data to the model, the rate of validity is up to over 85%. These results derived from diagnosis and prognosis are very helpful for system to take correct repairs to figure out faults or to avoid potential faults in advance. From the probabilistic network, it’ s easy for us to understand how the factors affect each other by changing the evidences of nodes with dynamic representation.

4.2 Evaluations

For proving the effects of the proposed Bayesian network approach to fault management for self−healing in performance evaluation, we apply testing data into the built model then compare the results with actual results. At first, we evaluate time consumption of structure learning and error rate given different numbers of parameters, showing that the obvious effect when using a certain number of parameters that are highly related with the domain. From Fig. 9, as the number of parameters grows, the time consumption mounts up but the error rate of detecting faults drops, and we can find that the number of parameters corresponding to the crossing of two elements can be chosen as the appropriate quantity of considering parameters in such domain.

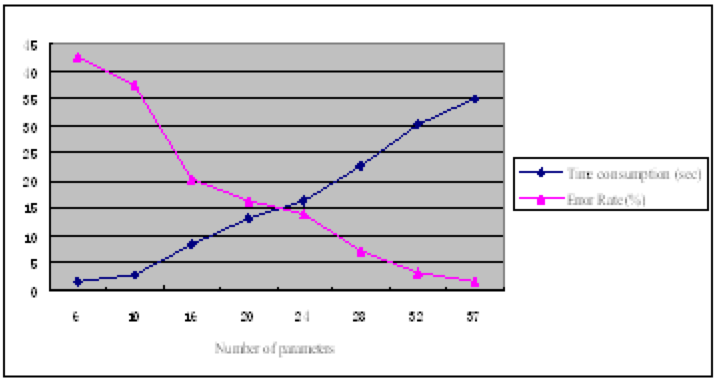


Fig. 9. Evaluation with Different Number of Nodes

Comparisons of time consumption and accuracy are evaluated in the case of selecting certain parameters applying preprocessing or not. Table 2 can tell us that with node ordering list, there are both improvements on time consumption and accuracy of inference results.

Dimensions	Time consumption (sec)	Accuracy (%)
with node ordering	15.48	90.3
without node ordering	16.37	85.2

Table 2. Comparison on cases with and without node ordering

For accuracy of root cause analysis, we estimate the position of the exact cause of current problems in the rankings that include all causal parameters after inferences with different numbers of parameters. Like the results showed in Table 3, the average ranks of root causes with ordering are quite close to that without ordering.

Table 3. Accuracy of root cause analysis

Numbers of Causal Parameters	Root cause in ranking With ordering	Root cause in ranking Without ordering
6	0.9	0.8
10	1.5	1.4
15	1.5	1.6
20	1.7	1.8

Table 4. Comparisons of structure learning methods in Bayesian network

	Manual construction	Scoring based method	Dependency analysis method
Requirements	Expert knowledge & Domain knowledge	Scoring measurement & Search algorithm (e.g. Greedy)	Conditional information measurement Domain knowledge
Advantages	Easy to construct network	Efficient for dense structure	Efficient for sparse structure
Disadvantages	Domain experts are lacking; Difficult to reflect real time data dynamically.	Time consuming Computing complexity	Need to define a threshold to decide relationship among nodes
Node Selection	Experts decide relationships	Reduce possible structures	Minimize computing complexity
Node ordering	No need	Reduce search space (select parent of one node before it)	Determine the direction between two nodes

As shown in Table 4, the comparison of structure learning under given certain quantity of parameters shows that taking an ordering list as input of structure learning can bring high efficiency and accuracy whatever leaning methods are used.

5. Conclusion

In this paper, an approach to fault management using Bayesian network is proposed to provide both diagnosis and prognosis function for self−healing in Ubiquitous computing environment. In order to improve the performance of learning with domain knowledge, a preprocessing step which reduces the size of parameters is added to improve the whole process of Bayesian network modeling. Using the proposed methods, we can transform a complex system into a compact model with high efficiency and accuracy, on which we depend to make inference via probabilistic dependency analysis. In contrast to other existing researches on using Bayesian network, it can process input data in advance, which is implemented with high accuracy to improve the efficiency of structure learning. In order to prove availability of the proposed approach, we perform it in the system performance domain to achieve automated system management and make various comparisons under different conditions.

As ongoing work we will continue to research on probabilistic dependency analysis, including designing an integrated version of diverse fault management techniques. Autonomic problem localization requires a mechanism that should be proactive, all around and accurate. There are many algorithms[22] [23] used in various fields for machine learning, including time−series, decision Tree, case−based reasoning, rule based reasoning and so on. Following these methods, it can provide multiple functions in fields of diagnosis, prediction, fault isolation and recovery. Furthermore, combining region-based and block-based techniques, we can develop a procedure for content-based fault management and retrieval system for self-managing with improved accuracy.

References

[1]. R. K. Sahoo, A. J. Oliner, I. Rish, M. Gupta, J. E. Moreira, S. Ma, R. Vilalta, and A. Sivasubramaniam, "Critical event prediction for proactive management in large-scale computer clusters", In Proceedings of the ACM SIGKDD, Intl. Conf. on Knowledge Discovery and Data Mining, pp.426-435, August 2003

[2]. Jeffrey O.KephartDavidM.ChessIBM ThomasJ.WatsonResearchCenter, "The Vision of Autonomic Computing", IEEE Computer Society, January 2003

[3]. Irina Rish, Mark Brodie, Sheng Ma, Natalia Odintsova, Alina Beygelzimer, Genady Grabarnik, and Karina Hernandez, "Adaptive Diagnosis in Distributed Systems", IEEE Transactions on Neural Networks, March 2005

[4]. M. Steinder and A.S. Sethi, "A Survey of Fault Localization Techniques in Computer Networks. Science of Computer Programming", Special Edition on Topics in System Administration Vol.53,2,pp.165-194,Nov.2004

[5]. M.Steinder, A.S. Sethi, "Probabilistic fault diagnosis in communication systems through incremental hypothesis updating", The International Journal of Computer and Telecommunications Networking archive, July 2004

[6]. Yuan-Shun Dai, "Autonomic Computing and Reliability Improvement", Proceedings of Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC'05) 204-206

[7]. IBM Self-Aware Distributed Systems:
http://domino.watson.ibm.com/comm/research.nsf/pages/r.ai.innovation.2.html

[8]. Sun Microsystems: Predictive Self-Healing in the Solaris 10 Operating System:
http://www.sun.com/bigadmin/content/selfheal

[9]. Manoj K.Agarwal, Karen Appleby, Manish Gupta, Gautam Kar, Anindya Neogi, and Anca Sailer, "Problem Determination Using Dependency Graphs and Run-Time Behavior Models", IFIP International Federation for Information Proceeding 2004

[10]. Yongning Tang, AI-Shaer.E.S., Boutaba.R., “Active Integrated Fault Localization in Communication Networks”, Integrated Network Management, 2005
[11]. M. Natu and A. S. Sethi, “Using temporal correlation for fault localization in dynamically changing networks,” International Journal of Network Management, 2007
[12]. Bilal Ahmed, Young-Koo Lee, Sungyoung Lee and Yonil Zhung, “Scenario Based Fault Detection in Context-Aware Ubiquitous Systems using Bayesian Networks”, Proceedings of the 2005 International Conference on Cmoputational Intelligence for Modeling, Control and Automation, 2005
[13]. J.Bronstein, A.Das., “Self-Aware Services - Using Bayesian Networks for Detecting Anomalies in Internet-based Services”, Hp Labs Technical Reports HPL-2001-23R1, 2001
[14]. Rui Zhang, Steve Moyle and Steve McKeever, and Alan Bivens, “Performance Problem Localization in Self-Healing, Service-Oriented Systems using Bayesian Networks”, Proceedings of the 2007 ACM symposium on Applied computing, pp.104-109, March 2007
[15]. Malgorzata Steineder, Adarshpal S.Sethi, “Probabilistic Fault Localization in Communication Systems Using Belief Networks”, IEEE/ACM Transactions on Networking, pp.809-822, October 2004
[16]. Jianguo Ding, Bernd Kramer, Yingcai Bai, and hansheng Chen, “Backward Inference in Bayesian Networks for Distributed Systems Management”, Journal of Network and Systems Management, Vol.13, No. 4, December 2005
[17]. Ethem Alpaydm: Introduction of Machine Learning. © 2004 Massachusetts Institute of Technology
[18]. Charles River Analytics Inc: About Bayesian Belief Networks. © Copyright, Charles River Analytics, Inc. 2004

[19]. Cheng, J., Bell,D. and W.Liu, “Learning Bayesian networks from data: an efficient approach based on information theory,” In Proceedings of the sixth ACM International Conference on Information and Knowledge Management , 1997

[20]. Jie Cheng, David A. Bell, Weiru Liu, “An Algorithm for Bayesian Belief Network Construction from Data”, In Proceedings of AI &STAT’(1997) 83-90

[21]. http://www.risi.com/services/sla.html

[22]. R.Vilalta, C.V.Apte, J.L.Hellerstein, S.Ma, S.M.Weiss, “Predictive algorithms in the management of computer systems”, IBM Systems Journal issue 41-3, Artificial Intelligence, Vol.41, No.3, 2002

[23]. Sa’adah Hassan, David McSherry, David Bustard, “Autonomic self healing and recovery informed by environment knowledge”, Artificial Intelligent Rev, pp. 89-101, 2006

논문요약

자가 치유를 위한 결함 진단과 예측 모델

성균관대학교
전자전기컴퓨터공학과
장 길 산

결함 관리는 관찰된 결함의 근본 원인을 자동 인식 하는 것이 가능하기 때문에 규모가 큰 분산시스템에서 중요 역할 수행하며 시스템의 신뢰성 개선을 위해 시스템의 관리와 제어가 가능한 자가 치유를 지원한다. 결함 지역화를 지원하는 기존 연구들은 유비쿼터스 환경에서 기계학습과 같은 인공지능 기술들을 주로 사용한다. 따라서, 본 논문에서는 자가 치유를 위해 실시간 시스템 성능 스트림에 대한 학습을 통해 확률적 의존 분석을 기반으로 하는 결함 지역화 방법을 제안하여 진단과 예측기능을 동시 제공한다. 학습 방법으로 베이지안 네트워크 알고리즘을 사용하여 각종 관련된 요소들을 연결함으로써 네트워크를 생성하고 확률적 의존 관계를 통해 귀납적과 연역적 추론기능을 제공한다. 베이지안 네트워크의 구성은 노드들 간의 연관성을 찾아내는 것이 중요하기 때문에 그것을 구성하는 인자의 개수가 많은 경우 노드 순서 리스트를 추출하는 사전처리 과정이 필요하다. 따라서 전체 모델링 프로세스에 대한 개선이 요구된다. 이러한 문제를 해결하기 위해 발생한 문제와 관련성이 높은 노드 순서 리스트를 추출하는 방법을 제공한다. 구조 학습을 지원 하는 사전처리 방법을 통해 다양한 문제 영역에서의 학습 효율성을 높이며 학습에 필요로 되는 시간을 줄인다. 제안 방법론을 통해서 시스템의 자원 문제를 신속하고 정확하게 진단하는 것이 가능하며, 관찰된 정보를 기반으로 실행

중에 발생되는 잠재적인 문제를 예측하는 것이 가능하다. 시스템 성능 평가 영역에서 제안 방법론을 적용한 시스템 성능 분석을 기반으로 진단, 예측의 효율성과 정확성을 평가하여 제안 방법론의 유효성을 입증하였다.

주제어: 결함 진단, 예측, 사전처리, 확률 의존 분석, 자가 치유