NETWORK PROBE SELECTION PROBLEM

PROJECT PROPOSAL FOR CS357

Presented by

Kunal Gupta (150001015)

Bhor Verma (150001005)

Computer Science and Engineering

3rd Year

Under the Guidance of

Dr. Kapil Ahuja

Department of Computer Science and Engineering

Indian Institute of Technology Indore

Autumn 2017

CONTENTS

INTRODUCTION

A defining characteristic of today's information age is our reliance on data centers. The scale and complexity of these data centers have been increasing rapidly. This in turn is limiting our ability to understand and control the data center operations.

Availability of good quality monitoring data is a vital need for management of today's data centers. However, effective use of monitoring tools demands an understanding of the monitoring requirements that system administrators most often lack. Instead of a well-defined process of defining a monitoring strategy, system administrators adopt a manual and intuition-based approach.

A quintessential requirement for development of automated analytics-led solutions for understanding of the as-is state of these systems is the availability of good quality monitoring data. It is very important to systematically deploy a monitoring framework to capture behavioral characteristics of all compute, communication, and storage components. Monitoring metrics need to be collected at various layers ranging from the hardware metrics such as CPU and memory utilization, to operating system and virtual machine layers, to database and application layers, among others. The good news is that a lot of monitoring tools exist that can monitor a wide variety of system components. Most of these tools are capable of monitoring a large range of metrics and can be configured as per needs.

However, the bad news is that the quality of the monitored data is often a suspect. Effective use of the monitoring tools demands an understanding of monitoring requirements that system administrators most often lack. Instead of a well-defined process of defining monitoring strategy, system administrators adopt an ad-hoc, manual, and intuition-based approach. This leads to inconsistent and inadequate data collection and retention policies.

For this project, we propose a solution to adapt the monitoring level of a system component using probes[1].

[1] Probes refer to test transactions that are sent in the network. The success and failure of these probes depend on the success and failure of the network components used by the probes. An example of a simple probing technique is to ping all nearby nodes and check the response time and packet loss.

PROBLEM STATEMENT

The key idea is to send probes in the network, infer the system state from the probe results, and adjust the monitoring of system components based on the inferred system state. The key objectives of the problem are:

1. Probe selection - how to select the right set of probes such that criticality of components can be correctly estimated and appropriate recommendations can be provided for each component.
2. Probe analysis - how to analyze probe results to infer health of components on probe-path and recommend monitoring levels.
3. Adapting monitoring level of a node - how to adapt the monitoring level of a node based on the recommendations provided by one or more probe stations.

Currently we have only concentrated on the probe selection objective of our problem. Thus, we will be solely be explaining the same. Other objectives will be added as deemed suitable.

MATHEMATICAL FORMULATION

NOTATIONS AND ASSUMPTIONS

The network is in the form of an undirected graph

Let the set of all probes be P and p represent any probe in P. N is the total number of probes.

Then,

- $nodes(p)$ represents all the nodes that p can monitor,
- $C(p)$ is the cost associated with probe p, where cost is defined as length of the probe = $|nodes(p)|$,
- and c is the minimum number of probes that monitor each node.

COST FUNCTION

Cost is defined as the length of the probe where length is the total number of nodes that can be monitored by any given probe.

WEIGHT

Weight of a node n_i, Weight (n_i), is the inverse of the number of probes probing a node. Thus, if k probes (including the new probe) pass through a node then the weight of the node is computed as $1/k$. The weight of a node decreases with an increasing number of probes passing through a node. This weight criteria is used in probe selection to give priority to less covered nodes.

CONDITIONS

Following are the set of conditions which need to be fulfilled for our objective:

- ***Condition 1*** - One probe passes through multiple nodes, and hence, it is not capable of uniquely identifying the health of each node on its probe path. Therefore, each node should be probed by multiple probes so that a more accurate estimate of the node's health can be made based on the analysis of multiple probe paths. The average number of probes probing each node:

$$\left(\sum_{p \in P_{sel}} |nodes(p)|\right) \Big/ N$$

should be maximized.

- ***Condition 2*** - In a probe passing through a large number of nodes, one node makes a small contribution in the overall end-to-end performance of the probe. Performance degradation of one node is thus not very effectively visible in the end-to-end performance of the probe. Hence, probes passing through a large number of nodes are not good candidates for providing monitoring recommendations. The average length of a probe:
$$\left(\sum_{p \in P_{sel}} |nodes(p)|\right) \Big/ |P_{sel}|$$
should be minimized.
- ***Condition 3*** - All nodes in the network should be covered. In other words, every node must have at least one probe passing through it.
$$\bigcup_{p \in P_{sel}} \Big(\big(nodes(p) \big) \Big) = N$$
- ***Condition 4*** - Finally, the probes come with a cost of increased probe traffic in the network and increased processing at the nodes. In order to minimize the probe traffic, the number of selected probes, $|P_{sel}|$, should be minimized.

OBJECTIVE FUNCTION

We use LP formulation of the standard set cover with a slight modification to incorporate the desired constraints. The modification is required as the set cover problem requires each element to be covered in the selected subsets only once. Whereas, our requirement is that the node should be probed multiple times (*condition 1*). That is, each element in the universe must get covered at least c times in selected subsets.

We propose the following linear program that includes the four stated objectives in the form of constraints and objective functions:

$$min \sum_{p \in P} x_p \times C(p)$$

$$s.t. \sum_{p \in P} \Big(x_p \times C(p) \Big) \geq c \quad \forall n \in N$$

$$x_p \in \{0,1\} \quad \forall p \in P$$

In the formulation, x_p, is a binary variable indicating whether probe p is selected (x_p = 1) or not (x_p = 0). $C(p)$ = $nodes(p) \cdot p_n$ is the variable indicating the presence of node n in the path of probe p. p_n is 1 if node $n \in nodes(p)$ and 0, otherwise.

The objective function tries to minimize the cost of all selected probes. It satisfies the third and the fourth requirements of the minimum length and minimum number of probes, respectively. The constraint enforces that each node n in probe p should get probed at least c times, thereby satisfying the second requirement (average coverage). The value of c can be tuned based on the system requirements.

ALGORITHMS

The above linear program is the Integer Linear Program (ILP). ILP is known to be NP complete. However, each probe should be either picked or left out, so we need integer solutions only.

The naive way to solve an ILP is to simply remove the constraint that x_p is integer, solve the corresponding LP (called the LP relaxation of the ILP), and then round the entries of the solution to the LP relaxation.

The problem can be formulated in many ways and for each version some algorithms are available – both deterministic and heuristic. Some of them are:

- Tabu Search
- Hill climbing
- Simulated annealing
- Reactive search optimization
- Ant colony optimization
- Hopfield neural networks

We have not yet decided which algorithm we will be working with, we will be considering the trade-offs between each of the algorithms and considering the same.

REFERENCES

- Adaptive Monitoring: Application of Probing to Adapt Passive Monitoring - Deepak Jeswani, Maitreya Natu, R. K. Ghosh (Journal of Network and Systems Management – October 2015)
- Adaptive path selection for link loss inference in network tomography applications - Y Qiao, J Jiao, Y Rao, H Ma (PloS one - 2016)