HAL
open science

Efficient fault monitoring with Collaborative Prediction

Dawei Feng, Cecile Germain-Renaud, Tristan Glatard

To cite this version:

Dawei Feng, Cecile Germain-Renaud, Tristan Glatard. Efficient fault monitoring with Collaborative Prediction. journées scientifiques mésocentres et France Grilles 2012, Oct 2012, Paris, France. hal-00766086

HAL Id: hal-00766086

https://hal.science/hal-00766086v1

Submitted on 17 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient fault monitoring with Collaborative Prediction

Dawei Feng (1), Cécile Germain-Renaud (1), Tristan Glatard (2)
(1) Laboratoire de Recherche en Informatique, CNRS − Université Paris-Sud
(2) CREATIS, CNRS − INSERM −Université Lyon 1 − INSA Lyon

Overview

Isolating users from the inevitable faults in large distributed systems is critical to Quality of Experience. We formulate the problem of probe selection for fault prediction based on end-to-end probing as a Collaborative Prediction (CP) problem. On an extensive experimental dataset from the EGI grid, the combination of the Maximum Margin Matrix Factorization approach to CP and Active Learning shows excellent performance, reducing the number of probes typically by 80% to 90%.

Problem statement

The recent crash of the Amazon Cloud [1] highlighted the importance of timely discovery of failures in large-scale distributed systems: a local, limited error may result in a global catastrophe. Thus a significant part of the software infrastructure of large scale distributed systems, whether grids or clouds, collects information (monitoring) that will be exploited to discover (knowledge) if, where, and when the system is faulty.

This paper addresses the knowledge-building step in the context of end-to-end probing as the class of monitoring techniques. In the end-to-end probing approach, a probe is a program launched from a reliable entry point (probe station), which tests the availability of the components on its path. The only observables are the outcomes of the probes, which are binary: success or failure. For instance, a ping command would test the overall software stacks and network connectivity from the probe station to the target computer. This abstract summarizes our approach to minimize the number of probes for a given discovery performance target.

Minimizing the number of probes can be addressed along three avenues: fault prediction, detection and diagnosis. In all cases, the system under consideration is a set of hardware and software components, featuring some dependencies – e.g. a service certainly depends on the hardware it is running on, and possibly of other services. The obvious advantage of detection and diagnosis is that they provide an explanation of the failure, by exhibiting culprits. On the other hand, they strongly rely on *a-priori* knowledge, namely which components are required for a probe to succeed. For massively distributed systems, where Lamport's famous definition "*A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable*" applies, assuming such knowledge might be hazardous in principle. Instead, this paper focuses on fault prediction. In this case, the overall infrastructure is a black box, with no *a-priori* knowledge of its structure.

Methods

A. Motivation for Collaborative Prediction

The motivating application comes from operations management in the European Grid Initiative (EGI), and specifically the Biomed Virtual Organization. Biomed has access to 256 Computing Elements (CEs) and 121 Storage Elements (SEs). CEs are shares of computing resources, implemented as queues of each site manager (e.g. PBS), and SEs are shares of storage resources; the formal definition is part of the Glue Information model [3]. The probes test the availability of all relations between its endpoints, namely Computing Elements (CEs) and Storage Elements (SEs).

Testing the availability of all CE-SE pairs is one of the most challenging issues encountered daily by monitoring operators. Brute force, that is periodically launching a fully distributed all-pairs availability test, for a total of 29512 tests, multiplied by the number of capacities to test at each run, is both intrusive and useless: human operators cannot handle so many results; in practice, only a few issues are reported, with questionable selection criteria. With our method, a massive reduction of the number of tests provides nearly similar availability evaluation performance, creating opportunities for better frequency/intrusiveness trade-off and selection of reported incidents.

In this context, fault prediction can be considered as a case for Collaborative Prediction (CP). CP is originally a technique for predicting unknown ratings of products for a particular user, based on observed data from other users and products. The success of CP relies on the hypothesis of a factorial model: hidden and partially shared factors affect the ratings. In our case, two nodes (CE or SE) may share several hidden factors - e.g. location, with the associated network connectivity issues, or use of a particular instance of any middleware service (e.g. brokering, authentication), such that the availability of the CE-SE relation may be affected similarly.

B. Selection and prediction

Minimizing the number of probes encompasses two issues: probe selection, *i.e.* which subset of the (CE,SE) pairs to actually test; and predicting the availability of all pairs from the outcome of these probes.

Probe selection.

We consider two probe selection strategies.

- *Static-Uniform*. The probes are selected uniformly at random. In this setting, the prediction step has no influence over the choice of the probes.
- *Active Probing*. With Active Probing (*Algorithm* 1), the set of probes is constructed dynamically, with an initial set of probes selected for instance by the Static-Uniform method, and run through the system to get basic information; then, additional probes are selected and launched with the goal of maximizing some measure of information; here we used the min-margin heuristic [12], which chooses the probe where the uncertainty of the classification result is maximal, and has been demonstrated to be efficient for CP problems [13].

input : Initial partially observed binary(-1/+1) matrix M_0, threshold λ, max # of new samples N, active-sampling heuristic h
output : Full binary-valued matrix M^{T_i} predicting unobserved entries of M_0
initialize: Initialize the vars
$S(T_0) = S(M_0)$ /*currently observed entries set*/ ;
$i = 0$ /*current iteration times*/ ;
$n = 0$ /*current number of new samples*/ ;
while $(n < N)$ **do**
  $M^{T_i} = StandardMC(S(T_i))$ /*Prediction based on observed entries via standard MC procedure*/ ;
  $S'(T_i) = ActiveSampling(M^{T_i}, h, \lambda)$ /*Actively choose the next set of new samples and query their labels*/ ;
  $S(T_{i+1}) = S(T_i) \cup S'(T_i)$;
  $n = n + \#S'(T_i)$;
  $i = i + 1$;
end

Algorithm 1: Generic active probing algorithm

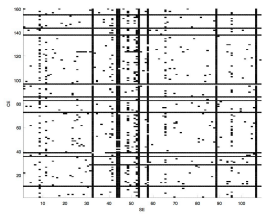
Collaborative Prediction with MMMF

This section sketches Maximum Margin Matrix Factorization as proposed by Srebro et al. in [2]. CP is formalized as a matrix completion problem: if Y is an observed (sparse) matrix, the CP problem is to find a full, real-valued, matrix X of the same size that approximates Y, i.e. that minimizes the "discrepancy" between X and Y, without any external information. Assuming a linear factor model, where k hidden factors define the user preference through a linear combination of them, X is constrained to be of rank k. However, bounding k to small values (low-rank approximation) does not lead to feasible optimization problems for a partially observed matrix and for the binary setting. The key insight in MMMF is to replace the bounded rank constraint with bounding the trace norm of X, under the constraint of no (hard-margin), or small (soft-margin), discrepancy.

Experimental setting

Different capabilities have to be tested; in the following, we consider three of them: probe *srm-ls* tests the list ability from a CE to a SE, probe *lcg-cr* tests the read ability from a CE to a SE, and probe *lcg-cp* tests the write ability alike. Thus, each CE works as a probe station, launching probes to test the functionalities between itself and each SE. For the Biomed grid a whole set of 29512 testing transactions were launched each day for each of the three probe classes. After nearly two months running, information for 51 validated days were collected. The probes themselves are gLite jobs, run by a regular Biomed user. Some of them fail (rejection) in the sense that gLite is not able to complete the job, denoting that some job management services may be down or misconfigured (e.g. authentication, brokering etc.). In the following, we consider only the accepted probes, i.e. those which run to completion, reporting success or failure; this approach amounts to consider that the data access capacities are independent from job management. This is a reasonable hypothesis in a gLite infrastructure because file transfers involved in job management use dedicated storage space independent from the one tested by our probes.

Figure 1 illustrates the structure for lcg-cr on day'07-05-2011', where rows represent CEs and columns stand for SEs. Each entry is the probe result between the corresponding CE and SE. Black columns correspond to sustained SE downtimes while black lines are CE failures leading to complete inability to communicate with any SE (e.g. network downtime or configuration issue). These are usually easily detected and reported by human operators with only a few incident reports. The scattered points, however, correspond to local or transient issues, which are very difficult to handle due to the amount of incident reports independently generated.



It is thus worth assessing the performance of the methods when sustained systematic errors are eliminated. To do that, we designed a second set of experiments, with curated matrices as the reference fault structure. A curated matrix is a new original matrix, where the lines and columns with only failed entries (black ones in figure 2) have been removed prior to analysis.

Experimental results

A. Static-Uniform. Figure 2 shows the accuracy (ratio of correctly predicted entries over total number of entries) for five randomly selected days. An excellent performance can be reached with a tiny fraction of the original probes, typically 5%. The baseline, a random guess following the distribution of the sample set, is plotted for comparison purpose, but can be computed *a-priori*. Figure 3 is the classical visualization of the confusion matrix in the ROC space for all the 51 days at 90% deletion rate (keeping 10% of the probes). Perfect prediction would yield a point in the upper left corner at coordinate (0,1), representing 100% sensitivity (no false negatives) and 100% specificity (no false positives).

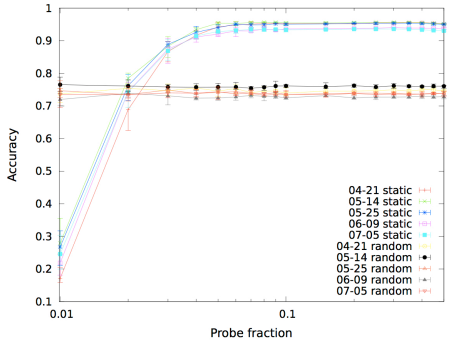


Figure 2. Accuracy for the Static-Uniform selection (five days)

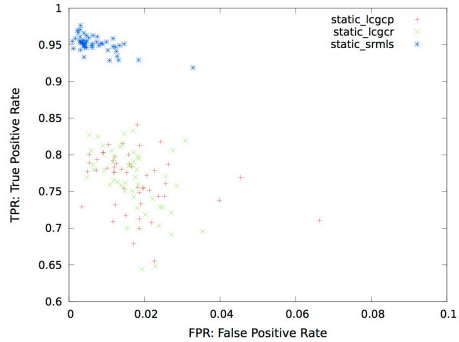
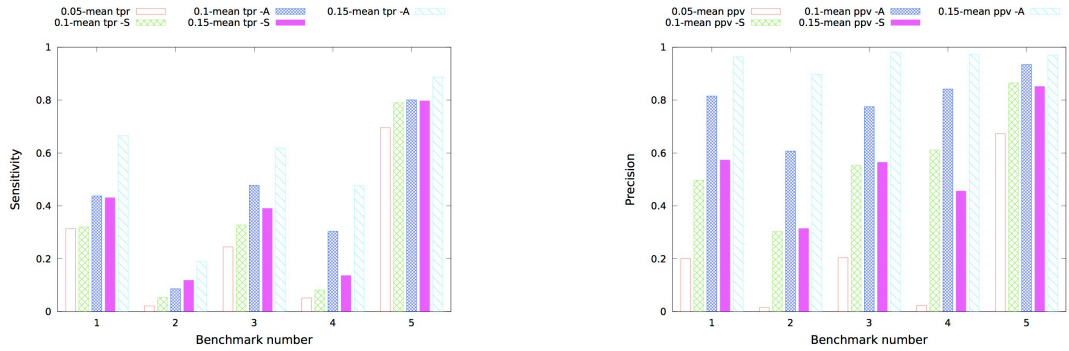


Figure 3. ROC values for for the Static-Uniform selection (all 51 days)

The *srm-ls* dataset shows nearly perfect prediction performance, being mostly very close to (0,1); *lcg-cp* and *lcg-cr* are vastly better than a random guess, which lies on the diagonal line (note the range of the axes, which cover only the part of the ROC space where the results belong, thus the diagonal line is not visible on the plot). The other classical indicators also show excellent performance: Area Under ROC Curve (AUC) as well as MCC (Matthews Correlation Coefficient) are close to 1.

The problem becomes much more difficult when the systematic faults are excluded, thus taking the curated matrices as inputs. The prediction accuracy remains excellent; however, as the number of failed entries left in the curated matrices is much less than in the un-curated ones, e.g. the fraction of failed entries on *srm-ls*, 04-21-2011 drops from 45.37% to 2.25%, accuracy is not meaningful: predicting all entries as negative would give a similar result. The strategy to tackle this issue is Active Probing.

B. Active probing. In this experiment, we compare the Active Probing strategy with the Static one at equal probing cost: first, a Static- Uniform method is applied, in order to get the reference information, then more probes are selected with the min- margin heuristic for Active Probing, while for the Static- Uniform method, the same number of probes are selected uniformly at random.



(a) Sensitivity *(b) Precision*

Figure 4. Performance comparison for Static-uniform and Active Probing, curated srm-ls for the five benchmark days.

Active Probing does improve accuracy over Static-Uniform. However, as explained in the previous section, the quality of failure prediction is the most important goal in this context. Figure 4 compares two relevant indicators: sensitivity (the probability of detecting an actual failure), and precision (the closer to 1, the smaller the risk of a false alarm). They are detailed for the initial probe fraction equal to 5%, then adding probes by step of 5% fractions. The results as given for a total of 10% and 15% probes. The first important result is that Active Probing always outperforms Static-Uniform. In fact, the performance of the Static-Uniform strategy is bad, except for 07-05-2011 (day 5). The performance bears some relation with the failure rate of the benchmark (table I): larger failure rates in the original curated matrix help uncovering the structure of the faults, even at quite low levels: with 4% failure rate, the 07-05-2011 benchmark exhibits acceptable performance when keeping only 5% of the probes and the Static-Uniform strategy. However, the failure rate does not tell the full story: days 2 and 4 have the same low one (1%), but the performance on day 4 is much better than on day 2. The likely explanation is that faults at day 2 do not present much correlation, while faults day 4 derive from a small number of shared causes.

The second result is that Active Probing is quite efficient. We have a good chance of predicting the actual failure, except on day 2, while the false alarm rate remains negligible.

A third strategy, omitted here for brevity, harnesses active probing with a differentiated cost policy [6, 7], where the false positives are more penalized. In all case, except day 2, sensitivity increases by 10-20%, while precision remains almost unharmed. A complete description and more experimental results can be found in [8].

Related Work

Collaborative Prediction associated with end-to-end probing, with the components structure considered as a black box,

participates in the general Quality of Experience approach [9]. More precisely, an important ingredient separating QoE from QoS is binary (possibly extended to discrete) classification. Most work in this area is devoted to network- based services (e.g. among many others [10]). Before QoE became a popular keyword, Rish and Tesauro [5] explored the combination of MMMF and Active probing for the selection of good servers in various distributed and P2P systems. Our work combines the goal of proposing fault-free services to the user exemplified in [4], [11], and the CP approach of [5]. Z. Zheng and M.R. Lyu propose explicit users collaboration for estimating failure probabilities [11]; while their end-to-end framework is related to ours, the goal is more in line with QoS (estimating the full distribution of probability instead of binary classification), and the correlation amongst users or services is modeled by ad hoc methods. To our knowledge, this work is the first one to exemplify MMMF on fault prediction.

Perspectives

The Achille's heel of large-scale production grids and clouds is reliability. Quality of Experience at reasonable human cost requires extracting the hidden information from monitoring data whose intrusiveness should be limited. Collaborative Prediction is one of the promising and scalable strategies that can address this goal. This paper demonstrates its effectiveness on a large experimental dataset. Further work will consider two avenues. Exploiting the resulting knowledge – the hidden causes uncovered by CP – towards diagnosis might be easier with alternative matrix factorization techniques; specifically Bi-LDA [12] provides a quantified model of latent factors, which, like LDA, could be exploited for interpretation [13] more easily than MMMF, but suffers from lower performance. The second avenue targets improving prediction by separating transient failures from more permanent non- systematic ones, with temporal models inspired from the hierarchical Hidden Markov Models approach of text mining.

Références

[1] H. Blodget. (2011, April) Amazon's cloud crash disaster permanently destroyed many customers' data. Business Insider. [Online]. Available: http://www.businessinsider.com/amazon-lost-data-2011-4

[2] N. Srebro, J. D. M. Rennie, and T. S. Jaakola, "Maximum-margin matrix factorization," in Advances in Neural Information Processing Systems 17, 2005, pp. 1329–1336.

[3] S. Andreozzi and al., "Glue Schema Specification, V.2.0," http://www.ogf.org/documents/GFD.147.pdf, Tech. Rep., 2009.

[4] I. Rish and al., "Adaptive diagnosis in distributed systems," IEEE Trans. Neural Networks, vol. 16, no. 5, pp. 1088 – 1109, 2005.

[5] I. Rish and G. Tesauro, "Estimating end-to-end performance by collaborative prediction with active sampling," in Integrated Network Management, 2007, pp. 294–303.

[6] K. Morik, P. Brockhausen, and T. Joachims, "Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring," in 16th Int. Conf. on Machine Learning, 1999, pp. 268–277.

[7] Y. Lin, G. Wahba, H. Zhang, and Y. Lee, "Statistical properties and adaptive tuning of support vector machines," Machine Learning, vol. 48, pp. 115–136, September 2002.

[8] D. Feng, C. Germain-Renaud and T. Glatard. "Distributed Monitoring with Collaborative Prediction". In 12th IEEE International Symposium on Cluster, Cloud and Grid Computing (CCGrid'12) 2012, pp 376-383.

[9] H. Rifai, S. Mohammed, and A. Mellouk, "A brief synthesis of QoS- QoE methodologies," in 10th Int. Symp. on Programming and Systems, 2011, pp. 32–38.

[10] H. Tran and A. Mellouk, "QoE model driven for network services," in Wired/Wireless Internet Communications, ser. LNCS, 2010, vol. 6074, pp. 264–277.

[11] Z.ZhengandM.R.Lyu,"Collaborative reliability prediction of service- oriented systems," in 32nd ACM/IEEE Int. Conf. on Software Engineering, 2010.

[12] I. Porteous, E. Bart, and M. Welling, "Multi-hdp: a non parametric bayesian model for tensor factorization," in 23rd Conf. on Artificial Intelligence, 2008, pp. 1487–1490.

[13] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Procs of the National Academy of Sciences, vol. 101, pp. 5228–5235, 2004.