Performance Evaluation of a Distributed and Probabilistic Network Monitoring Approach

R. Steinert and D. Gillblad
Industrial Applications and Methods Lab (IAM)
Swedish Institute of Computer Science (SICS)
SE-164 29 Kista, Sweden
Email: {rebste, dgi}@sics.se

***Abstract*—We investigate the effects of employing a probabilistic fault detection approach relative the performance of a deterministic network monitoring method. The approach has its foundation in probabilistic network management, in which performance limits and thresholds are specified in terms of e.g. probabilities or belief values. When combined with adaptive mechanisms, probabilistic approaches can potentially offer improved controllability, adaptivity and reliability, compared to deterministic monitoring methods. Results from synthetically generated and real network QoS measurements indicate that the probabilistic approach generally can perform at least as good as a deterministic algorithm, with a higher degree of predictable performance and resource-efficiency. Due to the stochastic nature of the algorithm, worse performance than expected is sometimes observed. Nevertheless, the results give additional support to some of the practical benefits expected in using probabilistic approaches for network management purposes.**

***Index Terms*—probabilistic network management; adaptive fault detection; network monitoring**

I. Introduction

In specific network environments, deterministic monitoring approaches can be configured with exact precision and operative control, given relatively detailed knowledge about the system behavior and the symptoms to be detected. Deterministic methods can be designed to be somewhat adaptive in controlled or known network situations, but do not handle uncertainty such as noise and small variations very well. This means for example that configurations that are tailor-made for similar network environments are likely to require additional fine-tuning for best monitoring performance. With respect to the development of increasingly complex networks (such as Internet of Things, cloud networking and mobile networks), the limitations of deterministic methods will likely lead to less efficient performance and usage of network resources, thereby contributing to unnecessary management costs.

The development of probabilistic network management (PNM) approaches target the challenges inherent to the development of complex future networks, such as scalability, controllability, adaptivity and reliability [1]. A PNM method is based on probabilistic modeling of observed network behavior, and might involve probabilistic input and output. Algorithmic behavior may for example be configured in terms of probabilities or distributions, reflecting a certain degree of uncertainty about the environment in which it operates, whereas the output may be provided as e.g. probability distributions. As such methods are more robust to network variations and uncertainty in terms of both algorithmic behavior and setup, it is likely that the development of network monitoring approaches to a higher degree will follow a probabilistic paradigm [1].

We here investigate the effects in employing a PNM approach compared to deterministic probing, with respect to adaptivity, predictability, resource utilization and configuration efficiency. For these purposes, we study and evaluate different performance aspects of a probabilistic fault detection approach (PFD) [2], and a deterministic fault detection (DFD) method based on heartbeat monitoring. The latter type of monitoring is often employed in current network management systems, using various OAM probing tools (e. g. [3], [4]).

A. Related Work

Following the development of complex networked systems, distributed and probabilistic approaches have started to emerge to a higher degree. Pras et al list the challenges of future networks, and discuss distributed probabilistic modeling as an important development for efficient handling of uncertainty in network management [5]. Similarly, the authors in [1] further identify necessary properties of PNM approaches and the requirements on scalability, adaptivity, controllability and predictability, followed by algorithm examples.

As an example of PNM algorithms with adaptive behavior, Brunner et al [6] propose a probabilistic approach based on random activation and deactivation of management monitoring functions, capable of resource-efficiently preserving accuracy in observed network behavior and algorithm performance. Another example is a probabilistic self-organizing, selective management scheme for clustering of nodes in ad-hoc networks, based on probabilities of spatio-temporal connectivity [7]. Probabilistic approaches are also often used for characterizing network connections based on probing measurements, by performing end-to-end measurements [8], [9] or by comparing measurements over shared connections [10].

In terms of adaptive fault and anomaly detection, Hajji et al propose a parametric approach to model the differences in parameter estimates for the purpose of detecting anomalies based on the log-likelihood ratio [11]. Huang et al provide a decentralized method for anomaly detection of unusual traffic loads, employing global PCA on locally filtered data

978-3-901882-48-7

streams [12]. The detection threshold adapts relative observed data and the tolerable relative error of eigenvalues, allowing for predictable performance in terms of false positives. Further, Agosta et al propose a distributed probabilistic method for worm traffic detection based on adaptive thresholding [13]. The threshold adapts based on the predicted class distributions of observed traffic and the tolerable amount of false positives.

Following the definitions and requirements in [1], our distributed method samples distributions of observed packet loss and link delays [2], [14]. Similarly to previous work [12], [13], algorithm parameters are adapted to observed measurements and high-level objectives, allowing for predictable and more efficient performance compared to deterministic methods.

II. Probabilistic Network Monitoring

The distributed and probabilistic fault detection approach has already been presented in previous work [2]. In the previous paper the detection performance was evaluated, whereas in this paper we evaluate the properties of the method relative the concept of PNM. Similarly, a change detection approach based on overlapping estimators was evaluated in [14], focused on finding changes in observed probe response delays. Here, we extend the fault detection approach with the adaptive properties of overlapping estimators used in [14]. The combination of probabilistic modeling and adaptive mechanisms makes the PFD approach adaptive in algorithmic behavior, while providing up-to-date estimates in varying network environments. In addition, the PFD can be implemented and used with existing OAM measurement tools, and may be deployed in network equipment with limited computational capabilities, as the computational requirements are relatively low.

In [2] detection of connectivity problems between two nodes was addressed using probes. Based on measurements of packet loss and probe reply delays, a model of the probability of obtaining a probe reply R is constructed as follows:

$$F(R) = (1 - p_D) \int_0^{\Delta t} f(t; \alpha, \beta) dt, \quad (1)$$

where Δt is the expected delay, p_D the observed drop rate modeled as a Bernoulli distribution, and $f(t; \alpha, \beta)$ the probability density function for a Gamma distribution. Parametric modeling may be less robust to model deviations in the data compared to non-parametric approaches, but may allow for more accurate predictions in behavior. The specific choices of distributions are supported by previous studies (e.g. [8], [9], [15]). From the inverted cumulative density function $F^{-1}(\tau; \alpha, \beta)$ the probabilistic threshold $\tau \in [0, 1]$ specifies an upper limit on the longest response delay tolerated before a probe is considered as lost, as well as the probing interval.

To detect faults with high certainty, we assume that the joint probability of *not* receiving *any* response R given a set of statistically independent probes in a *detection trial* is

$$P(\neg R|\Delta t^{(1)}, \Delta t^{(2)}, \ldots, \Delta t^{(n)}) = \prod_{i=1}^{n} (1 - F(R_i)) < \psi \quad (2)$$

The detection trial is stopped either when a probe response is obtained or when $P(\neg R|\Delta t^{(1)}, \Delta t^{(2)}, \ldots, \Delta t^{(n)})$ reaches below the detection threshold ψ. Thus, the parameter ψ controls the number of probes needed to decide whether a connectivity problem is encountered, and can be regarded as a limit on the highest acceptable rate of false positives [1].

Additional mechanisms based on overlapping estimators [14] enable long-term adaptation to variations in observed probe response delays and packet drop. Each estimator Θ models the current behavior of the observed delay or drop for the duration of N samples while overlapping the next estimator with $0 < T < N$ samples. The estimation of the Gamma parameters $\Theta = \{\alpha, \beta\}$ is based on the method of moments. For the Bernoulli distribution the success rate parameter $\Theta = (1 - p_D) = p$ is based on Bayesian parameter estimation. Each new Θ use a prior estimate from the previous estimator corresponding to first T samples.

III. Experiments and results

In three categories of experiments, we have tested the performance control, behavioral adaptation and configuration efficiency. The experiments are based on 10^5 synthetically generated measurements of probe response delays and packet drops from Gamma and Bernoulli distributions. The Gamma parameters α, β were randomly drawn from a Gaussian distribution and were allowed to vary based on a fraction σ and the base settings $\alpha_b = 2, \beta_b = 11$. Similarly, the success rate parameter p for the Bernoulli distribution was allowed to vary around $p = 0.9$. The overlapping estimators of the PFD were configured setting $N = 1000$ and $T = 500$. The DFD was implemented and configured using a probing interval δ and a threshold ϕ corresponding to the number of lost probes required until detection. It is assumed that both methods operate in a fully distributed manner. For comparison purposes, all settings of δ correspond to $F^{-1}(\tau; \alpha_b, \beta_b)$. Measurements larger than the probing interval are regarded as dropped, and we assume stationary network behavior.

In a fourth category, real uni-directional round-trip measurements were used for offline tests. The datasets were obtained by using IP Ping (with ~ 1 second intervals) in three different settings - 1) direct measurements between two hosts in a wired LAN; 2) remote measurements between two hosts on different networks (where the measuring host was wirelessly connected and the responding host was wired); and 3) direct probing between two wireless hosts connected ad-hoc. The datasets are of different sizes (282369, 201660 and 129296 samples) and contain 0, 116 and 2645 number of lost probes, respectively.

A. Performance control

We study the performance of the two detectors in terms of the predicted false positive rate. The PFD results were obtained by varying $\tau = \{0.1, \ldots, 0.9\}$ and $\psi = \{10^{-6}, \ldots, 10^{-2}\}$. The DFD was configured by varying the probing interval δ corresponding to τ as described above, and the detection threshold $\phi = \{2, \ldots, 18\}$. The experiments were repeated 200 times for one synthetic link using $\sigma = 0.0001$.

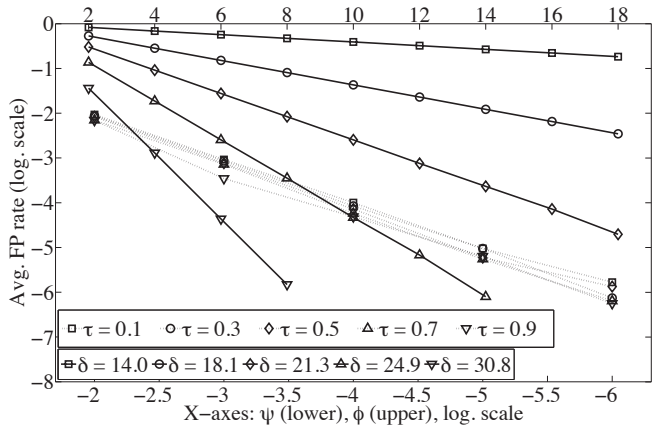

Fig. 1: Rate of false positives for varying detection thresholds ψ and ϕ for the PFD (dotted line) and DFD (solid line), respectively.

In Figure 1, we observe that the rate of false positives for the fault detector closely follows the setting of ψ for any setting of τ. The effect of different τ for a fixed ψ is that the number of lost probes until detection adapts to the requirements on detection certainty, specified by the fraction of acceptable false alarms (Figure 2d, section III-B). For the DFD approach, we see that the performance varies with the deterministic settings of δ and ϕ, thereby indicating less control in predicted performance compared to the probabilistic fault detector.

B. Behavioral adaptation

We investigate the effects of the adaptation properties of the distributed PFD approach. In a first experiment, the link load from produced probing traffic is studied for varying probing intervals, and compared to a baseline interval corresponding to the DFD. For each τ, σ, a simulation using a synthetically generated network consisting of 1000 connections (with randomly drawn QoS parameters) was performed for a simulated time of 24 hours. The probing intervals for the fault detector were determined by $mF^{-1}(\tau)$ and multiplied by $m = 227$ to approximately match the baseline DFD probing interval, $\delta = \alpha_b \beta_b m \approx 5000$ ms. In a second experiment, we study the adaptation of probes needed to match different ψ relative randomly drawn drop rates p_D and fixed α_b, β_b, computed as $\log_{10}(\psi)/\log_{10}(1-\tau(1-p_D))$, across 1000 connections.

In Figure 2a we observe that number of packets over the entire network varies with τ. As an effect of adapting the probing interval to the observed delay on each link (Figure 2b) the traffic load caused by monitoring can be significantly reduced, compared to when using fixed probing intervals. Moreover, the number of detection probes adapts to the observed drop rate for each link to match the setting of ψ (Figure 2c). The effect is that for lossy links, more probes are required to detect a disturbance with the certainty specified by ψ. As an additional example, we see that the length of the detection trial also depends on τ (Figure 2d). Too short probing intervals are thus compensated for with longer detection trials, as low values of τ may increase the number of false positives.

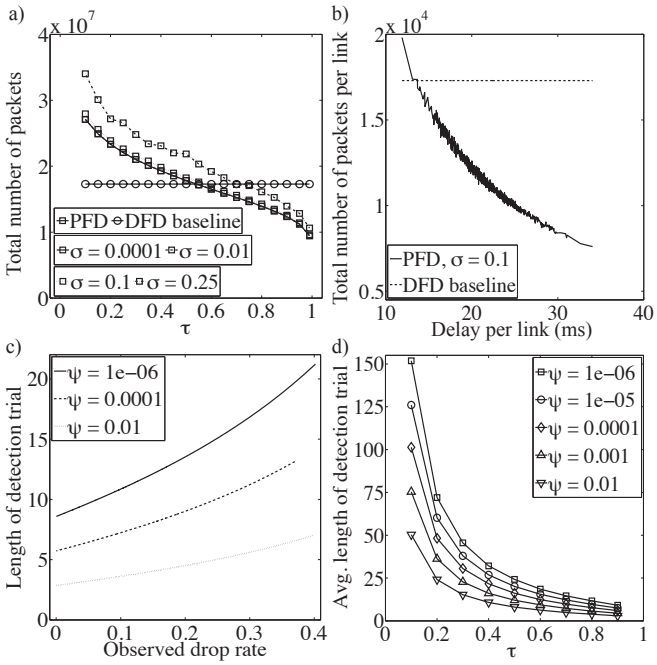

Fig. 2: In a) network traffic adaptation relative τ; in b) an example of network traffic adaptation per link for fixed $\tau = 0.95$ and $\delta \approx 5000$ ms; in c): adaptation of detection trials relative drop rates when $\sigma = 0.1$ and $\tau = 0.8$; in d): probe adaptation when varying τ, ψ.

C. Configuration efficiency

We investigate the configuration efficiency in terms of the rate of successful configurations and the number of configuration attempts needed to detect a randomly generated disturbance, with fewest false positives. A disturbance was simulated as a sequence of lost probes, randomly generated from a Poisson distribution with average length $\lambda = \{5, \ldots, 100\}$. In a first step, the detection threshold was isolated using binary search, testing the performance at each division. The search intervals were set $\psi = \{0, 1\}$ and $\phi = \{1, 10^5\}$ (using fixed $\tau = 0.9, \delta = 30.8133$), thus representing high uncertainty or lack of prior knowledge about the expected disturbance. In a second step, the probing interval of each method was determined by gradually halving the interval of $\tau = \{0.000001, 0.999999\}$ and $\delta = \{0, 75.4325\}$ towards the upper limit. The second search (using the result from the first search) was only carried out as a refinement step in the case false positives were not eliminated (or if the true disturbance remained undetected) in the first step. The searches were terminated either when the solution was found or until the interval became less than $\epsilon = 10^{-8}$. The results are based on the average of 200 runs. Although configuration may be done differently in practice, the testing method was designed w.r.t. the different types of discrete and continuous parameters, and for the purpose of imitating an alternating approach in which parameters are manually set followed by empirical testing.

In Figure 3a we observe that the success rate in general is higher for the PFD method than the DFD, specifically

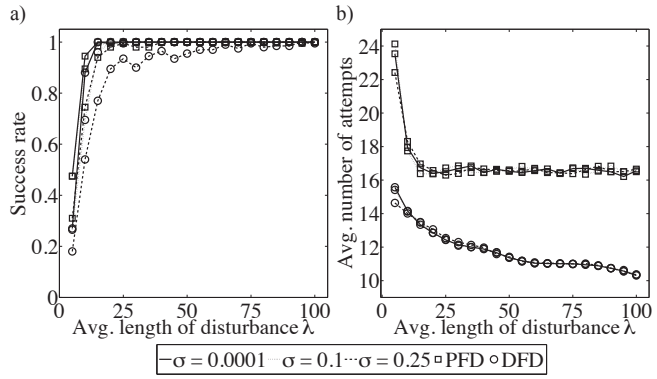

Fig. 3: In a): the rate of successful configurations finding the true disturbance without false positives; in b): the average number of attempts to achieve a successful configuration.

for shorter disturbances. Different σ indicate that in highly dynamic network environments, satisfactory configurations may require further efforts. As the disturbance becomes more prominent relative the observed drop rate p_D, the number of attempts levels out for the PFD, whereas it gradually decreases for the DFD (Figure 3b). The number of attempts is here dependent on the search intervals, and in general the algorithm performance is a trade-off between the desired performance and acceptable configuration efforts. Additionally, the search approach used is here more favorable to the DFD, as the number of attempts is limited relative to the discrete ϕ and the datastream length. Although the number of attempts is lower for the DFD, the solutions obtained for the PFD vary around $\tau \approx 0.9$ and $\psi \approx 10^-5$. This suggests that assuming no or limited prior knowledge about the network behavior, the PFD could perform satisfactory with less configuration efforts, based on knowing only the approximate probabilistic levels.

D. Performance using real data

Finally, we test the detectors on real measurements obtained from different network environments. As the moment estimates can be sensitive to outliers, the data was processed using a low-pass filter based on the mean and standard deviation, as $\mu + m\sigma$ where $m = 3$. The parameters τ,δ and ψ, ϕ were varied to test the behavior when using the filtered data as input. For the PFD, the actual probing intervals were computed based on τ after each observation not regarded as dropped. The DFD interval δ was for comparison reasons set to the average probing interval corresponding to a certain τ for each dataset, shown in Table I. As the datasets contain few or no packet losses, probing intervals were set directly as $mF^{-1}(\tau)$ with $m = 1$ to generate more false positives for analysis.

We observe in Table I that the average estimated probing interval adjusts to different settings of τ and datasets. Similarly, we see in Figure 4a that the length of the detection trials adjusts with different settings of τ and ψ. Furthermore, the observed DFD false positive rates are more scattered relative the PFD, indicating a higher degree of predictability in the PFD case (Figures 4b,c and d). In general, the results also suggest that configuring the DFD would to a higher degree require prior knowledge about individual link delays for more efficient resource utilization, compared to the PFD.

τ	0.55	0.65	0.75	0.85	0.95
Wired	0.1096	0.1147	0.1206	0.1283	0.1419
Wifi ad hoc	0.2919	0.5725	1.1824	2.6266	7.3563
Wifi remote	29.9524	30.3532	30.8038	31.3744	32.3499

TABLE I: Avg. probing intervals (ms) for different datasets and τ.

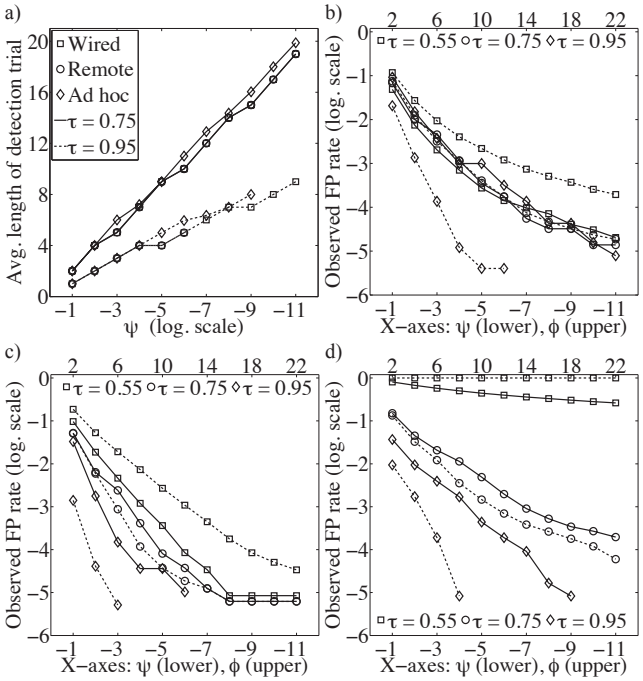

Fig. 4: Performance when using real datasets. In a): The avg. detection probe length; in b), c) and d): FP-rates for PFD (solid line) and DFD (dashed line) relative varying parameters, when using 'Wired','WIFI remote' and 'WIFI ad hoc', respectively, as input.

IV. Concluding Remarks

Compared to a simpler probing approach operating under deterministic limits, the distributed and adaptive PFD approach can perform with improved predictability and reliability in algorithm behavior, while being more resource-efficient and robust to noise. However, as indicated by the results, probabilistic approaches may sometimes perform worse than expected, due to e.g. model deviations or too few samples. As probabilistic limits allow for simplified configuration, approximate knowledge about the network behavior may be sufficient for achieving acceptable detection performance, which might contribute to lower network management costs.

Future work include further development of probabilistic monitoring approaches and possibly the implementation of a PNM protocol. Finally, despite the indicated benefits of using probabilistic algorithms (in this and previous work), the movement towards overall employment of PNM algorithms may not be trivial, as it likely requires revised practices and views on network management in general, and specifically with respect to how probabilities are used and interpreted [1].

REFERENCES

[1] A. Prieto, D. Gillblad, R. Steinert, and A. Miron, "Toward decentralized probabilistic management," *Communications Magazine, IEEE*, vol. 49, no. 7, pp. 80 –86, July 2011.

[2] R. Steinert and D. Gillblad, "Towards Distributed and Adaptive Detection and Localisation of Network Faults," in *2010 Sixth Adv. Int'l Conference on Telecommunications, AICT*. IEEE, 2010, pp. 384–389.

[3] Recommendation, "ITU-T Recommendation Y.1731," *OAM functions and mechanisms for Ethernet based networks*, 2008.

[4] R. Hofstede, I. Drago, G. Moura, and A. Pras, "Carrier Ethernet OAM: an overview and comparison to IP OAM," *Managing the Dynamics of Networks and Services*, pp. 112–123, 2011.

[5] A. Pras, J. Schonwalder, M. Burgess, O. Festor, G. Pérez, R. Stadler, and B. Stiller, "Key research challenges in network management," *Communications Magazine, IEEE*, vol. 45, no. 10, pp. 104–110, 2007.

[6] M. Brunner, D. Dudkowski, C. Mingardi, and G. Nunzi, "Probabilistic decentralized network management," in *Integrated Network Management, 2009.IFIP/IEEE Int'l Symp. on*. IEEE, 2009, pp. 25–32.

[7] R. Badonnel, R. State, and O. Festor, "Probabilistic management of adhoc networks," in *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*. IEEE, 2006, pp. 339–350.

[8] J. Bolot, "End-to-end packet delay and loss behavior in the Internet," *ACM SIGCOMM Computer Communication Review*, vol. 23, no. 4, pp. 289–298, 1993.

[9] A. Mukherjee, "On the dynamics and significance of low frequency components of Internet load," *Technical Reports (CIS)*, p. 300, 1992.

[10] A. Coates, A. Hero III, R. Nowak, and B. Yu, "Internet tomography," *Signal Processing Magazine, IEEE*, vol. 19, no. 3, pp. 47–65, 2002.

[11] H. Hajji, "Statistical analysis of network traffic for adaptive faults detection," *IEEE Trans. on Neural Netw.*, vol. 16, pp. 1053–1063, 2005.

[12] L. Huang, X. Nguyen, M. Garofalakis, M. I. Jordan, A. Joseph, and N. Taft, "In-Network PCA and Anomaly Detection," in *Advances in Neural Information Processing Systems 19*. Cambridge, MA: MIT Press, 2007, pp. 617–624.

[13] J. Agosta, C. Diuk-Wasser, J. Chandrashekar, and C. Livadas, "An adaptive anomaly detector for worm detection," in *Proceedings of the 2nd USENIX workshop on Tackling computer systems problems with machine learning techniques*. USENIX Association, 2007, p. 3.

[14] R. Steinert and D. Gillblad, "Long-Term Adaptation and Distributed Detection of Local Network Changes," in *GLOBECOM*, 2010, pp. 1–5.

[15] M. Kalman and B. Girod, "Modeling the delays of successively-transmitted Internet packets," in *Multimedia and Expo, 2004. ICME'04.*, vol. 3, 2004, pp. 2015–2018.