Multi-domain Diagnosis of End-to-End Service Failures in Hierarchically Routed Networks

Małgorzata Steinder[1] and Adarshpal S. Sethi[2]

[1] IBM T.J. Watson Research Center, Hawthorne, NY
steinder@us.ibm.com
[2] Computer and Information Sciences, University of Delaware, Newark, DE
sethi@cis.udel.edu

Abstract. This paper investigates an approach to improving the scalability and feasibility of probabilistic fault localization in communication systems by exploiting the domain semantics of computer networks. The proposed technique divides the computational effort and system knowledge among multiple, hierarchically organized managers. Each manager performs fault localization in the domain it manages and requires only the knowledge of its own domain. Since failures propagate among domains, domain managers cooperate with each other to find a consensus explanation of the observed disorder. We show through simulation that the proposed approach increases the effectiveness of probabilistic diagnosis and makes it feasible in networks of considerable size [1].

1 Introduction

End-to-end connectivity in a given protocol layer is provided through a sequence of intermediate nodes. Communication problems between a pair of these nodes, e.g., a malfunctioning interface, intermittent connectivity, etc., may disorder one or more end-to-end paths containing the failing link. These end-to-end problems propagate to higher system layers causing various application-level events, such as aborted transactions, session timeouts, or abnormal delays. The diagnosis of end-to-end network service failures [1,2] is a sub-task of fault localization [3,4,5] that isolates node-to-node services responsible for availability or performance problems experienced by end-to-end services. In the previous work [1,2], we investigated an application of probabilistic reasoning to end-to-end service failure diagnosis. The proposed approaches rely on a probabilistic *fault propagation model* (FPM), which represents causal relationships between end-to-end and node-to-node service failures. To solve the fault localization problem, in [1], an adaptation of Pearl's belief updating in belief networks [6] was used, and in [2], a novel algorithm was proposed, which is based on incremental hypothesis updating. The algorithms were shown effective in the diagnosis of end-to-end service failures in networks composed of tens of nodes.

[1] Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

N. Mitrou et al. (Eds.): NETWORKING 2004, LNCS 3042, pp. 1036–1046, 2004.

This paper introduces a distributed fault-localization technique [4,7], which increases the admissible network size by an order of magnitude by taking advantage of the domain semantics of communication systems. The technique divides the computational effort and system knowledge among hierarchically organized managers. Each manager is responsible for fault localization within the domain it governs, and reports to a higher-level manager that oversees and coordinates the fault-localization process of multiple domains. The technique is suitable for distributed diagnosis of end-to-end service failures in hierarchically routed networks such as the Internet. Although the technique is not expected to apply to the management of the entire Internet, we consider it applicable to failure diagnosis across a small subset of network domains that are used to provide a distributed service we want to manage.

The paper is structured as follows. Section 2 defines the problem of probabilistic end-to-end service failure diagnosis. In Section 3, an outline of a multi-domain fault localization technique for hierarchically routed networks is proposed. A distributed fault propagation model is proposed in Section 4, and a multi-domain fault localization algorithm is presented in Section 5. Section 6 evaluates the effectiveness of the proposed multi-domain techniques.

2 Probabilistic Diagnosis of End-to-End Service Failures

This paper adopts a service-oriented view of the network [8], in which end-to-end or node-to-node connectivity in a given protocol layer is considered a service provided by this layer to higher layers. The fault propagation model (FPM) for end-to-end service failure diagnosis is a bipartite causality graph in which parentless nodes (called *link* nodes) represent node-to-node service failures (faults) and childless nodes (called *path* nodes) represent end-to-end service failures (symptoms). Multiple *link* or *path* nodes may exist for every node-to-node or end-to-end service that correspond to different types of failures that may be experienced by the service. Since causal relationships between node-to-node and end-to-end service failures are difficult to determine due to their dynamic and unpredictable nature, the FPM is a probabilistic one, in which each *link* node is labeled with the probability of the corresponding fault's independent occurrence, and causal edges between *link* nodes and *path* nodes are weighted with the probability of the causal implication.

In our previous work, two approaches to solving this problem have been proposed. The first technique (referred to as **Alg. 1**) [1] adapts Pearl's belief updating [6] for polytrees to calculating the most probable explanation (MPE) of observed symptoms. The second approach (**Alg. 2**) [2,1] obtains the MPE by incrementally updating a set of alternative explanation hypotheses. Due to space limitations, we will not present these algorithms in this paper. Instead, we ask the reader to refer to the respective previous publications. We only state that both algorithms include an event-driven procedure called *inference* that analyzes an observed symptom and includes the results of this analysis in the fault-localization state. Also, both algorithms are capable of producing a conditional probability of a given fault's existence, $Prob\{f\}$, or non-existence, $Prob\{\neg f\}$ at any time in the process of fault localization.

Table 1. Basic notation

$n_k \rightarrow n_l$	A directed link from n_k to n_l, where n_k and n_l are node identifiers that are unique network-wide, e.g., IP addresses
$n_{p_1} \overset{*}{\rightarrow} n_{p_m}$	A directed, possibly multi-hop path from n_{p_1} to n_{p_m} consisting of links $n_{p_1} \rightarrow n_{p_2}$, . . ., $n_{p_{m-1}} \rightarrow n_{p_m}$.
$s : n_k \overset{*}{\rightarrow} n_l$	A symptom indicating a failure of path $n_k \overset{*}{\rightarrow} n_l$
$f : n_k \rightarrow n_l$	A fault associated with link $n_k \rightarrow n_l$
$i \overset{*}{\rightarrow} j$	The set of all paths that begin in domain $\mathcal{D}_i$ and end in domain $\mathcal{D}_j$, i.e., $i \overset{*}{\rightarrow} j = \{n_k \overset{*}{\rightarrow} n_l \mid n_k \in \mathcal{D}_i$ and $n_l \in \mathcal{D}_j\}$, where i and j are unique domain identifiers, e.g., IP subnet masks.
$s : i \overset{*}{\rightarrow} j$	A symptom associated with the set of paths $i \overset{*}{\rightarrow} j$. We say that symptom $s : i \overset{*}{\rightarrow} j$ occurred when at least one $s : n_k \overset{*}{\rightarrow} n_l$ occurred such that $n_k \in \mathcal{D}_i$ and $n_l \in \mathcal{D}_j$.

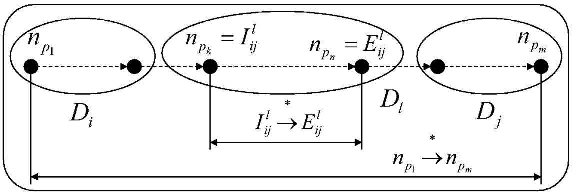

n_{p_1}
$n_{p_k} = I^l_{ij}$
$n_{p_n} = E^l_{ij}$
n_{p_m}
D_i
$I^l_{ij} \overset{*}{\rightarrow} E^l_{ij}$
D_l
D_j
$n_{p_1} \overset{*}{\rightarrow} n_{p_m}$

Fig. 1. Definition of a path segment, and ingress and egress gateways.

3 Multi-domain Approach to End-to-End Service Failure Diagnosis

The multi-domain approach to fault localization in hierarchically routed networks takes advantage of the domain semantics of communication systems. A management domain corresponds to a routing domain and may be identified, for example, by a subnet address. The technique proposed in this paper may be applied in networks with multiple levels of the routing hierarchy. However, for simplicity, we focus on a two-level architecture and use $\mathcal{N}$ and $\mathcal{D}_i$ to denote the entire network and its sub-domain, respectively. Domain $\mathcal{D}_i$ is managed by a separate manager, DM$_i$. At the root of the management hierarchy we place a network manager, NM, which coordinates the operation of managers DM$_i$.

We introduce the notation presented in Table 1. For an end-to-end path $n_{p_1} \overset{*}{\rightarrow} n_{p_m}$ consisting of links $n_{p_1} \rightarrow n_{p_2}$, . . ., $n_{p_{m-1}} \rightarrow n_{p_m}$ we define the following concepts.

Definition 1. *Path $n_{p_1} \overset{*}{\rightarrow} n_{p_m}$* traverses *$\mathcal{D}_i$ iff $\exists n_{p_j} | 1 \leq j \leq m, n_{p_j} \in \mathcal{D}_i$. Path $n_{p_1} \overset{*}{\rightarrow} n_{p_m}$ is an* intra-domain path *in $\mathcal{D}_i$ if $\forall n_{p_j} | 1 \leq j \leq m, n_{p_j} \in \mathcal{D}_i$. If $n_{p_1} \overset{*}{\rightarrow} n_{p_m}$ that traverses $\mathcal{D}_i$ is not an intra-domain path in $\mathcal{D}_i$, it is an* inter-domain path *with respect to $\mathcal{D}_i$.*

Definition 2. *Let $n_{p_1} \overset{*}{\rightarrow} n_{p_m}$ be an inter-domain path with respect to $\mathcal{D}_l$. Let $n_{p_1} \in \mathcal{D}_i$ and $n_{p_m} \in \mathcal{D}_j$. Node n_{p_k} such that $1 < k \leq m$, $n_{p_k} \in \mathcal{D}_l$, and $n_{p_{k-1}} \notin \mathcal{D}_l$ is an*

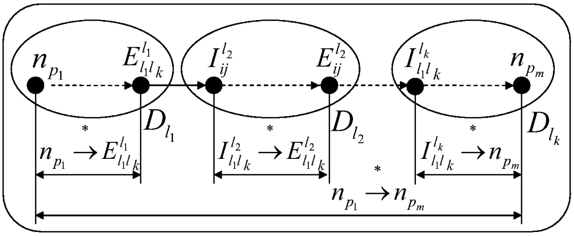

n_{p_1}
$E^{l_1}_{l_1 l_k}$
$I^{l_2}_{ij}$
$E^{l_2}_{ij}$
$I^{l_k}_{l_1 l_k}$
n_{p_m}
D_{l_1}
D_{l_2}
D_{l_k}
$n_{p_1} \overset{*}{\to} E^{l_1}_{l_1 l_k}$
$I^{l_2}_{l_1 l_k} \overset{*}{\to} E^{l_2}_{l_1 l_k}$
$I^{l_k}_{l_1 l_k} \overset{*}{\to} n_{p_m}$
$n_{p_1} \overset{*}{\to} n_{p_m}$

Fig. 2. Transformation of an end-to-end path into a sequence of inter-domain links and intra-domain path segments.

ingress gateway *from $\mathcal{D}_i$ to $\mathcal{D}_j$ in $\mathcal{D}_l$ and is denoted by $I^l_{i,j}$. Similarly, n_{p_n} such that $1 \leq n < m$, $n_{p_n} \in \mathcal{D}_l$, and $n_{p_{n+1}} \notin \mathcal{D}_l$ is an* egress gateway *from $\mathcal{D}_i$ to $\mathcal{D}_j$ in $\mathcal{D}_l$ and is denoted by $E^l_{i,j}$* (Fig. 1).

Consequently, if a node in $\mathcal{D}_l$ is an ingress/egress gateway both from $\mathcal{D}_i$ to $\mathcal{D}_j$ and from $\mathcal{D}_{i'}$ to $\mathcal{D}_{j'}$, then it is denoted by both $I^l_{i,j}/E^l_{i,j}$ and $I^l_{i',j'}/E^l_{i',j'}$.

Definition 3. *Let $n_{p_1} \overset{*}{\to} n_{p_m}$ such that $n_{p_1} \in \mathcal{D}_i$ and $n_{p_m} \in \mathcal{D}_j$ be inter-domain with respect to $\mathcal{D}_l$. Path $I^l_{i,j} \overset{*}{\to} E^l_{i,j}$ is called an* intra-$\mathcal{D}_l$ segment *of $n_{p_1} \overset{*}{\to} n_{p_m}$* (Fig. 1).

The solution proposed in this paper is based on the following assumptions, which are reasonable to make in hierarchically routed networks: (1) domains are disjoint, (2) no path enters the same domain more than once. In addition, to simplify the presentation of the methodology, we consider the case where, at a given point in time, all relevant traffic between two nodes is transfered using the same route (which is equivalent to single-path routing).

Each DM has the minimum knowledge necessary for fault diagnosis, i.e., it is aware of logical topology only in the domain it directly manages. DM$_i$ is aware of link $n_k \to n_l$ iff both n_k and n_l belong to $\mathcal{D}_i$, whereas NM is aware of link $n_k \to n_l$ iff $n_k \to n_l$ is a link between $\mathcal{D}_i$ and $\mathcal{D}_j$, and n_k and n_l are egress and ingress gateways in $\mathcal{D}_i$ and $\mathcal{D}_j$, respectively. Consequently, NM is able to transform path $n_{p_1} \overset{*}{\to} n_{p_m}$ that traverses $\mathcal{D}_{l_1}$, ..., $\mathcal{D}_{l_k}$ into a sequence of intra-domain path segments and links $n_{p_1} \overset{*}{\to} E^{l_1}_{l_1,l_k}$, $E^{l_1}_{l_1,l_k} \to I^{l_2}_{l_1,l_k}$, $I^{l_2}_{l_1,l_k} \overset{*}{\to} E^{l_2}_{l_1,l_k}, \ldots, E^{l_{k-1}}_{l_1,l_k} \to I^{l_k}_{l_1,l_k}$, $I^{l_k}_{l_1,l_k} \overset{*}{\to} n_{p_m}$ (Fig. 2). DM$_i$ is able to obtain a route for each path $n_k \overset{*}{\to} n_l$ such that $n_k, n_l \in \mathcal{D}_i$, but it cannot obtain the topology and routing information for any parts of the network located outside of $\mathcal{D}_i$.

4 Distributed Fault Propagation Model

In the multi-domain solution, the FPM of the entire network is distributed among DMs. Each manager maintains a part of the distributed FPM that represents the manager's knowledge of the system structure, i.e., it includes only faults of links that are located in its domain. Faults located in other domains that may propagate to the manager's domain are represented by proxy nodes, called $\mathcal{P}$-nodes.

4.1 Fault Propagation Model of the NM

Let us consider path $n_{p_1} \overset{*}{\to} n_{p_m}$ that traverses domains $\mathcal{D}_{l_1}$, ..., $\mathcal{D}_{l_k}$. Recall that NM transforms this path into a sequence of intra-domain path segments and links (Fig. 2). When only one path exists between $\mathcal{D}_{l_1}$ and $\mathcal{D}_{l_k}$ then all paths that begin in $\mathcal{D}_{l_1}$ and end in $\mathcal{D}_{l_k}$ are transformed into the same sequence of intra-domain path segments and links with the exception of the first and last segments. In this case, the FPM includes a single symptom node labeled $s : l_1 \overset{*}{\to} l_k$ that represents all paths that begin in $\mathcal{D}_{l_1}$ and end in $\mathcal{D}_{l_k}$.

In the FPM of NM, two types of fault nodes exist: (1) ordinary fault nodes, like ones in the centralized case, which represent failures of inter-domain links; these faults are directly isolated by NM, and (2) proxy fault nodes that represent path-segment failures, which cannot be isolated by NM alone because they are located in domains that are not directly managed by NM. For every $\mathcal{D}_i$, one or more $\mathcal{P}$-nodes are created as follows.

1. For every ingress gateway in $\mathcal{D}_i$, $I^i_{l,i}$, we create $\mathcal{P} : I^i_{l,i} \overset{*}{\to} *$ that represents all intra-$\mathcal{D}_i$ paths that begin in $I^i_{l,i}$.
2. For every egress gateway in $\mathcal{D}_i$, $E^i_{i,k}$, we create $\mathcal{P} : * \overset{*}{\to} E^i_{i,k}$ that represents all intra-$\mathcal{D}_i$ paths that end in $E^i_{i,k}$.
3. For each pair of gateways $I^i_{l,k}$ and $E^i_{l,k}$, we create $\mathcal{P} : I^i_{l,k} \overset{*}{\to} E^i_{l,k}$ that represents intra-$\mathcal{D}_i$ path $I^i_{l,k} \overset{*}{\to} E^i_{l,k}$.

In the FPM of NM, $s : l_1 \overset{*}{\to} l_k$ is connected to $\mathcal{P} : * \overset{*}{\to} E^{l_1}_{l_1,l_k}$, $f : E^{l_1}_{l_1,l_k} \to I^{l_2}_{l_1,l_k}$, $\mathcal{P} : I^{l_2}_{l_1,l_k} \overset{*}{\to} E^{l_2}_{l_1,l_k}$, ..., $f : E^{l_{k-1}}_{l_1,l_k} \to I^{l_k}_{l_1,l_k}$, $\mathcal{P} : I^{l_k}_{l_1,l_k} \overset{*}{\to} *$. The FPM of NM contains multiple such symptom nodes for all pairs of domains in $\mathcal{N}$.

The approach chosen in this paper assigns all conditional probabilities between $\mathcal{P}$-nodes and symptom nodes to 1. Prior failure probabilities associated with $\mathcal{P}$-nodes must be calculated by the multi-domain technique based on the state of the fault localization process in domains they represent.

4.2 Fault Propagation Model of DM

As it was stated at the beginning of this section, the FPM built by DM$_i$ includes all intra-$\mathcal{D}_i$ paths and links, i.e., all the information contained in the centralized model of $\mathcal{D}_i$. Such model is sufficient for the diagnosis of symptoms observed in $\mathcal{D}_i$ but is not sufficient for the diagnosis of symptoms DM$_i$ receives from NM. In particular, NM may delegate to DM$_i$ a part of a task involved in the diagnosis of path $n_{p_1} \overset{*}{\to} n_{p_m}$ that traverses $\mathcal{D}_i$. In this case, DM$_i$ will be notified about a failure of its intra-domain path that constitutes the intra-$\mathcal{D}_i$ segment of $n_{p_1} \overset{*}{\to} n_{p_m}$. This notification only indicates a possibility of the segment's failure, since the failure of $n_{p_1} \overset{*}{\to} n_{p_m}$ could have been caused by its path-segment or link that is not located in domain $\mathcal{D}_i$. Thus, symptoms received by DM from NM are typically associated with a high degree of uncertainty, i.e., they are likely to be spurious. To deal with spurious symptoms, we use ideas from previous work [1] as follows.

Let $s : n_r \overset{*}{\to} n_t$ be an intra-$\mathcal{D}_i$ symptom received by DM$_i$ from NM in the process of diagnosing a failure of inter-domain path $n_{p_1} \overset{*}{\to} n_{p_m}$. To model the possibility that

$s : n_r \xrightarrow{*} n_t$ is spurious in the FPM of DM$_i$, we create a $\tilde{\mathcal{P}}$-node that represents all possible causes of $s : n_r \xrightarrow{*} n_t$ that are not located in $\mathcal{D}_i$. Observe that, since $n_r \xrightarrow{*} n_t$ constitutes a segment of an inter-domain path, at least one of n_r, n_t is a gateway in $\mathcal{D}_i$. Let l and k be identifiers of domains that contain n_{p_1} and n_{p_m}, respectively. Let us consider three cases.

1. $i = l$; We create $\tilde{\mathcal{P}} : * \xrightarrow{*} E^i_{i,k}$ and connect it to $s : n_r \xrightarrow{*} n_t$.
2. $i = k$; We create $\tilde{\mathcal{P}} : I^i_{l,i} \xrightarrow{*} *$ and connect it to $s : n_r \xrightarrow{*} n_t$.
3. $i \neq l$ and $i \neq k$; We create $\tilde{\mathcal{P}} : I^i_{l,k} \xrightarrow{*} E^i_{l,k}$ and connect it to $s : n_r \xrightarrow{*} n_t$.

Similar to the FPM of NM, conditional probabilities on edges between $\tilde{\mathcal{P}}$-nodes and symptom nodes in the FPM of DM$_i$ are set to 1, while prior failure probabilities assigned to $\tilde{\mathcal{P}}$-nodes in the FPM of DM$_i$ are calculated by NM and sent to DM$_i$ together with reported symptoms.

5 Multi-domain Fault Localization Algorithm

In this section, we present an outline of a multi-domain fault localization algorithm (Alg. 3) based on the distributed FPM described in Section 4, which may be refined to create multi-domain versions of Algs. 1 and 2. In the pseudo-code of Alg. 3, which is presented on page 1043, sections of the algorithm that are specialized for different probabilistic reasoning mechanisms are underlined. The multi-domain fault localization algorithm proceeds in two phases performed by every DM and NM: (1) symptom analysis and (2) fault selection. Initially, the model is reset by assigning prior failure probabilities to proxy nodes. In our implementation, these probabilities are set to 0 in the FPM of NM. In the FPM of DM, no $\tilde{\mathcal{P}}$-nodes exist at the beginning, and therefore no assignment is needed. Symptom-analysis and fault-selection phases progress by traversing the hierarchy of managers in a bottom-up or top-down manner.

5.1 Symptom Analysis Phase

The symptom analysis phase is executed for every received alarm that indicates a failure of an end-to-end path. This alarm can be received either by the NM or a DM. A DM can start symptom analysis only if the entire failed path belongs to its domain. If the DM is not able to diagnose the symptom it forwards it to the NM, which initiates the symptom diagnosis (function *analyze_internal*).

Symptom Processing by NM: In the process of diagnosing $s : n_{p_1} \xrightarrow{*} n_{p_m}$ (see function *analyze_internal* in the pseudo-code of Alg. 3), the NM first maps it into node $s : l_1 \xrightarrow{*} l_k$ in its FPM, such that $n_{p_1} \in \mathcal{D}_{l_1}$ and $n_{p_m} \in \mathcal{D}_{l_k}$. Then, it splits $n_{p_1} \xrightarrow{*} n_{p_m}$ into path segments and links. Failures of path segments are then interpreted as symptoms $s_1, s_2, \ldots, s_k$ that will be reported to DM$_{l_1}$, DM$_{l_2}$, ..., DM$_{l_k}$, respectively. Note that in the FPM of DM$_{l_j}$, all causes of s_j that are not located in $\mathcal{D}_{l_j}$ are represented by a $\tilde{\mathcal{P}}$-node that is attached to node s_j. To indicate that s_j may be spurious in $\mathcal{D}_{l_j}$, NM calculates the prior probability associated with this $\tilde{\mathcal{P}}$-node in the FPM of DM$_{l_j}$. Suppose that $s_j = s : n_r \overset{*}{\to} n_t$. Then the probability that s_j is spurious is obtained as follows:

$$p_{\text{spurious}}(s : n_t \overset{*}{\to} n_r) = \prod_{\mathcal{P} \in \mathcal{P}(n_t \overset{*}{\to} n_r)} \frac{Prob\{\neg\mathcal{P}\}}{} \tag{1}$$

$$\mathcal{P}(n_t \overset{*}{\to} n_r) = \begin{cases} \{\mathcal{P} : n_r \overset{*}{\to} n_t,\ \mathcal{P} : * \overset{*}{\to} n_t, \mathcal{P} : n_r \overset{*}{\to} *\} & \\ & \text{if } n_r \text{ and } n_t \text{ are ingress and egress gateways} \\ \{\mathcal{P} : * \overset{*}{\to} n_t\} & \text{if } n_r \text{ is an ingress gateway} \\ \{\mathcal{P} : n_r \overset{*}{\to} *\} & \text{if } n_t \text{ is an egress gateway} \end{cases}$$

After calculating $p_{\text{spurious}}(s_j)$, NM delegates the diagnosis of s_j to DM$_{l_j}$, for $j = 1 \ldots k$ by invoking *analyze_external*. As a result of the diagnosis performed by DM$_j$, the NM obtains $p(\mathcal{P}_j)$, where $\mathcal{P}_j$ is the $\mathcal{P}$-node representing $\mathcal{D}_i$ that is connected to $s : l_1 \overset{*}{\to} l_k$ in the FPM of NM. Then NM updates its FPM. Finally, NM analyzes $s : l_1 \overset{*}{\to} l_k$ using the symptom-analysis procedure of either Alg. 1 or 2 (function *inference*).

To limit duplicate delegations of the same symptom to DM$_{l_j}$, NM marks nodes as either UNOBSERVED or OBSERVED_INTERNAL. While analyzing $s : n_{p_1} \overset{*}{\to} n_{p_m}$, when $s : l_1 \overset{*}{\to} l_k$ is marked OBSERVED_INTERNAL, the NM does not delegate symptoms to DM$_{l_j}$s for $j = 2 \ldots k-1$. It does, however, delegate the analysis to DM$_{l_1}$ and DM$_{l_k}$, since paths represented by $s : l_1 \overset{*}{\to} l_k$ differ in their segments located in $\mathcal{D}_{l_1}$ and $\mathcal{D}_{l_k}$.

Symptom processing by DM: DM$_i$ may start the processing of $s_r = s : n_{p_1} \overset{*}{\to} n_{p_m}$ when (1) it observes a failure of intra-$\mathcal{D}_i$ path $n_{p_1} \overset{*}{\to} n_{p_m}$ or (2) s_r is delegated to DM$_i$ by NM. In the former case, s_r is an internal symptom; in the latter case it is called an external symptom. To distinguish between different observations of the same symptom, DM$_i$ marks symptom nodes as either UNOBSERVED, OBSERVED_INTERNAL, and OBSERVED_EXTERNAL when they are not processed, processed as a result of internal observation, and processed as a result of a delegation by NM, respectively.

Internal symptoms are processed by function *analyze_internal*. First, the association between the observed symptom and its $\tilde{\mathcal{P}}$-node (if one exists) is removed, as the symptom can no longer be explained by external causes. Then, a symptom-analysis procedure is executed.

The processing of external symptoms is done by function *analyze_external*. Assume that $s_r = s : n_{p_1} \overset{*}{\to} n_{p_m}$ has been delegated to DM$_i$ as a result of a failure of a path between domains $\mathcal{D}_l$ and $\mathcal{D}_k$. DM$_i$ also receives two parameters from NM: $\mathcal{P}^i_{l,k}$ and p_{spurious}, where $\mathcal{P}^i_{l,k}$ is a description of a $\mathcal{P}$-node that is connected to node $s : l \overset{*}{\to} k$ in the FPM of NM, and p_{spurious} is the probability that s_r is spurious. DM$_i$ first updates its FPM by assigning p_{spurious} as the prior probability to the $\tilde{\mathcal{P}}$-node connected to symptom s_r. If the symptom has been previously analyzed, DM$_i$ returns the stored value of $p(\mathcal{P}^i_{l,k})$. Otherwise, it updates the FPM by connecting s_r to its corresponding $\tilde{\mathcal{P}}$-node, and updates the state of fault localization to reflect the modified value of its prior failure probability. Then, a probabilistic reasoning mechanism is used to analyze the symptom. Finally, $p(\mathcal{P}^i_{l,k})$ is calculated as follows:

$$p(\mathcal{P}^i_{l,k}) = \begin{cases} 0 & \mathcal{S}^i_{l,k} = \emptyset \\ \prod_{s_i \in \mathcal{S}^i_{l,k}} bel(s_i) & \text{otherwise} \end{cases} \tag{2}$$

$$\mathcal{S}^i_{l,k} = \{s_k = s : n_r \overset{*}{\to} n_t | n_r \overset{*}{\to} n_t \in \mathcal{P}^i_{l,k} \text{ and } s_k \text{is not } \texttt{UNOBSERVED}\} \tag{3}$$

$$bel(s_i) = \begin{cases} 1 & \text{if } s_i \text{ is } \texttt{OBSERVED\_INTERNAL} \\ 1 - \prod_{f_j \in \mathcal{F}} (1 - p(s_i|f_j)\underline{Prob\{f_j\}}) & \text{otherwise} \end{cases} \tag{4}$$

5.2 Fault Selection Phase

In the fault selection phase, DMs and NM have to synchronize their FPMs by updating prior failure probabilities associated with their proxy nodes. Afterward, DMs and NM choose the most likely hypotheses.

Algorithm 3: Multi-domain algorithm

```
Symptom analysis phase:
DM: FOR every observed symptom $s : n_{p_1} \overset{*}{\to} n_{p_m}$ DO
        IF internal symptom analyze_internal($s : n_{p_1} \overset{*}{\to} n_{p_m}$)
        ELSE NM$\to$analyze_internal($s : n_{p_1} \overset{*}{\to} n_{p_m}$)
NM: FOR every observed symptom $s : n_{p_1} \overset{*}{\to} n_{p_m}$ DO analyze_internal($s : n_{p_1} \overset{*}{\to} n_{p_m}$)
DM$_i$: FUNCTION analyze_internal($s_r$)
      IF $s_r$ is not marked OBSERVED_INTERNAL THEN update the model AND run $\underline{\text{inference}(s_r)}$
NM: FUNCTION analyze_internal($s : n_{p_1} \overset{*}{\to} n_{p_m}$)
      map $s : n_{p_1} \overset{*}{\to} n_{p_m}$ to $s : l_1 \overset{*}{\to} l_k$ such that $n_{p_1} \overset{*}{\to} n_{p_m} \in l_1 \overset{*}{\to} l_k$
      transform $n_{p_1} \overset{*}{\to} n_{p_m}$ into $n_{p_1} \overset{*}{\to} E^{l_1}_{l_1,l_k}, E^{l_1}_{l_1,l_k} \to I^{l_2}_{l_1,l_k}, I^{l_2}_{l_1,l_k} \overset{*}{\to} E^{l_2}_{l_1,l_k}, \ldots, I^{l_k}_{l_1,l_k} \overset{*}{\to} n_{p_m}$
      set $s_1 = s : n_{p_1} \overset{*}{\to} E^{l_1}_{l_1,l_k}$, $s_2 = s : I^{l_2}_{l_1,l_k} \overset{*}{\to} E^{l_2}_{l_1,l_k}, \ldots, s_k = s : I^{l_k}_{l_1,l_k} \overset{*}{\to} n_{p_m}$
      find $\mathcal{P}$-nodes connected to $s : l_1 \overset{*}{\to} l_k$: $\mathcal{P}_1 = \mathcal{P} : * \overset{*}{\to} E^{l_1}_{l_1,l_k}, \ldots, \mathcal{P}_k = \mathcal{P} : I^{l_k}_{l_1,l_k} \overset{*}{\to} *$
      FOR $1 \leq j \leq k$ DO
          IF $s : l_1 \overset{*}{\to} l_k$ is marked UNOBSERVED OR $j = 1$ OR $j = k$ THEN
              $p(\mathcal{P}_j) = \text{DM}_{l_j}$ $\to$analyze_external($s_j, \mathcal{P}_j, p_{\text{spurious}}(s_j)$)
      IF $s : l_1 \overset{*}{\to} l_k$ is not marked OBSERVED_INTERNAL THEN
          update the model AND run $\underline{\text{inference}(s : l_1 \overset{*}{\to} l_k)}$
DM$_i$: FUNCTION analyze_external($s_r, \mathcal{P}^i_{l,k}, p_{\text{spurious}}$)
      IF $s_r$ is not marked UNOBSERVED THEN return $p(\mathcal{P}^i_{l,k})$
      ELSE update the model, run $\underline{\text{inference}(s_r)}$, AND RETURN $p(\mathcal{P}^i_{l,k})$
Fault selection phase:
NM: FOR every $\mathcal{P}^i_{l,k}$ DO obtain $p(\mathcal{P}^i_{l,k})$ from DM$_i$ AND update the model
      FOR every $\mathcal{P}^i_{l,k}$ DO send $\underline{Prob\{\neg\mathcal{P}^i_{l,k}\}}$ to DM$_i$
      FOR every DM$_i$ DO obtain the most likely set of faults from DM$_i$
      obtain the most likely set of faults in NM
```

It is not difficult to refine Alg. 3 to create multi-domain versions of Algs. 1 and 2. The resultant multi-domain versions of Algorithms 1 and 2 are labeled **Alg. 3A** and **Alg. 3B**, respectively [9]. The computational complexities of fault localization performed by a single manager using Algs. 3A and 3B in a network domain composed of n nodes or domains are $\mathcal{O}(n^5)$ and $\mathcal{O}(n^4)$, respectively.

In the entire algorithm, the messaging overhead is $\mathcal{O}(\max(|\mathcal{S}_O|, n^3))$ per domain, where n is a number of nodes or sub-domains in the domain.

6 Simulation Study

We evaluate the performance of Algs. 3A and 3B through simulation. The study uses sets of fault localization scenarios in which faults and symptoms are randomly generated based on the conditional probability distribution that describes non-deterministic causal relationships between faults and symptoms.

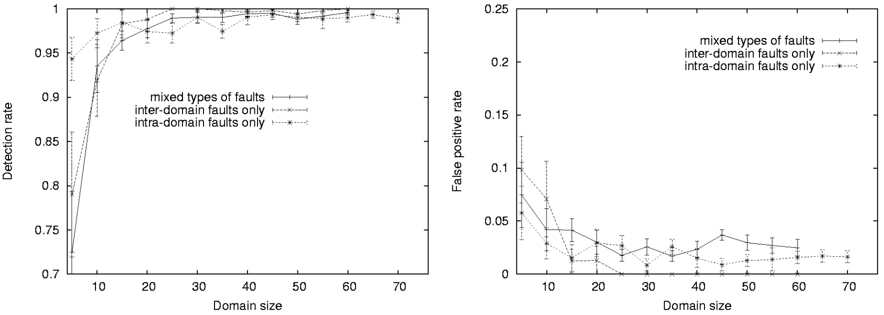
The simulation study uses network topologies similar to those of the Internet. The generation of random graphs resembling the topology of real-life networks has been a widely studied research area [10,11,12,13]. This study uses a generator based on Barabasi-Albert power-law model [11], because its implementation (BRITE [14]) is available in public domain, and because topologies built based on this model are representative of the Internet topology [15].

Using the topology generator we create a random network composed of $N = 10$ domains and n nodes in each domain, where n varies between 5 and 70. We determine routes between any source and destination using the shortest-path policy for intra-domain routes. We choose inter-domain routes such that the number of visited domains is minimized. Then, we generate prior failure probabilities for inter-domain and intra-domain links, which are uniformly distributed over the range $[0.0001, 0.001]$. For each intra-domain link l and path p, we randomly choose the probability that p fails if l fails from set $\{0.25, 0.5, 0.75\}$. In the FPM of the NM, the conditional probabilities are all equal to 1. We randomly generate a subset of symptoms observable in every domain to include 50% of all intra-domain paths. The observability ratio [1,2] for inter-domain paths is 2%.

We distinguish three types of experiments: those involving only intra-domain link failures,inter-domain link failures, and both types of failures. In every study, two performance metrics are calculated: detection rate, DR, defined as a percentage of faults occurring in the network which are isolated by the technique, and false positive rate, FPR, defined as a percentage of faults reported by the technique that are not occurring in the network [1].

In Figs. 3a-3(b), we show the accuracy of Alg. 3A in a ten-domain network, in which each domain is composed of up to 70 nodes. Thus the entire network consists of up to 700 nodes. Figs. 4(a)-4(b) present the results of the same experiment executed using Alg. 3B.

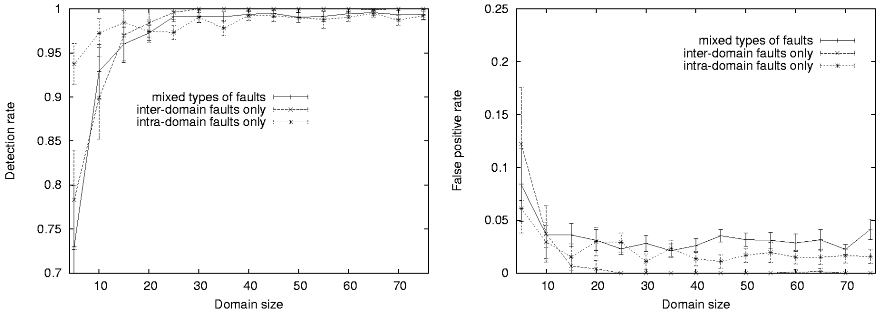
The figures compare the accuracy achievable in scenarios involving only inter-domain, only intra-domain, and both types of faults. Clearly, the mixed-failure scenarios are the most difficult to diagnose since they always involve at least two concurrent faults located in different network domains. The interpretation of the faults' symptoms, which may overlap, leads to ambiguity. This results in a lower fault-localization accuracy of



(a) Detection rate

(b) False positive rate

Fig. 3. Accuracy of Algorithm 3A in a ten-domain network.



(a) Detection rate

(b) False positive rate

Fig. 4. Accuracy of Algorithm 3B in a ten-domain network.

mixed-fault scenarios compared to that of other types of scenarios, which is conspicuous in networks of small size. Scenarios involving only inter-domain symptoms are the easiest to solve as the number of suspect faults is usually small compared to the amount of available evidence. In intra-domain- and mixed-fault scenarios, increasing the domain size also increases the frequency of multi-fault scenarios. For example, in mixed scenarios the number of simultaneous faults exceeds 3 in a 700-node network. In inter-domain scenarios, the number of faults is equal to 1.02 on average, and does not change when the network size increases, while the number of received symptoms grows with the increasing network size.

We repeated the same set of experiments using networks composed of 50 domains. The results, which are not shown in this paper, confirm the results obtained using ten-domain networks (see [1] for details).

7 Conclusion

The paper identifies two main difficulties of fault management in multi-domain networks: failure propagation among domains and a lack of global information about the system structure and state. To address these challenges, the paper proposes a multi-domain algorithm, which is shown to provide high accuracy while increasing the admissible network size by an order of magnitude. Clearly, the biggest challenge in applying the fault localization technique proposed in this paper to real-life problems is obtaining the probabilistic FPM. To build an FPM for end-to-end service failure diagnosis a knowledge of network logical topology and communication protocols is needed. The problem of building FPMs is beyond the scope of this paper. Our previous publications in this subject survey some of the techniques that can be used to build an FPM for end-to-end service failure diagnosis [9]. [2]

References

1. M. Steinder and A. S. Sethi, "Non-deterministic fault localization in communication systems using belief networks," *IEEE/ACM Transactions on Networking*, 2004, (to appear).
2. M. Steinder and A. S. Sethi, "Non-deterministic event-driven fault diagnosis through incremental hypothesis updating," in *Integrated Network Management VIII*, G. Goldszmidt and J. Schoenwaelder, Eds., Colorado Springs, CO, Mar. 2003.
3. G. Jakobson and M. D. Weissman, "Alarm correlation," *IEEE Network*, vol. 7, no. 6, pp. 52–59, Nov. 1993.
4. I. Katzela and M. Schwartz, "Schemes for fault identification in communication networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 6, pp. 733–764, 1995.
5. S. A. Yemini, S. Kliger, E. Mozes, Y. Yemini, and D. Ohsie, "High speed and robust event correlation," *IEEE Communications Magazine*, vol. 34, no. 5, pp. 82–90, 1996.
6. J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers, 1988.
7. A. T. Bouloutas, S. B. Calo, A. Finkel, and I. Katzela, "Distributed fault identification in telecommunication networks," *J. of Network and Systems Management*, vol. 3, no. 3, 1995.
8. P. Hasselmeyer, "An infrastructure for the management of dynamic service networks," *IEEE Communications Magazine*, vol. 41, no. 4, pp. 120–126, 2003.
9. M. Steinder and A. S. Sethi, "Multi-domain diagnosis of end-to-end service failures in hierarchically routed networks," Tech. Rep. 2003-10, CIS, University of Delaware, 2003.
10. W. Aiello, F. Chung, and L. Lu, "A random graph model for massive graphs," in *ACM Symposium on Theory of Computing*, Portland, OR, May 2000, pp. 171–180.
11. A. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, pp. 509–512, Oct. 1999.
12. K. Calvert, M. Doar, and E. Zegura, "Modeling Internet topology," *IEEE Transactions on Communications*, pp. 160–163, Dec. 1997.
13. C. Jin, Q. Chen, and S. Jamin, "Inet: Internet topology generator," Tech. Rep. CSE-TR443-00, EECS, University of Michigan, 2000.
14. A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE:universal topology generation from a user's perspective," Tech. Rep. BUCS-TR-2001-003, CS Dept, Boston Univ., 2001.
15. T. Bu and D. Towsley, "On distringuishing between Internet power law topology generators," in *Proc. of IEEE INFOCOM*, New York, NY, Jun. 2002.

[2] The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied of the Army Research Lab or the U.S. Government.