1.1 Related Work

The importance of failure monitoring in optical communication networks has generated a wide body of approaches to solve it. They differ in the information they require from the monitoring tools (timestamps, failure probabilities), the way they collect this information (via passive or active monitoring), the methodology they use for making a location (expert system or case-based rules, abstract network model or black box learning), etc (see [9] for a complete review). In this paper, We use the model based approach of [3] where a transparent optical network is modelled as a directed deterministic dependency graph. The dependency relations are dictated by the established light paths in the network. There is also much research dealing with monitoring the performance of transparent optical networks. Recent progress in optical performance monitoring can be found in [8]. Other researchers, e.g. [10], have proposed to use active probes as tools to monitor and diagnose failures in transparent optical networks. A detailed survey of the existing commercial optical equipment and monitoring devices in transparent optical networks is given in [3].

Accounting for the effects of noisy alarms is a significant challenge in network failure location. Although many researchers [3, 11, 12] have suggested that failure location algorithms must be able to cope with erroneous alarms, most location algorithms today have avoided the issue of alarm uncertainty because of the complexity of covering all possible failures and all possible alarm uncertainties. A few works [3, 11, 12] in the literature incorporate missing and false alarms into fault location. In [11], Yemini et al. only consider single failures. In [12], Steinder et al. use failure probabilities of network components and the probabilities of corrupted alarms to develop a belief network to solve the failure location problem. In [3], Mas et al. develop a fault location algorithm capable of handling multiple failures and erroneous alarms in opaque optical networks without using probabilities. However, the approach in [3] requires considering all different combinations of failures and corrupted alarms in the network and therefore requires an exponential location time (or an exponential memory space).

1.2 Our Contributions

In this paper, we first mathematically formulate the failure location problem as a maximum likelihood inference problem in Section 2. We then show in Section 3 that the failure location problem is equivalent to the famous NP-hard set cover problem [13]. The benefits of proving the equivalence of the two problems are two-fold. First, the results enable us to adapt existing set cover algorithms in the literature to solve the failure location problem in optical networks. Second, linking the failure location problem to the set cover problem allows us to discover that false and missing alarms do not have equal effects on the failure location. We show in Section 4 that false alarms can be corrected in polynomial time, but that correcting missing alarms is NP-hard. The rather surprising advantage of false alarms over missing alarms, in terms of time-complexity, carries directly to the accuracy of the failure location: a polynomial time failure location algorithm is more accurate when most of the corrupted alarms are false than when most of the corrupted alarms are missing. Our studies on real network topologies in Section 5 confirm the advantages of tilting thresholds towards low values, so that corrupted alarms are rather false than missing. We conclude the paper in Section 6.

$p_m)^{|\mathcal{M}_R|-|\mathcal{M}_F(\mathcal{C}')|}$. In the formula for $Prob(\mathcal{M}_R, \mathcal{M}_S|\mathcal{C}')$, the first term $p_f^{|\mathcal{M}_F(\mathcal{C}')|}$ comes from false alarms, the second term $(1-p_f)^{|\mathcal{M}_S|-|\mathcal{M}_M(\mathcal{C}')|}$ from the silent monitors that are not missing alarms, the third term $p_m^{|\mathcal{M}_M(\mathcal{C}')|}$ from missing alarms and the last term $(1-p_m)^{|\mathcal{M}_R|-|\mathcal{M}_F(\mathcal{C}')|}$ from ringing monitors that are not false alarms. We have made the implicit assumptions that given the set of faulty optical components $\mathcal{C}'$ the alarm status of monitoring devices are conditionally independent.

The probabilities p, p_m and p_f are difficult to obtain and are generally not available in practice. Therefore, the maximization problem in (1) cannot be solved analytically as is and requires some approximations. We propose a heuristic to solve the maximization problem in (1) by first minimizing the number of corrupted alarms $|\mathcal{M}_M(\mathcal{C}')| + |\mathcal{M}_F(\mathcal{C}')|$ and then minimizing the number of failures $|\mathcal{C}'|$. In other words, the failure location problem can be separated into two steps. The first step is to identify and correct corrupted alarms, which we call the Error Correction (EC) problem. The second step is to identify the failures, which we call the Multiple Fault (MFAULT) location problem, without corrupted alarms. For a logical development of the paper, we first present the MFAULT problem in Section 3 and then present the EC problem in Section 4.

3 The Failure Location Problem With Perfect Alarms

In this section, we study the failure location problem in the ideal scenario where there are no corrupted alarms. The objective of the MFAULT problem is to cover all ringing alarms but not silent alarms with the smallest number of failure candidates. The MFAULT problem can be formulated as follows. We want to find a subset $\mathcal{C}'$ of $\mathcal{C}$ such that the number of failures $|\mathcal{C}'|$ is minimized and there are no false alarms, which we can write as

$$\mathcal{M}_F(\mathcal{C}') = \emptyset, \tag{2}$$

and no missing alarm, which is to say that

$$\mathcal{M}_M(\mathcal{C}') = \emptyset. \tag{3}$$

The MFAULT is equivalent to the famous set cover problem (SC) [15]. The SC problem is a known NP-complete problem and can be defined as follows [13]: Given a finite set $\mathcal{S}$, a collection $\mathcal{X}$ of subsets of $\mathcal{S}$, we need to determine a subset $\mathcal{X}' \subseteq \mathcal{X}$ such that $|\mathcal{X}'|$ is minimized and $\bigcup_{X \in \mathcal{X}'} X = \mathcal{S}$.

Theorem 1. *Any instance of the MFAULT problem can be mapped into an instance of the SC problem and vice versa.*

The MFAULT problem therefore is NP-complete. In optical networks, failure location needs to be fast so that repair actions can be done swiftly. For this reason, we choose to adapt the greedy approximation algorithm for the set cover problem [14] to solve the MFAULT problem. This algorithm is fast and yet is the best polynomial time approximation algorithm for the SC problem (and hence the MFAULT problem) in terms of the worst case performance. The MFAULT approximation algorithm presented below is a greedy algorithm that chooses at each iteration the failure candidate whose domain contains the largest number of ringing alarms.

4.2 When There Are Only Missing Alarms

We now consider the dual case where there are only missing alarms. This scenario happens when threshold values in monitoring devices are set high. In this case, we need to determine a set of network components $\mathcal{F}_D \subseteq \mathcal{C}$ whose failures would produce the smallest set of missing alarms, i.e., minimizes $|\mathcal{M}_M(\mathcal{F}_\mathcal{D})|$ and satisfies Condition (2), i.e., $\mathcal{M}_F(\mathcal{F}_\mathcal{D}) = \emptyset$. Condition (2) is to guarantee that there are no false alarms.

We call this problem the Missing Alarm ($MALARM$) problem. The MALARM problem can be shown to be NP-complete by a reduction from the the red-blue set cover problem [16]. The red-blue set cover problem is defined as follows. Given a finite set $\mathcal{S}$, which contains two disjoint subsets $\mathcal{R}$ and $\mathcal{B}$ ($\mathcal{R} \cup \mathcal{B} = \mathcal{S}$ and $\mathcal{R} \cap \mathcal{B} = \emptyset$) and a collection $\mathcal{X}$ of subsets of $\mathcal{S}$, we need to determine a subset $\mathcal{X}' \subseteq \mathcal{X}$, such that every element of $\mathcal{B}$ belongs to at least one member of $\mathcal{X}'$ and the number of elements of $\mathcal{R}$ that are covered by members of $\mathcal{X}'$ is minimized. The proof is given in [15].

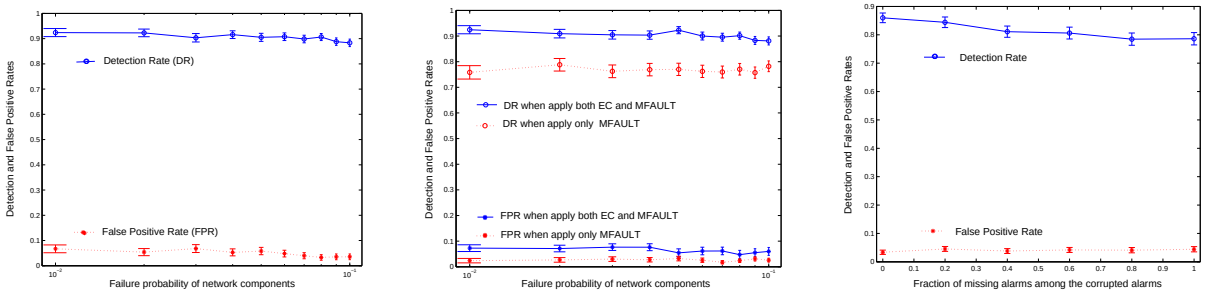
4.3 Approximation algorithm for the error correction problem

In real networks, one can never be sure that the corrupted alarms are only false or only missing. It is therefore important to have an algorithm for the error correction problem that works when there are both false and missing alarms. Finding an algorithm for the EC problem is not easy for the following reasons. The red-blue set cover problem is not only NP-hard, it is also much harder to approximate than the set cover problem [16]. To date, there is no known approximation algorithm with a finite bound for the red-blue set cover problem in the literature. The Error Correction (EC) algorithm below greedily tries to cover as many ringing alarms and as few silent alarms as possible at each iteration by picking a component c whose domain has the smallest number of silent monitors, i.e., c minimizes $|\mathcal{M}_S \cap Domain(c)|$. Although we cannot provide a bound on the performance of the EC algorithm, our experiments in Section 5 show that the EC algorithm performs quite well on real network topologies.

The EC algorithm

- *Step 1*:
  1. Initialize $\mathcal{F}_D$ to an empty set: $\mathcal{F}_D = \emptyset$.
  2. Calculate $\mathcal{M}_S \cap Domain(c_i)$ for each component $c_i \in \mathcal{C}$.
  3. Find c that minimizes $|\mathcal{M}_S \cap Domain(c)|$.
- *Step 2*: While $|\mathcal{M}_R \cap Domain(c)| - |\mathcal{M}_S \cap Domain(c)| \geq 0$
  1. Add c to $\mathcal{F}_D$: $\mathcal{F}_D = \mathcal{F}_D \cup \{c\}$.
  2. Update the sets: $\mathcal{M}_R = \mathcal{M}_R \backslash Domain(c)$, $\mathcal{M}_S = \mathcal{M}_S \backslash Domain(c)$ and $Domain(c_i) = Domain(c_i) \setminus Domain(c)$ for all $c_i \in \mathcal{C}$.
  3. Recalculate $\mathcal{M}_R \cap Domain(c_i)$ and $\mathcal{M}_S \cap Domain(c_i)$ for each components $c_i \in \mathcal{C}$.
  4. Find c that minimizes $|\mathcal{M}_S \cap Domain(c)|$.
- *Step 3*: Output $\mathcal{M}_F(\mathcal{F}_\mathcal{D})$ and $\mathcal{M}_M(\mathcal{F}_\mathcal{D})$.

The EC algorithm has the following notable features. When all corrupted alarms are false alarms, we have $\mathcal{M}_S \cap Domain(\{c_i\}) = \emptyset$ and hence $|\mathcal{M}_R \cap Domain(c_i)| \geq |\mathcal{M}_S \cap Domain(c_i)|$ for components c_i that are faulty. Therefore, the EC algorithm is



(a) when $MR = FR = 0$. (b) When $MR = FR = 0.01$. (c) When the nature of the alarms chages.

Fig. 2. Performance of the failure location algorithm on the NSFNET topology with 20 channels when there are no corrupted alarms and when there are corrupted alarms.

and FPR do not change much with the failure probability of network components. This observation suggests that our failure location algorithm performs well for small and large failure probabilities of network components.

Figure 2(b) plots the performance of the failure location algorithm when $MR, FR > 0$. To evaluate the effectiveness of the EC algorithm, we also plot the results when applying only the MFAULT algorithm in the non-ideal scenarios. We observe that when there are corrupted alarms, the EC algorithm helps improve the accuracy of the overall failure location by a large margin. The results demonstrate that even though the number of corrupted alarms is small, their effect on the failure location algorithm is substantial and cannot be neglected. Figure 3(b) also shows that in the non-ideal scenario, the False Positive Rate is higher than in the ideal scenario. This observation is explained by the fact that the EC algorithm is an approximation algorithm and its solution is not optimal.

5.3 Effect of False and Missing Alarms

We now evaluate the impact of false and missing alarms on the failure location algorithm. To compare the effect of false and missing alarms, we fix the total number of corrupted alarms, i.e., $MR + FR$ is fixed, but not the nature, i.e., MR and FR change. We run simulations on the NSFNET topology with 20 channels. The component failure probability is fixed at 0.01. The average number of monitors in this setting is 40 monitors. The average number of corrupted alarms is kept at 0.8 alarms (2% of the number of monitors). We vary the fraction of missing alarms among the corrupted alarms from 0 to 1, where 0 means there is no missing alarms and 1 means all corrupted alarms are missing alarms. The results are plotted in Figure 2(c).

We observe that as the fraction of missing alarms increases, the DR of the failure location algorithm decreases rapidly whereas the FPR does not change much. These results follow from our theoretical studies in Section 4 that when the majority of corrupted alarms are false alarms, these alarms can be corrected exactly by the EC algorithm; whereas when the majority of corrupted alarms are missing these alarms are only corrected approximately by the EC algorithm. We would like to emphasize here that the asymmetry between missing and false alarms is not an artifact of our EC algorithm. This asymmetry comes from the fact that false alarms are easier to correct than missing alarms, which we have solidly proved in Section 4.