Minimum Congestion Redundant Assignments to Tolerate Random Faults*

Dimitris A. Fotakis[1,2] Paul G. Spirakis[1,2]

[1] Computer Engineering and Informatics Department
University of Patras, 265 00 Rion, Patras, Greece

[2] Computer Technology Institute
Riga Feraiou 61, 262 21 Patras, Greece

Email: fotakis@cti.gr, spirakis@cti.gr

Abstract. We consider the problem of computing minimum congestion, fault-tolerant, redundant assignments of messages to faulty, parallel delivery channels. In particular, we are given a set K of faulty channels, each having an integer capacity c_i and failing independently with probability f_i. We are also given a set M of messages to be delivered over K, and a fault-tolerance constraint $(1-\epsilon)$, and we seek a redundant assignment ϕ that minimizes congestion $\mathsf{Cong}(\phi)$, i.e. the maximum channel load, subject to the constraint that, with probability no less than $(1-\epsilon)$, all the messages have a copy on at least one active channel. We present a polynomial-time 4-approximation algorithm for identical capacity channels and arbitrary message sizes, and a $2\left\lceil\frac{\ln(|K|/\epsilon)}{\ln(1/f_{\max})}\right\rceil$-approximation algorithm for related capacity channels and unit size messages.

Both algorithms are based on computing a collection $\{K_1, \ldots, K_\nu\}$ of disjoint channel subsets such that, with probability no less than $(1-\epsilon)$, at least one channel is active in each subset. The objective is to maximize the sum of the minimum subset capacities. Since the exact version of this problem is $\mathcal{NP}$-complete, we provide a 2-approximation algorithm for identical capacities, and a polynomial-time $(8+\mathrm{o}(1))$-approximation algorithm for arbitrary capacities.

Keywords: Polynomial-time approximation algorithms, fault-tolerance, redundant assignments.

* This work was partially supported by ESPRIT LTR Project no. 20244—ALCOM–IT.

1 Introduction

In many practical applications involving design with faulty components (e.g. fault-tolerant network design, fault-tolerant scheduling), a combinatorial structure, such as a graph, should be optimized to best tolerate random and independent faults with respect to a given property, such as connectivity, or non-existence of isolated points (e.g. [1]).

For instance, let us consider some messages to be delivered over a set of faulty, parallel delivery channels. Each channel fails independently with a given probability, and, in case of failure, it delivers no messages to the destination. Then, redundancy can be employed to increase the probability that all the messages have a copy on at least one active channel, i.e. all the messages are delivered to the destination. On the other hand, redundancy increases the channel congestion, that is the maximum channel load. A natural question arising from this setting is whether it is possible to deliver, with probability no less than $(1-\epsilon)$, at least one copy of all the messages to the destination, without the load of any channel exceeding a given bound.

In this work, we consider a set K of faulty, parallel channels, each having an integer capacity c_i and failing independently with probability f_i, for some rational $1 > f_i > 0$. We are also given a set M of messages, each of an integer size s_j, and a rational fault-tolerance constraint $(1-\epsilon)$. A message of size s_j causes a load s_j/c_i when assigned to a channel of capacity c_i. We seek a redundant assignment $\phi : M \mapsto 2^K$ that tolerates the faults (i.e. all the messages have a copy on at least one active channel) with probability no less than $(1-\epsilon)$, and minimizes the channel congestion $\mathsf{Cong}(\phi)$ (i.e. the maximum channel load caused by ϕ). Then, we distinguish the case of identical capacity channels, where all the capacities c_i are equal, and the case of related capacity channels, where the capacities c_i can be arbitrary integer numbers. The problem of determining Minimum Fault-Tolerant Congestion is $\mathcal{NP}$-hard, because it is a generalization of minimizing makespan in a reliable parallel machine environment (e.g. [2]).

A natural strategy for constructing a redundant, $(1-\epsilon)$-fault-tolerant assignment is to compute a collection $\mathcal{K} = \{K_1, \dots, K_\nu\}$ of disjoint channel subsets such that, with probability no less than $(1-\epsilon)$, at least one channel is active in each subset K_j. Then, each subset K_j can be thought as a reliable effective channel of capacity $C(K_j) = \min_{i \in K_j}\{c_i\}$, and any algorithm for non-redundant scheduling on $\mathcal{K}$ can be used for computing ϕ. A reasonable objective for such a collection $\mathcal{K}$ is to maximize the total effective capacity $C(\mathcal{K}) = \sum_{j=1}^{\nu} C(K_j)$.

We observe that the problem of computing an optimal, $(1-\epsilon)$-fault-tolerant collection of disjoint subsets is $\mathcal{NP}$-complete even for identical capacity channels. Then, we present a polynomial-time 2-approximation algorithm for partitioning a set of identical capacity channels. In case of related capacities, we obtain a simple $2\left\lceil \frac{\ln(|K|/\epsilon)}{\ln(1/f_{\max})} \right\rceil$-approximation algorithm, and, for any constant $\delta > 0$, a polynomial-time $(8+\delta)$-approximation algorithm.

As for the approximability of Minimum Fault-Tolerant Congestion, in case of identical capacities, we present a polynomial-time 4-approximation algorithm. The analysis of this algorithm is based on a combinatorial lemma of independent interest, that also bounds from above the probability that isolated nodes do not appear to a not necessarily connected hypergraph, whose edges fail randomly and independently. In case of related capacities, we restrict our attention to unit size messages, and we present a simple $2\left\lceil\frac{\ln(|K|/\epsilon)}{\ln(1/f_{\max})}\right\rceil$-approximation algorithm.

Up to the best of our knowledge, similar optimization problems concerning the computation of minimum congestion, redundant assignments, subject to the constraint to tolerate random and independent faults with a given probability, have not been studied so far. In [3], the fault-tolerant version of minimizing congestion of Virtual Path layouts in a complete ATM network has been studied mainly under the worst-case fault model, where each layout must tolerate any configuration containing up to a given number of faulty links. In case of random and independent faults, they only provide a trivial logarithmic bound.

Next, Section 2 is devoted to the introduction of the notation used throughout this paper, the formal definitions of Minimum Fault-Tolerant Congestion and Maximum Fault-Tolerant Partition, and the discussion of their complexities. In Section 3, we present a polynomial-time 2-approximation algorithm for Maximum Fault-Tolerant Partition of identical capacity channels, that is also used in Section 5 to obtain a 4-approximation algorithm for Minimum Fault-Tolerant Congestion. Section 4 is devoted to the presentation of two approximation algorithms for Maximum Fault-Tolerant Partition of related capacity channels. In particular, in Section 4.1, we present a simple, logarithmic approximation algorithm, which is also used in Section 6 to obtain a logarithmic approximation algorithm for Minimum Fault-Tolerant Congestion. In Section 4.2, we present a more sophisticated, (8+o(1))-approximation algorithm. Section 5.1 is devoted to the proof of a lower bound for Minimum Fault-Tolerant Congestion on identical capacity channels, which is employed to the analysis of the 4-approximation algorithm presented in Section 5. Section 6 is devoted to the presentation and the analysis of the logarithmic approximation algorithm for Minimum Fault-Tolerant Congestion, in case of unit size messages and related capacity channels. Finally, some directions for further research are discussed in Section 7.

2 Notation and Definitions

Let K be a set of faulty, parallel channels, all having the same source s and destination d. Each channel $i \in K$ has an integer capacity $c_i \geq 1$ and fails independently with probability f_i, for some rational $1 > f_i > 0$. For any subset $K' \subseteq K$, let $\Pr[K']$ denote the reliability of K', that is the probability that at least one channel of K' is active, $\Pr[K'] = 1 - \prod_{i \in K'} f_i$.

Also, $f_{\max} = \max_{i \in K}\{f_i\}$ denotes the failure probability of the most unreliable channel.

Let M be a set of messages to be delivered from s to d. Each message $j \in M$ has an integer size s_j, $S_{\mathrm{tot}} = \sum_{j \in M} s_j$ denotes the total size of M, and $s_{\max} = \max_{j \in M}\{s_j\}$ denotes the size of the largest message.

Then, we proceed to formally define the problems of Minimum Fault-Tolerant Congestion and Maximum Fault-Tolerant Partition and to discuss the complexity of each problem.

2.1 Minimum Fault-Tolerant Congestion

A redundant assignment $\phi : M \mapsto 2^K$ is a function that assigns each message $j \in M$ to a non-empty set of channels $\phi(j) \subseteq K$. An assignment ϕ is feasible for a set of channels $K' \subseteq K$, if, for all $j \in M$, $\phi(j) \cap K' \neq \emptyset$. Given an assignment ϕ, $\Pr[\phi]$ denotes the reliability of ϕ, that is the probability, over the channel availability distribution defined by the f_i's, ϕ to be feasible.

$$\Pr[\phi] = \sum_{\substack{K' \subseteq K \\ \phi \text{ is feasible for } K'}} \left(\prod_{i \in K'} (1 - f_i) \prod_{i \in K - K'} f_i \right)$$

Given a redundant assignment $\phi : M \mapsto 2^K$, a *minimal feasible set of channels for* ϕ is any subset $K' \subseteq K$ such that ϕ is feasible for K', but ϕ is not feasible for any $K'' \subset K'$. A *minimum feasible set of channels for* ϕ is a minimal feasible set for ϕ of minimum cardinality. Moreover, $\mathsf{MF}(\phi)$ denotes the cardinality of any minimum feasible set for ϕ.

Given an assignment ϕ that is feasible for K, $\mathsf{Cong}(\phi)$ denotes the channel congestion, that is the maximum load assigned by ϕ to the channels of K,

$$\mathsf{Cong}(\phi) = \max_{i \in K}\left\{ \sum_{j : i \in \phi(j)} \frac{s_j}{c_i} \right\}$$

Definition 2.1. (Minimum Fault-Tolerant Congestion)
INSTANCE: A set of faulty, parallel channels $K = \{(f_1, c_1), \ldots, (f_n, c_n)\}$. Each channel i has an integer capacity $c_i \geq 1$ and fails independently with probability f_i, for some rational $1 > f_i > 0$.
A set of messages $M = \{s_1, \ldots, s_m\}$ to be delivered over K. Each message j has an integer size $s_j \geq 1$.
A fault-tolerance constraint $(1 - \epsilon)$, for some rational $1 > \epsilon \geq \prod_{i=1}^{n} f_i$.
FEASIBLE SOLUTION: A $(1 - \epsilon)$-fault-tolerant, redundant assignment $\phi : M \mapsto 2^K$, i.e. an assignment of each message $j \in M$ to a non-empty set of channels $\phi(j) \subseteq K$, such that $\Pr[\phi] \geq 1 - \epsilon$.
OBJECTIVE: Minimize $\mathsf{Cong}(\phi) = \max_{i \in K}\left\{ \sum_{j : i \in \phi(j)} \frac{s_j}{c_i} \right\}$.

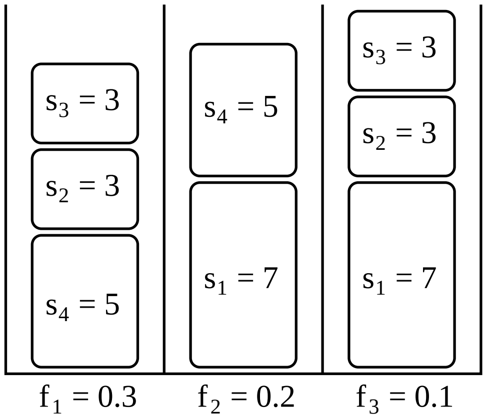

Fig. 1. The optimal assignment $\phi^{\star}$.

We distinguish the identical capacities case, where all the channels have unit capacity (i.e. $c_i = 1$), and the related capacities case, where each channel can have an arbitrary integer capacity. In the related capacities case, we further assume that the channels are numbered in non-increasing order of their capacities, i.e. $c_1 \geq \cdots \geq c_n \geq 1$.

For example, we consider an instance of Minimum Fault-Tolerant Congestion consisting of three identical capacity channels of failure probabilities $f_1 = 0.3, f_2 = 0.2$, and $f_3 = 0.1$, four messages of sizes $s_1 = 7$, $s_2 = s_3 = 3$, and $s_4 = 5$, and a fault-tolerance constraint $1 - \epsilon = 0.9$. A lower bound on the optimal Fault-Tolerant Congestion is $18 \cdot \frac{2}{3} = 12$, because any 0.9-fault-tolerant assignment that does not assign all the messages to the third channel, must assign each message to at least two different channels. An optimal 0.9-fault-tolerant assignment $\phi^{\star}$ has $\phi^{\star}(1) = \{2, 3\}$, $\phi^{\star}(2) = \phi^{\star}(3) = \{1, 3\}$, and $\phi^{\star}(4) = \{1, 2\}$ (see also Figure 1). It is easy to verify that the reliability of $\phi^{\star}$ is $\Pr[\phi^{\star}] = 0.902$, and $\mathsf{Cong}(\phi^{\star}) = 13$.

The problem of Minimum Fault-Tolerant Congestion remains $\mathcal{NP}$-hard for identical capacity and failure probability channels, since it is $\mathcal{NP}$-complete even to determine the minimum makespan for scheduling a set of jobs on a set of reliable, identical parallel machines (e.g. [2]). Furthermore, Minimum Fault-Tolerant Congestion, especially in case of related capacity channels, does not seem to belong to the class $\mathcal{NP}$, because, for an arbitrary redundant assignment ϕ, it is $\#\mathcal{P}$-complete to exactly compute the reliability of ϕ. In particular, given a set of channels K, each failing independently with probability $f = \frac{1}{2}$, a set M of m unit size messages, and a redundant assignment $\phi : M \mapsto 2^K$, computing $\Pr[\phi]$ is equivalent to #Monotone-Sat originally shown $\#\mathcal{P}$-complete by Valiant [4].

2.2 Maximum Fault-Tolerant Partition

The problem of the Maximum Fault-Tolerant Partition arises from the following natural strategy for computing a $(1-\epsilon)$-fault-tolerant, redundant assignment ϕ.

(1) Compute a collection of reliable effective channels by partitioning some $K' \subseteq K$ into disjoint groups $\mathcal{K} = \{K_1, \ldots, K_\nu\}$, such that the probability of at least one channel being active in each group is at least $(1-\epsilon)$.

(2) Use an appropriate algorithm for scheduling the message set M on the set $\mathcal{K}$ of reliable effective channels. For all messages $j \in M$ assigned to the effective channel K_l, set $\phi(j) = K_l$.

The first step of this approach actually determines an upper bound on the amount of redundancy being necessary for satisfying the fault-tolerance constraint. Moreover, if we set the effective capacity $C(K_l)$ equal to the minimum capacity of the corresponding group, $C(K_l) = \min_{i \in K_l}\{c_i\}$, $1 \le l \le \nu$, then the makespan of the non-redundant schedule obtained in the second step equals the congestion of the redundant assignment ϕ.

The redundant assignments that can be produced by this approach are called *partition assignments*. In particular, an assignment $\phi : M \mapsto 2^K$ is called a *ν-partition assignment* if, for any pair $j_1, j_2 \in M$, either $\phi(j_1) = \phi(j_2)$ or $\phi(j_1) \cap \phi(j_2) = \emptyset$, and ϕ assigns the messages of M to exactly ν disjoint channel subsets $\{K_1, \ldots, K_\nu\}$.

Since there exist many efficient algorithms for the implementation of the second step (e.g. see Chapter 1 of [5]), we focus on the design and analysis of approximation algorithms for the first step, that is the computation of a $(1-\epsilon)$-fault-tolerant collection $\{K_1, \ldots, K_\nu\}$ of disjoint channel subsets.

Definition 2.2. (Maximum Fault-Tolerant Partition)
INSTANCE: A set of channels $K = \{(f_1, c_1), \ldots, (f_n, c_n)\}$. Each channel i has an integer capacity $c_i \ge 1$ and fails independently with probability f_i, for some rational $1 > f_i > 0$.
A fault-tolerance constraint $(1-\epsilon)$, for some rational $1 > \epsilon \ge \prod_{i=1}^n f_i$.
FEASIBLE SOLUTION: A partition of a subset $K' \subseteq K$ into disjoint groups $\mathcal{K} = \{K_1, \ldots, K_\nu\}$ such that:

$$\Pr[\mathcal{K}] = \Pr[K_1 \wedge \ldots \wedge K_\nu] = \prod_{l=1}^{\nu} \Pr[K_l] = \prod_{l=1}^{\nu} \left(1 - \prod_{i \in K_l} f_i\right) \ge 1 - \epsilon$$

OBJECTIVE: Maximize the total effective capacity of the partition $\mathcal{K}$:

$$C(\mathcal{K}) = \sum_{l=1}^{\nu} C(K_l) = \sum_{l=1}^{\nu} \min_{i \in K_l}\{c_i\}$$

The term "partition" is somewhat abused, because the definition of Fault-Tolerant Partition allows $K' = \bigcup_{l=1}^{\nu} K_l \subset K$. This is crucial in the related capacities case, because there exist many instances, where the optimal solution is a partition of a proper subset K' of K.

For example, we consider the following collection of related capacity channels $K = \{(0.05, 1000), (0.05, 500), (0.05, 200), (0.05, 100), (0.05, 90), (0.05, 80), (0.05, 70)\}$. If the fault-tolerance constraint is $1 - \epsilon = 0.95$, the optimal solution only consists of $\{(0.05, 1000)\}$ and has objective value equal to 1000. If $1 - \epsilon = 0.945$, there exists a solution of total effective capacity 1270 consisting of the groups $K_1 = \{(0.05, 1000)\}$, $K_2 = \{(0.05, 500), (0.05, 200)\}$, and $K_3 = \{(0.05, 100), (0.05, 90), (0.05, 80), (0.05, 70)\}$. We can further increase the optimal effective capacity to 1580 by relaxing the fault-tolerance constraint to $1 - \epsilon = 0.9$. In this case, the optimal solution consists of the groups $K_1 = \{(0.05, 1000)\}$, $K_2 = \{(0.05, 500)\}$, and $K_3 = \{(0.05, 200), (0.05, 100), (0.05, 90), (0.05, 80)\}$. Notice that in the first and the third case, the optimal solutions correspond to partitions of proper subsets of K.

In the identical capacities case, where the objective is simply to maximize the number of groups ν, the addition of some items to K' cannot decrease the total effective capacity of the resulting partition. Hence, in the identical capacities case, we can always assume that $\bigcup_{l=1}^{\nu} K_l = K$.

The Fault-Tolerant Partition problem can be thought as a version of Bin Covering [6], where, instead of a threshold T on the total size of each separate bin, we have to cover a constraint on the product of the bin sizes. Therefore, any feasible solution of a Bin Covering instance can be mapped to a feasible solution of a corresponding instance of Fault-Tolerant Partition. On the other hand, there may exist feasible Fault-Tolerant Partitions (including the optimal one) that are not feasible solutions of the corresponding Bin Covering instance, because they contain some underfilled bins.

As for the complexity of Maximum Fault-Tolerant Partition, there exists a direct transformation from Subset Product (problem SP14, [2]) showing that Fault-Tolerant Partition into two groups is $\mathcal{NP}$-complete, even for channels of identical capacity. In particular, it is not hard to verify that, given a set $K = \{f_1, \ldots, f_n\}$, for rational $1 > f_i > 0$, and a rational $\epsilon > 0$, it is $\mathcal{NP}$-complete to decide if K can be partitioned into two sets K_1, K_2 such that $\Pr[K_1] \Pr[K_2] \geq 1 - \epsilon$.

3 Partition of Identical Capacity Channels

The similarity of Fault-Tolerant Partition and Bin Covering suggests that it may be possible to design an approximation algorithm for the former problem using the ideas of some algorithm for the latter one. In this section, we present a 2-approximation algorithm for Maximum Fault-Tolerant Partition of identical capacity channels based on the Next Fit algorithm for Bin Covering [6], and an additional idea for handling oversized/over-reliable items.

The analysis of Next Fit is based on the fact that the total size of the items divided by the threshold T provides an upper bound on the optimal number of bins. In case of Bin Covering, this upper bound is a reasonable one, because, since any feasible solution must cover all the bins up to T, there is no need to consider oversized items, that are items of size greater than T.

Similarly, in case of Fault-Tolerant Partition, given a set $K = \{f_1, \ldots, f_n\}$, the solution x_0 to the following equation provides a trivial upper bound on the optimal number of groups.

$$\left[1 - \left(\prod_{i=1}^{n} f_i\right)^{1/x_0}\right]^{x_0} = 1 - \epsilon \tag{1}$$

Equation (1) implies that if non-integral bins/groups and placement of items were allowed, then an optimal solution would consist of x_0 groups, each of reliability $1 - \alpha_0$, where $\alpha_0 = (\prod_{i=1}^{n} f_i)^{1/x_0}$. Assume that the items are sorted in non-increasing order of reliability, and $f_1 < \alpha_0$. Then, since each item must be placed into a single group, f_1 can contribute at most 1 to any optimal solution, i.e. f_1 contributes a group $K_1 = \{f_1\}$. However, f_1 contributes more than 1 to x_0 defined by Equation (1). Therefore, in case that f_1 is an over-reliable item, i.e. it is more reliable than the average reliability of a partition into x_0 groups, Equation (1) provides a poor upper bound on the optimal number of groups, because the contribution of f_1 is over-estimated.

On the other hand, an optimal solution will take advantage of the over-reliable group containing f_1 to relax the average reliability of the remaining groups. In particular, given that f_1 contributes 1 to the optimal number of groups, the remaining items can contribute up to x_1 groups to any optimal solution, where x_1 is defined by the following equation.

$$\left[1 - \left(\prod_{i=2}^{n} f_i\right)^{1/x_1}\right]^{x_1} = \frac{1-\epsilon}{1-f_1}$$

It is easy to see that $\alpha_1 = (\prod_{i=2}^{n} f_i)^{1/x_1} > \alpha_0$, because $f_1 < \alpha_0$. This situation must go on, until no more over-reliable items exist, i.e. until we find the first index l such that $f_{l+1} > \alpha_l$.

The algorithm Next Fit Decreasing - NFD (Figure 2) initially computes a collection of l over-reliable items placed into the single item groups $K_1, \ldots, K_l$ (steps (1) - (3)). The remaining items are partitioned using Next Fit with threshold $1-\alpha_l$. For example, we consider an instance consisting of four items $f_1 = 0.01$, $f_2 = 0.05$, $f_3 = 0.08$, and $f_4 = 0.09$, and a fault-tolerance constraint $1 - \epsilon = 0.9$. The initial upper bound $x_0 = 3.549$ results in $\alpha_0 = 0.0293$. Since $f_1 = 0.01 < 0.0293$, the item f_1 is over-reliable and contributes $\log_{0.0293} 0.01 = 1.30452$ to x_0. Therefore, f_1 should be placed into a single item group $K_1 = \{f_1\}$. Then, $x_1 = 2.490$ and $\alpha_1 = 0.0414 > \alpha_0$. The remaining three items are placed into a single group by Next

Algorithm Next Fit Decreasing – NFD

Input: $K = \{f_1, \ldots, f_n\}$, failure probabilities $1 > f_i > 0$.

Fault-tolerance constraint $(1-\epsilon)$, $1 > \epsilon \geq \prod_{i \in K} f_i$.

Output: A $(1-\epsilon)$-fault-tolerant partition of K into λ disjoint groups.

(1) Sort the items of K so that $f_1 \leq f_2 \leq \cdots \leq f_n$, i.e. in non-increasing order of reliability.

(2) Compute the first index l, $0 \leq l < n$, such that $f_{l+1} > \alpha_l$, where $\alpha_l = F_l^{1/x_l}$, and x_l, F_l are defined by the following equations:

$$\left(1 - F_l^{1/x_l}\right)^{x_l} = \frac{1-\epsilon}{P_l} \ ,\ F_l = \prod_{i=l+1}^{n} f_i \ ,\ \ P_l = \prod_{i=1}^{l}(1-f_i) \tag{2}$$

(3) For $j = 1, \ldots, l$, $K_j = \{f_j\}$, i.e. the group K_j only consists of the item f_j.

(4) The set $\{f_{l+1}, \ldots, f_n\}$ is partitioned using Next Fit [6] with threshold $1 - \alpha_l$.

```
j = l + 1;  λ = l + 1;
while j ≤ n do
    if Pr[K_λ] = 1 − ∏_{i∈K_λ} f_i < 1 − α_l then
        K_λ = K_λ ∪ {f_j};     /* If K_λ is not yet filled, place f_j to K_λ */
    else
        λ = λ + 1;  K_λ = {f_j};     /* Else place f_j to a new group K_{λ+1} */
    j = j + 1;
end_while;
if Pr[K_λ] < 1 − α_l then             /* If the last group is not filled, */
    λ = λ − 1;  K_λ = K_λ ∪ K_{λ+1};   /* merge the last two groups */
```

Fig. 2. The Algorithm Next Fit Decreasing (NFD).

Fit with threshold $1 - \alpha_1 = 0.9586$, while the optimal solution consists of three groups $K_1 = \{f_1\}$, $K_2 = \{f_2\}$, and $K_3 = \{f_3, f_4\}$.

The following theorem shows that the approximation ratio of Next Fit Decreasing is actually determined by the approximation ratio of Next Fit.

Theorem 3.1. Next Fit Decreasing *(Figure 2) runs in time $\mathcal{O}(\log n(n + \sum \log(1/f_i)))$ and is a 2-approximation algorithm for Maximum Fault-Tolerant Partition of identical capacity channels.*

Proof. We start by showing that the partitions produced by NFD are indeed $(1-\epsilon)$-fault-tolerant. Let $P_l = \prod_{i=1}^{l}(1-f_i) = \Pr[K_1 \wedge \ldots \wedge K_l]$ be the probability that all the first l groups $K_1, \ldots K_l$, each containing a single item, are active. Therefore, the remaining groups

$K_{l+1}, \ldots, K_\lambda$ must contain an active item with probability at least $\frac{1-\epsilon}{P_l}$. Since, for $i = l + 1, \ldots, \lambda$, $\Pr[K_i] \geq 1 - \alpha_l$ and $\alpha_l = F_l^{1/x_l}$, the number of groups $K_{l+1}, \ldots, K_\lambda$ cannot be more than x_l, i.e. $\lambda - l \leq x_l$. Hence,

$$\Pr[K_{l+1} \wedge \ldots \wedge K_\lambda] = \prod_{i=l+1}^{\lambda} \Pr[K_i] \geq (1-\alpha_l)^{\lambda - l} \geq (1-\alpha_l)^{x_l} = \frac{1-\epsilon}{P_l} \text{ , and}$$

$$\Pr[K_1 \wedge \ldots \wedge K_\lambda] \geq P_l \cdot \frac{1-\epsilon}{P_l} = 1 - \epsilon \ .$$

Performance: Initially, by an argument similar to one used in the analysis of Next Fit for Bin Covering [6], we show that NFD always produces no less than $l + \frac{\lfloor x_l \rfloor}{2}$ groups, where x_l is defined by Equation (2). In particular, we show that NFD partitions the set

$$K_{l+} = \{f_{l+1}, \ldots, f_n\} = K - \bigcup_{i=1}^{l} K_i$$

into at least $\frac{\lfloor x_l \rfloor}{2}$ disjoint groups such that $\Pr[K_{l+1} \wedge \ldots \wedge K_\lambda] = \prod_{i=l+1}^{\lambda} \Pr[K_i] \geq \frac{1-\epsilon}{P_l}$. Since, for all $f_j \in K_{l+}$, $f_j > \alpha_l$, for all but possibly the last group containing items of K_{l+}, $1 - \alpha_l \leq \Pr[K_i] < 1 - \alpha_l^2$. Moreover, the last group has $\Pr[K_\lambda] < 1 - \alpha_l^3$. Therefore,

$$F_l = \prod_{i=l+1}^{\lambda} (1 - \Pr[K_i]) > \alpha_l^{2(\lambda - l)+1}$$

The definition of $\alpha_l = F_l^{1/x_l}$ implies that $\lambda - l > \frac{1}{2}(x_l - 1)$, and, since the number of groups must be integer, $\lambda \geq l + \frac{\lfloor x_l \rfloor}{2}$.

Next, we show that any optimal partition cannot have more than $l + x_l$ groups. We consider the most reliable, $(1-\epsilon)$-fault-tolerant, optimal partition $\mathcal{K}^\star = \{K_1^\star, \ldots, K_\nu^\star\}$. Let $\mathsf{MRI}(K_i^\star) = \min_{j \in K_i^\star}\{f_j\}$ be the Most Reliable Item of $K_i^\star$, and assume that the groups $K_i^\star$ are numbered in non-decreasing order of their $\mathsf{MRI}(K_i^\star)$ values, i.e. $\mathsf{MRI}(K_1^\star) \leq \mathsf{MRI}(K_2^\star) \leq \cdots \leq \mathsf{MRI}(K_\nu^\star)$. Therefore, the first l groups in $\mathcal{K}^\star$ contain the items $f_1, \ldots, f_l$, i.e. the l most reliable items, and the following inclusion holds:

$$\bigcup_{i=1}^{l} K_i \subseteq \bigcup_{i=1}^{l} K_i^\star \tag{3}$$

Let $P_l^\star = \prod_{i=1}^{l} \Pr[K_i^\star]$, $K_{l+}^\star = K - \bigcup_{i=1}^{l} K_i^\star$, and $F_l^\star = \prod_{j \in K_{l+}^\star} f_j$. Equation (3) implies that $P_l \leq P_l^\star$ and $F_l \leq F_l^\star$. In case that (3) holds with equality, the set $K_{l+}^\star$ cannot contribute more than x_l groups to any optimal partition. Otherwise, each group would have reliability less than $1 - \alpha_l$ and, since the number of groups would be greater than x_l, the reliability of such a partition would be less than $\frac{1-\epsilon}{P_l}$.

Then, we assume that the most reliable, optimal solution $\mathcal{K}^\star$ corresponds to $K_{l+}^\star \subset K_{l+}$, and $K_{l+}^\star$ is partitioned into $x_l^\star = \nu - l > x_l$ groups of total reliability at least $\frac{1-\epsilon}{P_l^\star}$, and we show that this contradicts to the selection of $\mathcal{K}^\star$ as the most reliable, optimal partition. Since $\bigcup_{i=1}^{l} K_i \subset \bigcup_{i=1}^{l} K_i^\star$, some of the groups $K_1^\star, \ldots, K_l^\star$ of the optimal solution $\mathcal{K}^\star$ must contain more than one items. Let $K_{z_1}^\star$, $1 \leq z_1 \leq l$, be the first such group of $\mathcal{K}^\star$. Therefore, the item f_{z_1} belongs to $K_{z_1}^\star$. Also, let f'_{z_1} be an item other than f_{z_1} placed into $K_{z_1}^\star$ by the optimal solution $\mathcal{K}^\star$. Clearly, $\Pr\left[K_{z_1}^\star - \{f'_{z_1}\}\right] \geq 1 - \alpha_l$, because $f_{z_1} \leq \alpha_l$. Furthermore, since $F_l \leq F_l^\star$, $x_l^\star > x_l$, and $\alpha_l = F_l^{1/x_l}$, $\mathcal{K}^\star$ must contain another group $K_{z_2}^\star$, $l+1 \leq z_2 \leq \nu$, such that $\Pr[K_{z_2}^\star] < 1 - \alpha_l$.

Since $\Pr\left[K_{z_2}^\star\right] < \Pr\left[K_{z_1}^\star - \{f'_{z_1}\}\right]$, the partition obtained by removing f'_{z_1} from $K_{z_1}^\star$ and adding it to $K_{z_2}^\star$ is also optimal and strictly more reliable than $\mathcal{K}^\star$. This contradicts the selection of $\mathcal{K}^\star$ as the most reliable, optimal partition, and implies that any set $K_{l+}^\star \subseteq K_{l+}$ cannot contribute more than x_l groups to any optimal solution.

Since the optimal number of groups $\nu \leq l + x_l$ must be integer, the above discussion implies that $2\lambda \geq \nu + l$ and the number of groups obtained by NFD is at least half the optimal number of groups. Then, we show that there exists a family of instances such that $2\lambda = \nu$.

Let $x \geq 1$ be an integer and let α, ϵ be rational numbers, $1 > \alpha > 0$, $1 > \epsilon > 0$, that fulfill $(1-\alpha)^{2x} = 1 - \epsilon$. Also, let δ be any small rational that fulfills $0 < \delta < \left(\frac{1}{\alpha}\right)^{1/2x} - 1$. Consider an instance of Fault-Tolerant Partition consisting of $4x$ items, where $2x$ of them, $f_1, \ldots, f_{2x}$, have failure probability equal to $\alpha(1+\delta)$ and the remaining $2x$ items, $f_{2x+1}, \ldots, f_{4x}$, have failure probability equal to $\frac{1}{1+\delta}$. Also, let the fault-tolerance constraint be equal to $(1-\epsilon)$. By the choice of x, α and ϵ, the optimal partition consists of $2x$ groups $\{f_i, f_{2x+i}\}$, $i = 1, \ldots, 2x$. Furthermore, since $\alpha(1+\delta) < \frac{1}{1+\delta}$, and $f_1 > \alpha$, NFD simply uses $\mathsf{Next\ Fit}$ with threshold $1 - \alpha$. Therefore, it places the first $2x$ items into x groups, and all the remaining $2x$ items into the last group, because $(1 - (1+\delta)^{-2x}) < 1 - \alpha$ by the choice of δ.

Complexity: The complexity of the algorithm is dominated by the steps (1) and (2). $\mathcal{O}(n \log n)$ time is needed for the step (1). As for the step (2), the value of l can be decided using binary search because, by the definition of l, for all $l' > l$, $f_{l'+1} > \alpha_{l'}$ and, for all $l' < l$, $f_{l'+1} \leq \alpha_{l'}$. Additionally, each iteration of binary search can be implemented in time $\mathcal{O}(n + \sum_{i \in K} \log(1/f_i))$, because:

1. The function $g(y) = (1-y)^{\ln(F_l)/\ln(y)}$ is monotone decreasing with respect to y, and
2. Even though α_l can be a real number, we need to determine only the first $\sum_{i=1}^{n} \lceil \log(1/f_i) \rceil$ bits of α_l in order to (correctly) perform the subsequent comparisons.

□

In the sequel, we extensively use the upper bound $(l + x_l)$ on the number of reliable groups that can be obtained from a set of identical capacity channels. In particular, given a set K of identical capacity channels and a fault-tolerance constraint $(1-\epsilon)$, $\mathsf{IUB}(K, 1-\epsilon) = l + x_l$

Algorithm Safe Partition – SP

Input: $K = \{(f_1, c_1) \ldots, (f_n, c_n)\}$, failure probabilities $1 > f_i > 0$, capacities c_i, $c_1 \geq \cdots \geq c_n$.
Fault-tolerance constraint $(1-\epsilon)$, $1 > \epsilon \geq \prod_{i \in K} f_i$.

Output: A $(1-\epsilon)$-fault-tolerant partition of K into disjoint groups.

```
j = 1;  λ = 1;  K_λ = ∅;
while j ≤ n do
    if Pr[K_λ] = 1 − ∏_{i∈K_λ} f_i < 1 − ε/n then K_λ = K_λ ∪ {(f_j, c_j)};
    else λ = λ + 1;  K_λ = {(f_j, c_j)};
    j = j + 1;
end_while;
if Pr[K_λ] < 1 − ε/n then λ = λ − 1;
C(𝒦) = Σ_{l=1}^{λ} C(K_l) = Σ_{l=1}^{λ} min_{i∈K_l}{c_i};
Let d ≥ 0 be the last index such that ∏_{i=1}^{d} f_i > ε.
If c_{d+1} ≥ C(𝒦) then return {{(f_1, c_1), ..., (f_{d+1}, c_{d+1})}};
else return 𝒦 = {K_1, ..., K_λ};
```

Fig. 3. The Algorithm Safe Partition (SP).

bounds from above the number of groups that can be produced by K with constraint $(1-\epsilon)$. The bound $\mathsf{IUB}(K, 1-\epsilon) = l + x_l$ consists of the integer l denoting the number of over-reliable items, and the real x_l denoting the optimal non-integral number of groups that can be obtained from the instance $\left(K_{l^+}, \frac{1-\epsilon}{P_l}\right)$, if non-integral placement of items is allowed.

4 Partition of Related Capacity Channels

4.1 A Simple Logarithmic Approximation Algorithm

The Safe Partition – SP algorithm (Figure 3) exploits a simple idea to approximate Maximum Fault-Tolerant Partition of related capacity channels within a logarithmic factor. Safe Partition starts by applying Next Fit with threshold equal to $1-\frac{\epsilon}{n}$. Since any feasible solution cannot have more than n groups, the resulting partition is always $(1-\epsilon)$-fault-tolerant. Then, Safe Partition computes the largest effective capacity, $(1-\epsilon)$-fault-tolerant group consisting of the first $d+1$ channels, where d is the largest index such that $\prod_{i=1}^{d} f_i > \epsilon$. The Safe Partition algorithm returns the best of these feasible solutions. The analysis of Safe Partition is simple and based on the facts that all the groups of both the aforementioned partitions have cardinality at most $\left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$, and any $(1-\epsilon)$-fault-tolerant partition cannot have effective capacity more than $\sum_{i=d+1}^{n} c_i$.

Lemma 4.1. Safe Partition – SP *(Figure 3) is a polynomial-time* $2\left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$*-approximation algorithm for Maximum Fault-Tolerant Partition of related capacity channels, where* $f_{\max} = \max_{i \in K}\{f_i\}$.

Proof. We start by observing that the output of SP is always $(1-\epsilon)$-fault-tolerant.

Let $K_{\lambda+1} = K - \bigcup_{l=1}^{\lambda} K_l$. For all $l = 1, \ldots, \lambda+1$, the group K_l cannot contain more than $n^\star = \left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$ channels, because

$$1 - f_{\max}^{\left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil} \geq 1 - \frac{\epsilon}{n} \ .$$

Since the effective capacity of each group K_l cannot be less than any $c_i \in K_{l+1}$, we obtain that, for all $l = 1, \ldots, \lambda$,

$$n^\star C(K_l) \geq |K_{l+1}| C(K_l) \geq \sum_{i \in K_{l+1}} c_i \ .$$

Similarly, $n^\star c_{d+1} \geq |K_1| c_{d+1} \geq \sum_{i=d+1}^{|K_1|} c_i$.

To obtain an upper bound on the optimal effective capacity, notice that, by the definition of d as the largest index such that $\prod_{i=1}^{d} f_i > \epsilon$, any $(1-\epsilon)$-fault-tolerant partition must place each channel (f_j, c_j), $j = 1, \ldots, d$, into the same group with some of the channels (f_i, c_i), $i = d+1, \ldots, n$. Therefore, no $(1-\epsilon)$-fault-tolerant partition can have effective capacity more than $\sum_{i=d+1}^{n} c_i$. Hence, the approximation ratio can be derived by the following inequality

$$n^\star \left(c_{d+1} + \sum_{l=1}^{\lambda} C(K_l) \right) \geq \sum_{i=d+1}^{n} c_i \ .$$

□

4.2 A Constant Factor Approximation Algorithm

In this section, we present Capacity Class Partition – CCP (Figure 4), that is a constant factor approximation algorithm for Maximum Fault-Tolerant Partition of related capacity channels. This algorithm divides the original instance into classes I_j of almost identical capacity channels. The analysis of Capacity Class Partition is based on a technical lemma stating that there exists an allocation of portions $(1-\epsilon)^{\beta_j^\star}$ of the fault-tolerance constraint to the classes I_j, $\sum_j \beta_j^\star \leq 1$, so that the total effective capacity of an optimal solution can be bounded from above by the sum, over all j, of I_j's capacity times the upper bound $\mathsf{IUB}\left(I_j, (1-\epsilon)^{\beta_j^\star}\right)$ on the number of groups obtained from the class I_j with fault-tolerance constraint $(1-\epsilon)^{\beta_j^\star}$.

In order to approximate the values $\beta_j^\star$, the Capacity Class Partition algorithm computes an appropriately selected set of samples $\beta_j(i)$ on the $[0,1]$-interval, and for each sample $\beta_j(i)$, Algorithm Capacity Class Partition – CCP

Input: $K = \{(f_1, c_1) \ldots, (f_n, c_n)\}$, failure probabilities $1 > f_i > 0$, capacities c_i, $c_1 \geq \cdots \geq c_n$. Fault-tolerance constraint $(1-\epsilon)$, $1 > \epsilon \geq \prod_{i \in K} f_i$.

Output: A $(1-\epsilon)$-fault-tolerant partition of K into disjoint groups.

(1) Let $d \geq 0$ be the last index such that $\prod_{i=1}^{d} f_i > \epsilon$, and $\psi = \lfloor \log c_{d+1} \rfloor$. For all $i = 1, \ldots, d$, set $c_i = c_{d+1}$. For all $j = 0, \ldots, \psi$, let $I_j = \{f_i : (f_i, c_i) \in K \wedge 2^j \leq c_i < 2^{j+1}\}$ be the class of channels, whose capacities belong to $[2^j, 2^{j+1})$. In the following, we assume that, for all $f_i \in I_j$, $c_i = 2^j$.

(2) For each class I_j, compute a set Γ_j of pairs $(\nu_j(i), \beta_j(i))$, where $1 \geq \beta_j(i) \geq 0$, and $\nu_j(i)$'s are defined by $\nu_j(i) = \mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j(i)}\big)$. Each Γ_j must contain pairs $(\mu, \beta_j(i))$, for all integers $\mu = 0, 1, \ldots, \lfloor \mathsf{IUB}(I_j, 1-\epsilon) \rfloor$, and, for all $i \geq 0$, $\nu_j(i) \geq \nu_j(i+1) \geq \nu_j(i) - 1$.

(3) For each class I_j, compute a set $\hat{\Gamma}_j$ of pairs $(\lambda_j(i), \beta_j(i))$ as follows: For all $(\nu_j(i), \beta_j(i)) \in \Gamma_j$, use NFD to compute the pair $(\lambda_j(i), \beta_j(i))$, where $\lambda_j(i) = \mathsf{NFD}\big(I_j, (1-\epsilon)^{\beta_j(i)}\big)$. A detailed implementation of the steps (2) and (3) is described in Figure 5.

(4) For each class I_j, select exactly one pair $(\lambda_j(i_j^\star), \beta_j(i_j^\star))$ from $\hat{\Gamma}_j$ so as to maximize the function

$$\sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star) \quad , \quad \text{subject to} \quad \sum_{j=0}^{\psi} \beta_j(i_j^\star) \leq 1 \quad .$$

(5) If $c_{d+1} \geq \sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star)$ then return a single group $\{(f_1, c_1), \ldots, (f_{d+1}, c_{d+1})\}$. Otherwise, return $\mathcal{K} = \bigcup_{j=0}^{\psi} \mathcal{K}^j$, where $\mathcal{K}^j = \{K_1^j, \ldots, K_{\lambda_j(i_j^\star)}^j\}$ is the partition produced by $\mathsf{NFD}\big(I_j, (1-\epsilon)^{\beta_j(i_j^\star)}\big)$.

Fig. 4. The Algorithm Capacity Class Partition (CCP).

evaluates the number of groups $\lambda_j(i) = \mathsf{NFD}\big(I_j, (1-\epsilon)^{\beta_j(i)}\big)$ produced by NFD from the capacity class I_j with fault-tolerance constraint $(1-\epsilon)^{\beta_j(i)}$. For all the classes I_j, the profit-size pairs $(\lambda_j(i), \beta_j(i))$ form an instance of Generalized Knapsack, whose solution suggests a near optimal allocation of portions $(1-\epsilon)^{\beta_j(i_j^\star)}$ of the fault-tolerance constraint to each class I_j. Then, a feasible solution $\mathcal{K}$ consists of the union of the partial solutions produced by NFD on instances $\big(I_j, (1-\epsilon)^{\beta_j(i_j^\star)}\big)$. The Capacity Class Partition algorithm returns the best of $\mathcal{K}$ and the largest effective capacity, $(1-\epsilon)$-fault-tolerant group consisting of the first $d+1$ channels. The following shows that this approach indeed yields a constant factor approximation algorithm.

Theorem 4.2. *For any constant $\delta > 0$,* Capacity Class Partition – CCP *(Figure 4) is a polynomial-time $(8+\delta)$-approximation algorithm for Maximum Fault-Tolerant Partition of* *related capacity channels. Moreover, the running time of* CCP *is polynomial in* $\frac{1}{\delta}$.

Proof. We start by observing that wlog. we can assume that $c_i = c_{d+1}$, for all $i = 1, \ldots, d$, because, by the definition of d, all the channels with index no more than d must be groupped together with channels of capacity no more than c_{d+1} in any $(1-\epsilon)$-fault-tolerant partition. Therefore, this cannot decrease the effective capacity of an optimal solution. At the steps (2) and (3), CCP computes appropriately selected samples $\beta_j(i)$'s used for the formulation of the Generalized Knapsack instance at the step (4). In particular, $\beta_j(i)$'s are computed so as the corresponding $\nu_j(i)$'s to be integers, and not differ too much from each other, i.e. $1 \geq \nu_j(i) - \nu_j(i+1) \geq 0$. In Section 4.2.1, we provide a detailed implementation of the steps (2) and (3) (Figure 5), and we prove that an appropriate set of samples $\beta_j(i)$ can be computed in polynomial time.

Feasibility: Clearly, if a single group of capacity c_{d+1} is returned, then $\prod_{i=1}^{d+1} f_i \leq \epsilon$. Otherwise, the feasibility of NFD implies that, for all $j = 0, \ldots, \psi$, $\Pr[\mathcal{K}^j] \geq (1-\epsilon)^{\beta_j(i_j^\star)}$, and $\Pr[\mathcal{K}] = \prod_{j=0}^{\psi} \Pr[\mathcal{K}^j] \geq 1-\epsilon$, since $\sum_{j=0}^{\psi} \beta_j(i_j^\star) \leq 1$.

Performance: In the following, we only consider channel capacities $c_{d+1}, \ldots, c_n$ that are integer powers of 2. Obviously, the original capacities $c_{d+1}, \ldots, c_n$ can be reduced to the nearest integer power of 2 by only losing a factor of 2 in the approximation ratio. Therefore, the capacities of all the channels belonging to each class I_j are assumed to be equal to 2^j.

The performance analysis of the CCP algorithm is based on the following technical lemma, whose proof is deferred to Section 4.2.2. This lemma implies that Fault-Tolerant Partition of related capacities can be reduced to Fault-Tolerant Partition of identical capacities by appropriately allocating portions $(1-\epsilon)^{\beta_j^\star}$ of the fault-tolerance constraint to the capacity classes I_j.

Lemma 4.3. *There exist reals* $0 \leq \beta_j^\star \leq 1$, $j = 0, \ldots, \psi$, $\sum_{j=0}^{\psi} \beta_j^\star \leq 1$, *such that the objective value* $C(\mathcal{K}^\star)$ *of an optimal partition* $\mathcal{K}^\star$ *fulfills the following inequality:*

$$C(\mathcal{K}^\star) \leq \sum_{j=0}^{\psi} 2^j \, \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j^\star}\Big)$$

Since $\lambda_j(i)$'s are computed by NFD, for all $i \geq 0$, $\lambda_j(i) \geq \frac{\lfloor \nu_j(i) \rfloor}{2}$. Thus, for the optimal pairs $(\lambda_j(i_j^\star), \beta_j(i_j^\star)) \in \hat{\Gamma}_j$, we have

$$\sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star) \geq \sum_{j=0}^{\psi} 2^j \, \frac{\lfloor \nu_j(i_j^\star) \rfloor}{2} \,, \tag{4}$$

where $\nu_j(i_j^\star) = \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i_j^\star)}\Big)$.

Then, we consider the largest valued sample $\beta_j(i)$ for the class I_j that does not exceed $\beta_j^\star$, and we show that the corresponding $\nu_j(i) = \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$ satisfies the inequality

$\lfloor \nu_j(i) \rfloor \geq \mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j^\star}\big) - 1$. To see this, recall that the samples $\beta_j(i)$ are computed so as to satisfy the following properties (see also the step (2) of CCP, Figure 4, and the implementation of the steps (2) and (3), Figure 5):

(a) Γ_j contains pairs $(\mu, \beta_j(i))$, for all integers $\mu = 0, 1, \ldots, \lfloor \mathsf{IUB}(I_j, 1-\epsilon) \rfloor$, and

(b) For all $i \geq 0$, $\nu_j(i) \geq \nu_j(i+1) \geq \nu_j(i) - 1$.

Therefore, if the sample $\beta_j(i)$ corresponds to an integer valued $\nu_j(i)$, then $\nu_j(i)$ cannot be less than $\mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j^\star}\big) - 1$, because of (b). Otherwise, the smallest sample $\beta_j(i')$ exceeding $\beta_j^\star$ must correspond to a $\nu_j(i')$ such that $\mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j^\star}\big) \leq \nu_j(i') \leq \lceil \nu_j(i) \rceil$, because of the property (a). Hence, $\lfloor \nu_j(i) \rfloor \geq \mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j^\star}\big) - 1$. Therefore, by the inequality (4),

$$\begin{aligned}
\sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star) &\geq \sum_{j=0}^{\psi} 2^j \, \frac{\mathsf{IUB}\big(I_j, (1-\epsilon)^{\beta_j^\star}\big) - 1}{2} \\
&\geq \frac{1}{2}\left(C(\mathcal{K}^\star) - \sum_{j=0}^{\psi} 2^j \right) \\
&\geq \frac{(C(\mathcal{K}^\star)}{2} - c_{d+1} \quad \Rightarrow \\
C(\mathcal{K}^\star) &\leq 2\sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star) + 2c_{d+1}
\end{aligned}$$

However, the problem of computing an optimal collection $(\lambda_j(i_j^\star), \beta_j(i_j^\star)) \in \hat{\Gamma}_j$, $j = 0, \ldots, \psi$, corresponds to the following generalization of Knapsack: Given $\psi + 1$ sets $\hat{\Gamma}_j$, each containing items of profit $\hat{\lambda}_j(i) = 2^j \lambda_j(i)$ and size $\beta_j(i)$, $(0,0) \in \hat{\Gamma}_j$, select exactly one item i_j from each $\hat{\Gamma}_j$ so as to maximize $\sum_{j=0}^{\psi} \hat{\lambda}_j(i_j)$, subject to $\sum_{j=0}^{\psi} \beta_j(i_j) \leq 1$. Obviously, this problem is $\mathcal{NP}$-complete, because it is a generalization of Knapsack. It is not hard to verify that the FPTAS for ordinary Knapsack (e.g., see Section 9.3 of [5], or [7, 8]) can be generalized to an FPTAS for Generalized Knapsack.

Hence, since the solution returned by CCP has objective value no less than both c_{d+1} and $\frac{1}{1+\delta} \sum_{j=0}^{\psi} 2^j \lambda_j(i_j^\star)$, for any constant $\delta > 0$, CCP is a $(4+\delta)$-approximation algorithm for instances of Fault-Tolerant Partition with channel capacities equal to integer powers of 2. Furthermore, for any constant $\delta > 0$, CCP is a $(8+\delta)$-approximation algorithm for Fault-Tolerant Partition of arbitrary capacity channels.

Complexity: The steps (1)-(3) can be implemented in time polynomial in the size of the input (see also Section 4.2.1). Moreover, the step (4) can be implemented in time polynomial in the size of the input and $\frac{1}{\delta}$, for any $\delta > 0$.

The CCP algorithm uses the pairs $(\nu_j(i), \beta_j(i)) \in \Gamma_j$ to decide an appropriate collection of instances $\big(I_j, (1-\epsilon)^{\beta_j(i)}\big)$ for which the performance of NFD must be computed. Therefore, a polynomial number of bits suffices for storing the values $(1-\epsilon)^{\beta_j(i)}$, since NFD needs only to compare the value of the fault-tolerance constraint with rational numbers.

Let $\mathcal{K}^j(i)$ be the partition that NFD produces on input $\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$. Then, since we can approximate Generalized Knapsack within any constant $\delta > 0$ in time polynomial in $\frac{1}{\delta}$, we can use approximate values for $\hat{\beta}_j(i) \approx \log_{(1-\epsilon)} \Pr[\mathcal{K}^j(i)]$ without significantly decreasing the overall performance of CCP. We can also use precise values by replacing $\beta_j(i)$'s with the corresponding rational numbers $\Pr[\mathcal{K}^j(i)]$, which can be stored using at most $\sum_{\kappa \in I_j} \log(1/f_\kappa)$ bits. Then, the algorithm for Generalized Knapsack must store the rational numbers $\prod \Pr[\mathcal{K}^j(i)]$, and check whether these products are at least $(1-\epsilon)$. □

4.2.1 Implementation of the CCP Algorithm: Steps (2) and (3)

In this section, we show that the sets Γ_j and $\hat{\Gamma}_j$ computed by the algorithm described in Figure 5 fulfill the properties required by the algorithm CCP.

For each iteration, the variable l represents the number of single item groups consisting of the over-reliable items for the current value $(1-\epsilon)^{\beta_j(i)}$ of the fault-tolerance constraint. Additionally, the variable $x_j(i)$ represents the number of the remaining groups computed by evenly distributing the failure probability F_l in order to fulfill the fault-tolerance constraint $\frac{(1-\epsilon)^{\beta_j(i)}}{P_l}$. Clearly, the value of l either remains the same or decreases as the value of $(1-\epsilon)^{\beta_j(i)}$ increases towards to 1.

By construction, $1 \geq \beta_j(i) \geq 0$. We show that, for all $i \geq 1$, $\nu_j(i) \geq \nu_j(i-1)-1$. The steps (2.3) - (2.5) correspond to the case that the value of l remains the same for the new value of $\beta_j(i)$. If l does not decrease, then the initial estimate $l + x_j(i)$ equals the corresponding upper bound $\mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$. Moreover, if $\nu_j(i-1)$ is integer, then $\nu_j(i) = \nu_j(i-1)-1$. Otherwise, $\nu_j(i) = \lfloor \nu_j(i-1) \rfloor \geq \nu_j(i-1) - 1$.

In case that the value of l decreases as a result of decreasing $\beta_j(i)$, the failure probabilities of some of the l single item groups should be greater than the failure probabilities of the remaining $x_j(i)$ groups. Therefore, a partition into at least $l + x_j(i)$ groups can be obtained by placing the items belonging to some of the less reliable single item groups together with the remaining items $f_{l+1}, \ldots, f_{n_j}$, where $n_j = |I_j|$. Hence, the corresponding upper bound $\nu_j(i) = \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$ cannot be less than the initial estimate $l + x_j(i)$, that equals either $\lfloor \nu_j(i-1) \rfloor$ or $\nu_j(i-1)-1$. Then, we compute the right index so as the $\nu_j(i)$ values to form a non-increasing sequence.

From the discussion above, it also becomes clear that Γ_j contains pairs $(\mu, \beta_j(i))$ for all integers $\mu = 0, 1, \ldots, \lfloor \mathsf{IUB}(I_j, (1-\epsilon)) \rfloor$. Also, since for each value $\beta_j(i)$, $\nu_j(i) = \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$, the analysis of NFD implies that $\lambda_j(i) \geq \frac{\lfloor \nu_j(i) \rfloor}{2}$.

Complexity: At any point in time, neither l nor $x_j(i)$ can be more than n_j. Also, in each iteration, except for the iteration that follows a decrease of l, either l or $x_j(i)$ decrease by 1.

(1) Compute $\mathsf{IUB}(I_j, 1-\epsilon) = l^j + x_l^j$.
Set $\beta_j(0) = 1$, $\nu_j(0) = \mathsf{IUB}(I_j, 1-\epsilon)$, $x_j(0) = x_l^j$, $l = l^j$, $i = 0$, $n_j = |I_j|$.
Set $\lambda_j(0) = \mathsf{NFD}(I_j, 1-\epsilon)$.

(2) while $\nu_j(i) > 1$ do

(2.1) $i = i + 1$
If $x_j(i-1)$ is integer, then $x_j(i) = x_j(i-1) - 1$.
Otherwise, $x_j(i) = \lfloor x_j(i-1) \rfloor$.

(2.2) Compute $\beta_j(i)$ from the following equations

$$F_l = \prod_{\kappa=l+1}^{n_j} f_\kappa \ , \ \ P_l = \prod_{\kappa=1}^{l} (1 - f_\kappa) \ , \ \ \left(1 - F_l^{1/x_j(i)}\right)^{x_j(i)} = \frac{(1-\epsilon)^{\beta_j(i)}}{P_l}$$

(2.3) If $l = 0$, then $\nu_j(i) = x_j(i)$, $\lambda_j(i) = \mathsf{NFD}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$, and go to (2).

(2.4) Compute $\mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big) = l^j + x_l^j$.

(2.5) If $l = l^j$, then $\nu_j(i) = l + x_j(i)$, $\lambda_j(i) = \mathsf{NFD}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$, and go to (2).

(2.6) If $l > l^j$, then $\nu_j(i) = \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$, $x_j(i) = x_l^j$, $l = l^j$, $\lambda_j(i) = \mathsf{NFD}\Big(I_j, (1-\epsilon)^{\beta_j(i)}\Big)$.

(2.7) If $\nu_j(i) < \nu_j(i-1)$ then go to (2).

(2.8) Otherwise, find the smallest index $i \geq t \geq 0$, such that $\nu_j(i) > \nu_j(t)$ and $\nu_j(t-1) \geq \nu_j(i)$. Then, set $\nu_j(t) = \nu_j(i)$, $x_j(t) = x_j(i)$, $\beta_j(t) = \beta_j(i)$, $\lambda_j(t) = \lambda_j(i)$, $i = t$, and go to (2).

(3) Compute $\beta_j(i+1) = \log_{(1-\epsilon)}\Big(1 - \prod_{\kappa=1}^{n_j} f_\kappa\Big)$.

(3.1) If $\beta_j(i+1) < 1$, then set $\nu_j(i+1) = 1$, $\lambda_j(i+1) = 1$, $\nu_j(i+2) = 0$, $\lambda_j(i+2) = 0$, $\beta_j(i+2) = 0$.

(3.2) Otherwise, set $\nu_j(i+1) = 0$, $\lambda_j(i+1) = 0$, $\beta_j(i+1) = 0$.

Fig. 5. An implementation of the steps (2) and (3) of CCP.

Hence, the algorithm terminates after at most n_j^2 iterations. Moreover, the analysis of NFD implies that each iteration can be performed in polynomial time.

Additionally, notice that we do not need to calculate the values of $\beta_j(i)$, since we only use the corresponding values $(1-\epsilon)^{\beta_j(i)}$ of the fault-tolerance constraint. Since the algorithm CCP only uses the $\lambda_j(i)$ values computed by NFD for appropriately selected values of the fault-tolerance constraint, it is sufficient to store the first $\sum_{\kappa=1}^{n_j} \log(1/f_\kappa)$ bits of each $(1-\epsilon)^{\beta_j(i)}$.

$\square$

4.2.2 The Proof of Lemma 4.3

Lemma 4.3 states that there exist real numbers $\beta_j^\star \geq 0$ for the capacity classes I_j, $\sum_{j=0}^{\psi} \beta_j^\star \leq 1$, such that the total effective capacity $C(\mathcal{K}^\star)$ of an optimal partition $\mathcal{K}^\star$ can be bounded from above by $\sum_{j=0}^{\psi} 2^j \,\mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j^\star}\Big)$, where all the channels of the class I_j are assumed to have capacity equal to 2^j. The proof actually shows how to calculate an appropriate set of real numbers from the most reliable, optimal partition $\mathcal{K}^\star$.

Proof. Let $\mathcal{K}^\star = \{K_1^\star, \ldots, K_\nu^\star\}$ be the most reliable, optimal, $(1-\epsilon)$-fault-tolerant partition of K, and let $C(\mathcal{K}^\star) = \sum_{i=1}^{\nu} C(K_i^\star) = \sum_{i=1}^{\nu} \min_{\kappa \in K_i^\star}\{c_\kappa\}$ be the effective capacity of $\mathcal{K}^\star$. Given such an optimal solution $\mathcal{K}^\star$, we show how to calculate an appropriate set of $\psi + 1$ real numbers β_j.

In order to calculate β_j's, we examine the contribution of each class I_j to the groups $K_i^\star$. For each group $K_i^\star$, we calculate the contribution d_j^i of the I_j items to $F^i = \prod_{\kappa \in K_i^\star} f_\kappa = 1 - \Pr[K_i^\star]$. Then, we calculate the contribution β_j^i of the I_j items to the portion $\beta^i = \log_{(1-\epsilon)} \Pr[K_i^\star]$ of the fault-tolerance constraint that has been devoted to $K_i^\star$.

Due to technical reasons, we have to distinguish two cases. For the groups $K_i^\star$ entirely consisting of items belonging to the class I_j, the contributions d_j^i and β_j^i are accumulated to $d_j^{(1)}$ and $\beta_j^{(1)}$ respectively. Both $d_j^{(1)}$ and $\beta_j^{(1)}$ are initially equal to 0. For the groups $K_i^\star$ not entirely consisting of items belonging to a single class I_j, the contributions d_j^i and β_j^i are accumulated to $d_j^{(2)}$ and $\beta_j^{(2)}$ respectively. Also, both $d_j^{(2)}$ and $\beta_j^{(2)}$ are initially equal to 0. For each class I_j, the real number β_j is the sum of $\beta_j^{(1)}$ and $\beta_j^{(2)}$.

For each group $K_i^\star$ entirely consisting of items belonging to the class I_j, we increase $d_j^{(1)}$ by $d^i = d_j^i = 1$, and $\beta_j^{(1)}$ by $\beta^i = \beta_j^i = \log_{(1-\epsilon)} \Pr[K_i^\star]$. Clearly, the contribution of each group $K_i^\star$ to the objective value of $\mathcal{K}^\star$ is exactly 2^j. Let I_j^1 contain all the items of I_j belonging to some group $K_i^\star \subseteq I_j$. The quantity $d_j^{(1)}$ cannot be more than the optimal number of groups ν_{j_1} obtained from I_j^1 with fault-tolerance constraint $(1-\epsilon)^{\beta_j^{(1)}}$, because the groups $K_i^\star$, $K_i^\star \subseteq I_j$, form a $\Big((1-\epsilon)^{\beta_j^{(1)}}\Big)$-fault-tolerant partition of I_j^1. Moreover, by the analysis of NFD,

$$d_j^{(1)} \leq \nu_{j_1} \leq \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j^{(1)}}\Big) \tag{5}$$

Let $Z_j \subseteq \mathcal{K}^\star$ contain all the groups $K_i^\star$ such that $K_i^\star \cap I_j \neq \emptyset$, and $K_i^\star$ does not entirely consist of I_j items. For each $K_i^\star \in Z_j$, let $F^i = 1 - \Pr[K_i^\star]$ and $F_j^i = \prod_{\kappa \in K_i^\star \cap I_j} f_\kappa$. Then, for each $K_i^\star \in Z_j$, we increase $d_j^{(2)}$ by $d_j^i = \log_{F^i} F_j^i$ and $\beta_j^{(2)}$ by $\beta_j^i = d_j^i(\log_{(1-\epsilon)} \Pr[K_i^\star]) = d_j^i \beta^i$. Obviously, the contribution of each $K_i^\star$ to $C(\mathcal{K}^\star)$ is bounded by

$$C(K_i^\star) \leq \sum_{j : I_j \cap K_i^\star \neq \emptyset} 2^j d_j^i \ .$$

Let I_j^2 contain all the items of I_j belonging to the groups $K_i^\star \in Z_j$. Then, we show that $d_j^{(2)}$ cannot be more than the upper bound $\mathsf{IUB}\left(I_j^2, (1-\epsilon)^{\beta_j^{(2)}}\right)$ on the optimal number of groups obtained from I_j^2 with fault-tolerance constraint $(1-\epsilon)^{\beta_j^{(2)}}$.

Assume that, for some class I_j, $d_j^{(2)} > \mathsf{IUB}\left(I_j^2, (1-\epsilon)^{\beta_j^{(2)}}\right)$, and let $F_{j2} = \prod_{\kappa \in I_j^2} f_\kappa = \prod_{K_i^\star \in Z_j}\left((F^i)^{d_j^i}\right)$. It should be clear that the number $x \geq 0$ defined by the equation

$$\left(1 - F_{j2}^{1/x}\right)^x = (1-\epsilon)^{\beta_j^{(2)}}$$

cannot be less than $d_j^{(2)}$, because of the following inequality (see also Proposition 4.4):

$$\left[1 - \left(\prod_{K_i^\star \in Z_j}\left((F^i)^{d_j^i}\right)\right)^{1/d_j^{(2)}}\right]^{d_j^{(2)}} \geq \prod_{K_i^\star \in Z_j}\left(1 - F^i\right)^{d_j^i} = (1-\epsilon)^{\sum_{K_i^\star \in Z_j} d_j^i \beta^i} = (1-\epsilon)^{\beta_j^{(2)}} \qquad (6)$$

Therefore, it must be the case that $d_j^{(2)} > l^{j_2} > 0$, where l^{j_2} is the number of over-reliable items included in the instance $\left(I_j^2, (1-\epsilon)^{\beta_j^{(2)}}\right)$. Hence, by the analysis of NFD, there must exist a group $K_{z_1}^\star \in Z_j$ and item $f_{z_1} \in K_{z_1}^\star \cap I_j^2$, such that $f_{z_1} < F_{j2}^{1/x}$. This implies that

$$1 - f_{z_1} > 1 - F_{j2}^{1/x} = (1-\epsilon)^{\beta_j^{(2)}/x} \geq (1-\epsilon)^{\beta_j^{(2)}/d_j^{(2)}} \quad ,$$

because $x \geq d_j^{(2)}$. Additionally, since $d_j^{(2)} = \sum_{K_i^\star \in Z_j} d_j^i$ and $\beta_j^{(2)} = \sum_{K_i^\star \in Z_j} d_j^i \beta^i$, where $\beta^i = \log_{(1-\epsilon)} \Pr[K_i^\star]$, it must exist another group $K_{z_2}^\star \in Z_j$ being allocated less than the average portion of the fault-tolerance constraint. Hence,

$$\Pr[K_{z_2}^\star] = (1-\epsilon)^{\beta^{z_2}} < (1-\epsilon)^{\beta_j^{(2)}/d_j^{(2)}} \quad .$$

Then, we consider the partition $\mathcal{K}'$ obtained from $\mathcal{K}^\star$ by replacing the group $K_{z_1}^\star$ by $K'_{z_1} = \{f_{z_1}\}$, and the group $K_{z_2}^\star$ by $K'_{z_2} = K_{z_2}^\star \cup \left(K_{z_1}^\star - \{f_{z_1}\}\right)$. Since $1 - f_{z_1} > \Pr[K_{z_2}^\star]$, by an argument similar to one used in the analysis of NFD, $\mathcal{K}'$ is strictly more reliable than $\mathcal{K}^\star$. Moreover, it is not hard to verify that $C(K'_{z_1}) + C(K'_{z_2}) \geq C(K_{z_1}^\star) + C(K_{z_2}^\star)$. This is a contradiction to the selection of $\mathcal{K}^\star$ as the most reliable, optimal, $(1-\epsilon)$-fault-tolerant partition of K. Therefore, there always exists an optimal solution that, for all capacity classes I_j, satisfies the following inequality:

$$d_j^{(2)} \leq \mathsf{IUB}\left(I_j^2, (1-\epsilon)^{\beta_j^{(2)}}\right) \qquad (7)$$

Obviously, since $I_j^1 \cap I_j^2 = \emptyset$ and $I_j^1 \cup I_j^2 \subseteq I_j$, Equations (5) and (7) imply that $d_j = d_j^{(1)} + d_j^{(2)} \leq \mathsf{IUB}\left(I_j, (1-\epsilon)^{\beta_j}\right)$. Therefore, since, for all $j = 0, \ldots, \psi$, $\sum_{i=1}^{\nu} d_j^i = d_j^{(1)} + d_j^{(2)}$, the following inequalities hold for the real numbers β_j:

$$\sum_{i=1}^{\nu} C(K_i^\star) \leq \sum_{i=1}^{\nu}\sum_{j=0}^{\psi} d_j^i 2^j = \sum_{j=0}^{\psi} 2^j\left(d_j^{(1)} + d_j^{(2)}\right) \leq \sum_{j=0}^{\psi} 2^j \; \mathsf{IUB}\left(I_j, (1-\epsilon)^{\beta_j}\right) \quad .$$

By the definition of β_j, it is straight-forward that $\beta_j \geq 0$, and, since $\Pr[\mathcal{K}^\star] \geq 1-\epsilon$,

$$\sum_{i=1}^{\nu} \beta^i = \sum_{i=1}^{\nu}\sum_{j=0}^{\psi} \beta_j^i = \sum_{j=0}^{\psi} \beta_j \leq 1 \ .$$

□

In the proof of Lemma 4.3, Equation (6), we have used the following inequality that holds because the function $g(y) = \ln(1-e^y)$ is concave for all $y < 0$.

Proposition 4.4. *For any F_i, $1 > F_i > 0$, and d_i, $d_i > 0$, $i = 1, \ldots, n$, the following inequality holds:*

$$\left[1 - \left(\prod_{i=1}^{n} F_i^{d_i}\right)^{1/D}\right]^D \geq \prod_{i=1}^{n}(1-F_i)^{d_i} \ , \tag{8}$$

where $D = \sum_{i=1}^{n} d_i$.

5 Assignments on Identical Capacity Channels

In this section, we present NFD-LS, that is polynomial-time 4-approximation algorithm for Minimum Fault-Tolerant Congestion on identical capacity channels. Given a set K of faulty, parallel, identical capacity channels, a set M of messages to be delivered over K, and a fault-tolerance constraint $(1-\epsilon)$, the NFD-LS algorithm works as follows:

1. It invokes Next Fit Decreasing – NFD on instance $(K, 1-\epsilon)$ to compute a $(1-\epsilon)$-fault-tolerant partition $\mathcal{K}$ into λ disjoint groups, $\mathcal{K} = \{K_1, \ldots, K_\lambda\}$.

2. It invokes List Scheduling – LS (e.g. see Section 1.1 of [5]) to compute a non-redundant schedule ϕ' of the message set M on λ reliable, identical capacity channels.

3. For all messages $j \in M$, if $\phi'(j) = l$, for some integer $1 \leq l \leq \lambda$, NFD-LS sets $\phi(j) = K_l$. The algorithm returns the redundant assignment ϕ.

It should be clear that the reliability of the resulting assignment ϕ equals the reliability of the underlying partition $\mathcal{K}$ produced by NFD, $\Pr[\phi] = \Pr[\mathcal{K}] \geq 1-\epsilon$. Additionally, since all the channels are of identical capacity, the congestion of ϕ equals the makespan of the non-redundant assignment ϕ' produced by List Scheduling, $\mathsf{Cong}(\phi) = \mathsf{Makespan}(\phi')$.

The analysis of NFD-LS is based on the following technical lemma, whose proof is deferred to Section 5.1.1. This lemma states that the optimal Fault-Tolerant Congestion $\mathsf{Cong}^\star$ cannot be less that the total size S_{tot} of the messages divided by the ceiling of $\mathsf{IUB}(K, 1-\epsilon)$, that bounds from above the optimal number of groups produced from K with fault-tolerance constraint $(1-\epsilon)$.

Lemma 5.1. *Given a set $K = \{f_1, \ldots, f_n\}$ of identical capacity channels, m unit size messages, and a fault-tolerance constraint $(1-\epsilon)$, the optimal Fault-Tolerant Congestion* $\mathsf{Cong}^\star$ *cannot be less than $m \cdot \lceil \mathsf{IUB}(K, 1-\epsilon) \rceil^{-1}$.*

Based on Lemma 5.1 and the analyses of Next Fit Decreasing and List Scheduling, we can show that the partition assignment ϕ produced by NFD-LS approximates Minimum Fault-Tolerant Congestion within a factor of 4.

Theorem 5.2. *The redundant assignment ϕ produced in polynomial-time by* NFD-LS *approximates within a factor of 4 Minimum Fault-Tolerant Congestion on identical capacity channels.*

Proof. Given an instance $(K, M, 1-\epsilon)$ of Minimum Fault-Tolerant Congestion, let $\nu + 1 = \lceil \mathsf{IUB}(K, 1-\epsilon) \rceil$. Obviously, Lemma 5.1 implies that the optimal Fault-Tolerant Congestion $\mathsf{Cong}^\star$ cannot be less than $\max\left\{\frac{S_{\text{tot}}}{\nu+1}, s_{\max}\right\}$.

If $\nu = 1$, then the NFD algorithm on instance $(K, 1-\epsilon)$ produces at least 1 reliable effective channel, and, since List Scheduling is optimal for a single channel, we obtain that

$$\mathsf{Cong}(\phi) \leq 2\mathsf{Cong}^\star \ .$$

For $\nu \geq 2$, the analysis of NFD implies that $\mathsf{NFD}(K, 1-\epsilon) \geq \frac{\nu}{2}$, and, by the analysis of List Scheduling, we obtain that

$$\mathsf{Cong}(\phi) \leq \frac{2S_{\text{tot}}}{\nu} + s_{\max} \leq \frac{3S_{\text{tot}}}{\nu+1} + s_{\max} \leq 4\mathsf{Cong}^\star$$

□

5.1 A Lower Bound on Minimum Fault-Tolerant Congestion

The proof of Lemma 5.1, that bounds from below Minimum Fault-Tolerant Congestion on identical capacity channels, is based on the following combinatorial lemma, that applies to identical channels, that are channels of both identical capacity and identical failure probability. This lemma states that the reliability of the most reliable ν-partition assignment ϕ_ν bounds from above the reliability of any redundant assignment ϕ, $\mathsf{Cong}(\phi) \leq \Lambda$, of $\Lambda\nu$ unit size messages to at least $\nu\mu$ identical channels.

Lemma 5.3. *Given an integer $\nu \geq 1$, for any integer $\Lambda \geq 1$, let ϕ be any redundant assignment of $\Lambda\nu$ unit size messages to a set K of identical channels, each of failure probability f. If each message is assigned to exactly μ channels, $|K| \geq \nu\mu$, and* $\mathsf{Cong}(\phi) \leq \Lambda$*, then* $\Pr[\phi] \leq (1-f^\mu)^\nu = \Pr[\phi_\nu]$.

Proof. Clearly, the most reliable ν-partition assignment ϕ_ν has $\Pr[\phi_\nu] = (1 - f^\mu)^\nu$. Then, we show that ϕ_ν is at least as reliable as any other assignment ϕ that fulfills the hypothesis. Recall that $\mathsf{MF}(\phi)$ denotes the cardinality of a minimum feasible set for ϕ. Initially, we assume that $\mathsf{MF}(\phi) = \nu$, i.e. for the assignment ϕ, at least ν channels must be active in order at least one copy of all the messages to be delivered to the destination.

Next, we adopt a time-evolving representation of random channel subsets. Similar time-evolving representations of random edge subsets have been used in [1, 9] for proving lower bounds on graph reliability.

We introduce non-negative time t, and we define a time-dependent random subset $K(t) \subseteq K$ representing the set of active channels by the time t. Initially, $K(0) = \emptyset$. Each channel is given an arrival time chosen independently from the exponential distribution with mean 1, and each time a channel arrives, it is included in $K(t)$. In particular, the time t_i at which the channel i is attached to $K(t)$, has a probability distribution function $1 - e^{-t}$, and the times t_i, $i \in K$, are independent.

Each time a channel i is included in $K(t)$, all the copies of the messages assigned to i are removed from the set of the remaining channels $K - K(t)$, because, since the set $K(t)$ represents the set of active channels, the delivery of these messages is assured by the channel i. Then, $\phi(t)$ denotes the assignment at time t, that is obtained from ϕ by removing all the copies of the messages assigned to $K(t)$ by ϕ. Also, all the channels of $K - K(t)$ assigned zero load by $\phi(t)$ can be included in $K(t)$, because they do not further affect the stochastic process due to the exponential distribution of the channel arrival times. Then, $n(t) = |K - K(t)|$ denotes the number of channels whose arrival can alter $\phi(t)$, that are all the channels assigned non-zero load by $\phi(t)$.

The random variable $T_r(\phi)$, $r = 1, \ldots, \nu$, is defined as the time t at which the arrival of a channel causes $\phi(t)$ to become an assignment of $\mathsf{MF}(\phi(t)) = r - 1$ for the very first time. We will show that $T_1(\phi)$ stochastically dominates $T_1(\phi_\nu)$, that is

$$\forall t \geq 0, \quad \Pr[T_1(\phi) \geq t] \geq \Pr[T_1(\phi_\nu) \geq t] \ .$$

Clearly, this implies the lemma, because $\Pr[T_1(\phi) \leq t]$ is the probability that ϕ has become feasible by the time t.

Additionally, let $T_{\nu+1}(\phi) = 0$, and let the random variables $\theta_r(\phi) = T_r(\phi) - T_{r+1}(\phi)$, $r = 1, \ldots, \nu$, denote the length of the time interval for which $\mathsf{MF}(\phi(t))$ is equal to r. Obviously, $\theta_r(\phi)$ is the time it takes for decreasing the cardinality of a minimum feasible set for $\phi(t)$ from r to $r-1$. Moreover, since a channel arrival can decrease $\mathsf{MF}(\phi(t))$ by at most one, $T_1(\phi) = \sum_{r=1}^{\nu} \theta_r(\phi)$.

Due to the memoryless property of the exponential distribution, the time it takes for the next channel arrival is independent of the previous channel arrivals and the time t. Therefore, once we have conditioned on the value of $K(t)$, the time for the next arrival has probability distribution function $1-e^{-n(t)t}$. In particular, it is shown in [9] that, since the time of arrival has no impact on which of the channels of $K-K(t)$ is the first to arrive, the time for the next arrival has the right exponential distribution, regardless of the values of $K(t)$ we condition on.

As Lomonosov observed [1], we can imagine that $K(t) = K_\sigma$ corresponds to a state σ of a discrete Markov chain on 2^K. From σ, we can only visit the states ρ corresponding to $K_\rho = K_\sigma \cup \{i\}$, for some $i \in K-K_\sigma$, with transition probabilities $1/n_\sigma$, $n_\sigma = |K-K_\sigma| = n(t)$. Then, we can first chose a path ω of this Markov chain with transition probabilities $1/n_\sigma$ out of each state σ, and then traverse this path with a sojourn time in each σ exponentially distributed with mean $1/n_\sigma$. Hence, the probability distribution function $D_\omega(t)$ of the time for traversing the path ω can be obtained by the convolution of exponential distributions with mean values $1/n_{\sigma_0} < 1/n_{\sigma_1} < \cdots < 1/n_{\sigma_{l-1}}$, where l is the length of the path. In particular,

$$D_\omega(t) = \Pr[\theta_\omega \le t] = \sum_{j=0}^{l-1}\left(1-e^{-n_{\sigma_j}t}\right) \prod_{0\le i\le l-1, i\ne j} \frac{n_{\sigma_i}}{n_{\sigma_i}-n_{\sigma_j}}$$

Therefore, the probability distribution function of each random variable $\theta_r(\phi)$ is

$$D_r(\phi, t) = \sum_\omega p_\omega D_\omega(t) \ ,$$

where ω is any path/sequence of channel arrivals that causes $\mathsf{MF}(\phi(t))$ to decrease from r to $r-1$, and p_ω denotes the probability of choosing this path. Additionally, notice that the random variables $\theta_r(\phi)$, $r = 1, \ldots, \nu$ are mutually independent, because they are weighted sums of disjoint sets of mutually independent random variables. Additionally, it is well known that, if X_1, X_2, Y_1, Y_2 are independent random variables so that X_1 dominates Y_1 and X_2 dominates Y_2, then $X_1 + X_2$ also dominates $Y_1 + Y_2$. Therefore, in order to prove the lemma, we have to show that, for all $r = 1, \ldots, \nu$, $\theta_r(\phi)$ dominates $\theta_r(\phi_\nu)$.

Notice that, at some time t, the stochastic process may be in a state σ, such that $\mathsf{MF}(\sigma) = r$ (i.e. the minimum feasible set cardinality of the assignment $\phi(t)$ corresponding to σ is equal to r) and $n(t) = n_\sigma > r\mu$. The stochastic process is said to be in a *good state* ρ, if $\mathsf{MF}(\rho) = r$ and the arrival of more than $r\mu$ channels leads the process to a state ρ' that is either good or $\mathsf{MF}(\rho') = r-1$. Wlog. we can assume that a transition to a good state ρ of $\mathsf{MF}(\rho) = r$ is equivalent to an immediate transition to a state ρ' of $\mathsf{MF}(\rho') = r-1$. This eliminates the need to consider a transition from a good state ρ to another good state ρ', such that $\mathsf{MF}(\rho) = \mathsf{MF}(\rho')$. The stochastic process is said to be in a *bad state* σ, if $\mathsf{MF}(\sigma) = r$ and the arrival of at most $r\mu$ channels leads the process to a state ρ' that is either good or $\mathsf{MF}(\rho') = r-1$.

Let σ be any state reached at time t for which $\mathsf{MF}(\phi(t))$ has become equal to r for the very first time, and let $n_\sigma > r\mu$. Notice that there always exists a path $\omega = (\sigma, \sigma_1, \ldots, \sigma_x)$, such that $\mathsf{MF}(\sigma) = \mathsf{MF}(\sigma_1) = \cdots = \mathsf{MF}(\sigma_x) = r$ and $n_{\sigma_x} \le r\mu$. The path ω corresponds to the arrival of a channel set $K_{\rm bad} \subseteq K - K(t)$, $|K_{\rm bad}| \ge n_\sigma - r\mu$, that does not change $\mathsf{MF}(\phi(t))$, but it reduces the cardinality of $n(t)$ to $r\mu$ or less. Moreover, the removal of $K_{\rm bad}$ from $\phi(t)$ causes the state σ_x to be reached, regardless of the order the channels of $K_{\rm bad}$ actually arrive. Since σ_x is a bad state, all the states $\sigma, \sigma_1, \ldots, \sigma_{x-1}$ are also bad ones. Therefore, for all $r = 1, \ldots, \nu$, all the states for which $\mathsf{MF}(\phi(t))$ becomes equal to r for the very first time are bad ones.

Let r, $1 \le r \le \nu$, be any integer, and assume that some channel arrival has just caused $\phi(t)$ to be an assignment of $\mathsf{MF}(\phi(t)) = r$ for the first time. Hence, the corresponding state σ is a bad one. Then, we show that the random variable $\theta_r(\phi)$ dominates an exponentially distributed random variable with mean $1/r\mu$. One possibility for σ to be a bad state is $n(t) = n_\sigma \le r\mu$. Then, the sojourn time in σ is exponentially distributed with mean at least $1/r\mu$, and therefore $\theta_r(\phi)$ dominates an exponentially distributed random variable with mean $1/r\mu$.

Next we consider the case that $n_\sigma > r\mu$. Since σ is a bad state, with probability at most $\frac{r\mu}{n_\sigma}$, the stochastic process moves into a state ρ of $\mathsf{MF}(\rho) = r-1$, while with probability at least $1 - \frac{r\mu}{n_\sigma}$, it enters some other bad state σ'. (Recall that a move into a good state $\hat{\rho}$ of $\mathsf{MF}(\hat{\rho}) = r$ is also considered as a move into a state ρ of $\mathsf{MF}(\rho) = r-1$). Additionally, regardless of the new state, the sojourn time in σ is exponentially distributed with mean $1/n_\sigma$.

In case that the new state is a bad one, the aforementioned situation goes on until $n_{\sigma'} \le r\mu$, when wlog. we can assume that $\mathsf{MF}(\phi(t))$ becomes $r-1$ with probability 1. Therefore, we can inductively assume that the random variable $\theta_r(\sigma')$, which denotes the time it takes for getting from the bad state σ' to a state ρ' of $\mathsf{MF}(\rho') = r-1$, dominates an exponentially distributed random variable with mean $1/r\mu$. Since

$$\frac{n_\sigma - r\mu}{n_\sigma}\left(1 + \frac{r\mu}{n_\sigma - r\mu}e^{-n_\sigma t} - \frac{n_\sigma}{n_\sigma - r\mu}e^{-r\mu t}\right) + \frac{r\mu}{n_\sigma}\left(1 - e^{-n_\sigma t}\right) = 1 - e^{-r\mu t} \quad ,$$

$\theta_r(\sigma)$ also dominates an exponentially distributed random variable with mean $1/r\mu$. Therefore, the random variable $\theta_r(\phi)$ dominates the random variable $\theta_r(\phi_\nu)$, which is exponentially distributed with mean $1/r\mu$.

Finally, let ϕ' be any redundant assignment of $\Lambda\nu$ unit size messages, such that ϕ' assigns all the messages to exactly μ channels, $\mathsf{Cong}(\phi') \le \Lambda$, and $\mathsf{MF}(\phi') > \nu$. It should be clear that ϕ' cannot be more reliable than ϕ, because the stochastic process corresponding to ϕ' must also pass through a bad state σ of $\mathsf{MF}(\sigma) = \nu$. □

Remark. Lemma 5.3 also applies to the probability that isolated nodes do not appear to a not necessarily connected hypergraph with $\nu\mu$ "faulty" hyperedges of cardinality Λ. In particular,

3	3	3	4	4	4
1	1	1	2	2	2
a	b	c	d	e	f

2	3	4	3	4	4
1	1	1	2	2	3
a	b	c	d	e	f

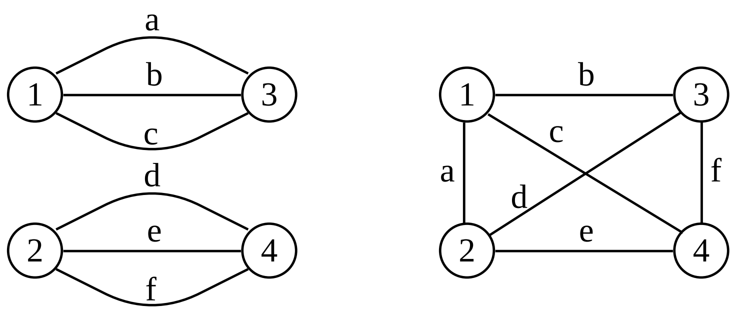

Fig. 6. Two different assignments of 4 unit size messages to 6 identical channels ($\Lambda = 2$).

assume that $|K| = \nu\mu$, and ϕ assigns each message to exactly μ channels, and the load of each channel is exactly Λ. Given such an assignment ϕ, we can construct a hypergraph $H_\phi(N, E)$, where N consists of $\Lambda\nu$ nodes corresponding to the unit size messages, and E consists of $\nu\mu$ hyperedges corresponding to the channels of K. Moreover, each $e \in E$ consists of the messages assigned to the corresponding channel of K by ϕ. Therefore, the cardinality of each $e \in E$ is exactly Λ. Moreover, for any node $v \in N$, let $\deg_{H_\phi}(v) = |\{e \in E(H_\phi) : v \in e\}|$.

Clearly, an assignment ϕ is feasible for a $K' \subseteq K$, iff the removal of the hyperedges corresponding to $K - K'$ does not create any isolated nodes in H_ϕ, that are nodes of degree 0. Lemma 5.3 implies that the hypergraph Π_ν corresponding to the most reliable ν-partition assignment ϕ_ν achieves the greatest probability of not having any isolated points under random and independent edge faults.

Figure 6 depicts two different assignments of 4 unit size messages, $M = \{1, 2, 3, 4\}$, to 6 identical channels, $K = \{a, b, c, d, e, f\}$, for $\nu = 2$ and $\Lambda = 2$, and the corresponding graphs. If all the channels fail with probability f, $1 > f > 0$, then the failure probability of the most reliable 2-partition is $2f^3 - f^6$, while the failure probability of the "clique" assignment ϕ is $4f^3 - 6f^5 + 3f^6$. It is straight-forward to verify that $4f^3 - 6f^5 + 3f^6 > 2f^3 - f^6$, for all $1 > f > 0$. Additionally, the probability distribution function for $T_1(\phi)$ is $D(\phi, t) = 1 - 3e^{-6t} + 6e^{-5t} - 4e^{-3t}$, while the probability distribution function for $T_1(\phi_2)$ is $D(\phi_2, t) = 1 + e^{-6t} - 2e^{-3t}$. □

The following corollary is an immediate consequence of Lemma 5.3 and applies to the more general situation that the total number of copies of all the $\Lambda\nu$ messages is equal to $\Lambda\nu\mu$.

Corollary 5.4. *Given an integer $\nu \geq 1$, for any integer $\Lambda \geq 1$, let ϕ be any redundant assignment of $\Lambda\nu$ unit size messages to a set K of identical channels of failure probability f. If each message j, $j = 1, \ldots, \Lambda\nu$, is assigned to exactly μ_j channels so as $\sum_{j=1}^{\Lambda\nu} \mu_j = \Lambda\nu\mu$, $|K| \geq \nu\mu$, and $\mathsf{Cong}(\phi) \leq \Lambda$, then $\Pr[\phi] \leq (1 - f^{\mu})^{\nu} = \Pr[\phi_\nu]$.*

Proof. It is easy to verify (e.g. by the inclusion-exclusion formula) that there always exists a redundant assignment ϕ' of $\Lambda\nu$ unit size messages to K, such that ϕ' assigns each message to exactly μ channels, $\mathsf{Cong}(\phi') = \mathsf{Cong}(\phi)$, and $\Pr[\phi] \leq \Pr[\phi']$. □

5.1.1 The Proof of Lemma 5.1

Then, we prove Lemma 5.1 stating that the optimal Fault-Tolerant Congestion $\mathsf{Cong}^\star$ cannot be less than the total load divided by the ceiling of $\mathsf{IUB}(K, 1-\epsilon)$. In order to handle channels of different failure probability, we replace each channel by an appropriate number of identical parallel channels (see also [9]). Then, we apply Corollary 5.4 to show that any $(1-\epsilon)$-fault-tolerant redundant assignment cannot use more than $\lceil \mathsf{IUB}(K, 1-\epsilon) \rceil$ effective reliable channels.

Proof. Let ϕ be any $(1-\epsilon)$-fault-tolerant redundant assignment of m unit size messages to K. At first, we assume that the l-component of the upper bound $\mathsf{IUB}(K, 1-\epsilon)$, that corresponds to the number of over-reliable items, is equal to 0, and we set $\nu = \lceil \mathsf{IUB}(K, 1-\epsilon) \rceil = \lceil x \rceil$, where x is defined by

$$\left(1 - F^{1/x}\right)^x = 1 - \epsilon \ , \quad F = \prod_{i=1}^{n} f_i \ .$$

Therefore, at most ν groups can be obtained from the Fault-Tolerant Partition instance $(K, 1-\epsilon)$, each of failure probability at least $F^{1/x}$.

In order to handle different failure probabilities, we choose a sufficiently small real number δ, and replace each channel i of failure probability f_i by a "bundle" of $n_i = \left\lceil \frac{-\ln f_i}{\delta} \right\rceil$ parallel channels each of failure probability $f = 1 - \delta$ (see also [9]). Furthermore, the messages assigned to the channel i are assigned to all n_i parallel channels. Therefore, the bundle of n_i parallel channels contains at least one active channel with probability

$$1 - (1-\delta)^{\left\lceil \frac{-\ln f_i}{\delta} \right\rceil} \ .$$

Since this quantity converges to f_i as $\delta \to 0$, the reliability of the assignment $\hat{\phi}$ obtained from ϕ by applying this transformation tends to $\Pr[\phi]$.

Corollary 5.4 implies that we can only consider an assignment ϕ that assigns each message j to a set of channels K_j of reliability $\Pr[K_j] = 1 - F^{1/x}$. Therefore, $\hat{\phi}$ assigns each message to exactly μ channels, where $\mu = \left\lceil \frac{-\ln F}{x\delta} \right\rceil$. Lemma 5.3 implies that, if $\mathsf{Cong}(\phi) < \frac{m}{\nu}$, then $\Pr[\hat{\phi}] < 1 - \epsilon$.

In case that $l > 0$, we assume that the channels are indexed in non-increasing order of reliability. Then, the failure probability of each of the first l most reliable channels is less than the failure probability F_l^{1/x_l} of the remaining x_l groups, where $F_l = \prod_{i=l+1}^{n} f_i$ (see also the analysis of NFD for the definitions of l and x_l). Therefore, wlog. we can assume that ϕ assigns each message j to either exactly one of the first l most reliable channels, or a subset K_j of $K_{l^+} = K - \bigcup_{i=1}^{l}\{f_i\}$ of reliability $\Pr[K_j] = 1 - F_l^{1/x_l}$. If $\mathsf{Cong}(\phi) < \frac{m}{l+\lceil x_l \rceil} = \frac{m}{\lceil \mathsf{IUB}(K, 1-\epsilon) \rceil}$, then the reliability of the partial assignment to the set K_{l^+} must be strictly less than $\frac{1-\epsilon}{P_l}$, where $P_l = \prod_{i=1}^{l}(1 - f_i)$. □

6 Assignments on Related Capacity Channels

In this section, we present SP-OPT, that is a logarithmic approximation algorithm for Minimum Fault-Tolerant Congestion, in case of unit size messages and related capacity channels. The redundant assignments produced by SP-OPT are based on an optimal, non-redundant schedule of $|M|$ unit size messages on the related capacity, reliable, effective channels obtained by the Safe Partition algorithm. In particular, given an instance $(K, M, 1-\epsilon)$, SP-OPT works as follows:

1. It calls Safe Partition – SP on instance $(K, 1 - \epsilon)$ to obtain a $(1 - \epsilon)$-fault-tolerant partition.

2. If SP returns a single group consisting of the first $d + 1$ channels, where d is the largest index such that $\prod_{i=1}^{d} f_i > \epsilon$, then SP-OPT assigns all the messages to the first $d + 1$ channels, i.e. for all $j \in M$, $\phi(j) = \{(f_1, c_1), \ldots, (f_{d+1}, c_{d+1})\}$, and returns ϕ.

3. If SP returns a partition $\mathcal{K}$ into λ disjoint groups, $\mathcal{K} = \{K_1, \ldots, K_\lambda\}$, then SP-OPT computes an optimal, non-redundant schedule ϕ' of M on λ reliable, related, parallel channels, each of capacity $C(K_l) = \min_{i \in K_l}\{c_i\}$, $l = 1, \ldots, \lambda$. Then, for all messages $j \in M$, if $\phi'(j) = l$ for some integer $1 \le l \le \lambda$, SP-OPT assigns $\phi(j) = K_l$, and returns the assignment ϕ.

Since the partitions computed by Safe Partition are always $(1 - \epsilon)$-fault-tolerant, the reliability of the resulting assignment ϕ cannot be less than $(1 - \epsilon)$. The analysis of SP-OPT is based on the analysis of Safe Partition, and the fact that any $(1 - \epsilon)$-fault-tolerant assignment must have at least one copy of all the messages to some channel of index greater than d.

Theorem 6.1. *The algorithm SP-OPT is a polynomial-time $2\left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$-approximation algorithm for Minimum Fault-Tolerant Congestion, in case of unit size messages and related capacity channels.*

Proof. Since it is straight-forward that the reliability of the resulting assignment ϕ cannot be less than $(1-\epsilon)$, we focus on the justification of the claimed approximation ratio.

By the definition of d as the largest index so that $\prod_{i=1}^{d} f_i > \epsilon$, if all the copies of a message $j \in M$ are only assigned to the first d channels, the probability that j has a copy on some active channel is less than $(1-\epsilon)$. Hence, such an assignment cannot be $(1-\epsilon)$-fault-tolerant. Therefore, if $\hat{\phi}$ denotes an optimal, non-redundant schedule of $|M|$ unit size messages to the set $\{c_{d+1}, \ldots, c_n\}$ of reliable, related, parallel channels, and $\mathsf{Makespan}(\hat{\phi}) = B \geq \frac{|M|}{\sum_{i=d+1}^{n} c_i}$ denotes the makespan of $\hat{\phi}$, then B is a lower bound on the optimal Fault-Tolerant Congestion $\mathsf{Cong}^{\star}$. Next, we show that the congestion of the assignment produced by $\mathsf{SP\text{-}OPT}$ cannot exceed $2B\left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$.

The $\mathsf{Safe\ Partition}$ algorithm returns either a single group $\{(f_1, c_1), \ldots, (f_{d+1}, c_{d+1})\}$ consisting of the first $d+1$ channels, or a partition $\mathcal{K} = \{K_1, \ldots, K_\lambda\}$ consisting of λ groups. Moreover, in both cases, the cardinality of the groups $K_1, \ldots, K_\lambda, K_{\lambda+1} = K - \cup_{i=1}^{\lambda} K_i$ is at most $n^\star = \left\lceil \frac{\ln(n/\epsilon)}{\ln(1/f_{\max})} \right\rceil$.

At first, we consider the case that $c_{d+1} \geq C(\mathcal{K})$, and SP returns a single reliable effective channel of capacity c_{d+1}. Then, all the messages assigned by $\hat{\phi}$ to the channels $\{c_{d+1}, \ldots, c_{n^\star - 1}\}$ can cause a load of at most $n^\star B$ to a channel of capacity c_{d+1}. Let $b = \left\lceil \frac{n}{n^\star} \right\rceil$. Recall that the partition $\mathcal{K}$ consists of at least $b-1$ groups, and the effective capacity of the group K_l cannot be less than $c_{ln^\star}$, for all $l = 1, \ldots, \lambda$. Therefore, for all $j = 0, \ldots, n^\star - 1$, the quantity $\sum_{y=1}^{b-1} c_{yn^\star + j}$ cannot exceed the effective capacity of the partition $\mathcal{K}$. This implies that the messages scheduled on the set $K^{(j)} = \{(f_i, c_i) \in K : i = yn^\star + j, y = 1, \ldots, b-1\}$ by $\hat{\phi}$ can cause a load of at most B to a channel of capacity c_{d+1}. Since there exists exactly $n^\star$ sets $K^{(j)}$, and

$$\bigcup_{j=0}^{n^\star - 1} K^{(j)} = \{(f_{n^\star}, c_{n^\star}), \ldots, (f_n, c_n)\} \ ,$$

all the messages scheduled by $\hat{\phi}$ to the channels $\{c_{n^\star}, \ldots, c_n\}$ can cause a load of at most $n^\star B$ to a channel of capacity c_{d+1}. Hence, $\mathsf{Cong}(\phi) \leq 2n^\star B$.

Then, we consider the case that $C(\mathcal{K}) > c_{d+1}$, and SP returns a partition $\mathcal{K} = \{K_1, \ldots, K_\lambda\}$ of effective capacities $C(K_l) = \min_{i \in K_l}\{c_i\}$. Since, for all $l = 1, \ldots, \lambda$, any $c_i' \in K_{l+1}$ cannot exceed any $c_i \in K_l$, we obtain that $n^\star C(K_l) \geq \sum_{i \in K_{l+1}} c_i$. Therefore, all the messages assigned by $\hat{\phi}$ to the channels of the group K_{l+1} can cause a load of at most $n^\star B$, if they are assigned to a channel of capacity $C(K_l)$. Additionally, since we only consider unit size messages, for all $i = d+1, \ldots, n_1 = |K_1| \leq n^\star$, the messages assigned by $\hat{\phi}$ to any channel c_i can be assigned to a set of channels of total capacity $C(\mathcal{K}) > c_i$ so as to cause a maximum load of at most B. Therefore, $\mathsf{Cong}(\phi) \leq 2n^\star B$. □

In the proof of Theorem 6.1, the restriction to unit size messages is necessary only for bounding the maximum load that the messages assigned to the channels $c_{d+1}, \ldots, c_{n_1}$ by

$\hat{\phi}$ can cause to the set of channels $\{C(K_1), \ldots, C(K_\lambda)\}$, where $\sum_{l=1}^{\lambda} C(K_l) > c_i$, for all $i = d+1, \ldots, n_1$. Even though it is possible to replace the unit size assumption by a more general one on the relation between the message sizes and the capacities c_i and $C(K_l)$, we do not know how to completely avoid such a restriction.

The effective channel configurations computed by CCP are expected to be much more efficient than the configurations computed by SP. However, we do not know how to obtain a lower bound on $\mathsf{Cong}^\star$ by relating it with either the effective capacity $C(\mathcal{K}^\star)$ of an optimal partition, or the maximum value of the function $\sum_{j=0}^{\psi} 2^j \ \mathsf{IUB}\Big(I_j, (1-\epsilon)^{\beta_j^\star}\Big)$, subject to $\beta_j^\star \geq 0$, and $\sum_{j=0}^{\psi} \beta_j^\star \leq 1$.

7 Open Problems

A challenging open problem is to determine the complexity of Minimum Fault-Tolerant Congestion, and similar problems. It is straight forward that Minimum Fault-Tolerant Congestion is in $\mathcal{PSPACE}$, but we do not know if it belongs to the Polynomial Hierarchy $\mathcal{PH}$ (e.g., [10]). Obviously, Minimum Fault-Tolerant Congestion can be included in a class containing all the languages L that can be decided by a polynomial-time, non-deterministic Turing machine T reducing L to a single call of a function $g \in \#\mathcal{P}$. Moreover, after calling the oracle g once, the only additional computation that T needs to perform is an arithmetic comparison involving the outcome of g. Let this class denoted by $\mathcal{NP}^{\#\mathcal{P}[1,\mathrm{comp}]}$.

Up to the best of our knowledge, the class $\mathcal{NP}^{\#\mathcal{P}[1,\mathrm{comp}]}$ has not been studied so far. In addition to Minimum Fault-Tolerant Congestion, Stochastic Knapsack defined in [11] belongs to this class. A result of Toda and Watanabe [12] implies that $\mathcal{NP}^{\#\mathcal{P}[1,\mathrm{comp}]}$ contains the whole Polynomial Hierarchy. Moreover, an application of Cook's Theorem [13] (see also Section 17.2 of [10]) implies that the following problem is complete for $\mathcal{NP}^{\#\mathcal{P}[1,\mathrm{comp}]}$:

Given a Boolean formula $F(X_1, X_2)$ with Boolean variables partitioned into two sets X_1 and X_2, does there exist a partial truth assignment A for X_1 such that the remaining formula $F(A(X_1), X_2)$ has at least $(1-\epsilon)2^{|X_2|}$ satisfying partial truth assignments for X_2?

The research question arising from the above discussion is whether Minimum Fault-Tolerant Congestion, especially in case of related capacity channels, is also a complete problem for the complexity class $\mathcal{NP}^{\#\mathcal{P}[1,\mathrm{comp}]}$.

Another direction for further research is to derive a non-trivial lower bound for Minimum Fault-Tolerant Congestion in case of related capacity channels. This lower bound may be combined with the CCP algorithm in order to obtain a constant factor approximation algorithm.

Additionally, the fault-tolerant generalizations of some fundamental graph optimization problems, such as shortest path or minimum connectivity certificate, have not been studied so far under random and independent faults. In particular, the fault-tolerant generalization of minimum connectivity certificate is, given a graph $G(V, E)$, where each edge $e \in E$ fails independently with probability f_e, and a fault-tolerance constraint $(1-\epsilon)$, to compute the minimum (w.r.t. the number of edges) subgraph $G'(V, E')$, $E' \subseteq E$, that remains connected with probability at least $(1-\epsilon)$.

References

[1] M.V. Lomonosov, Bernoulli Scheme with Closure, *Problems of Information Transmission* **10**(1974), 73–81.

[2] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, 1979.

[3] L. Gasieniec, E. Kranakis, D. Krizanc, and A. Pelc, Minimizing Congestion of Layouts for ATM Networks with Faulty Links, *Proc. of the 21st Symposium on Mathematical Foundations of Computer Science*, 1996, 372–381.

[4] L.G. Valiant, The Complexity of Enumeration and Reliability Problems, *SIAM Journal on Computing*, **8**(1979), 410–421.

[5] D.S. Hochbaum (ed.), *Approximation Algorithms for $\mathcal{NP}$-hard problems*, PWS Publishing, 1997.

[6] S.F. Assmann, D.S. Johnson, D.J. Kleitman, and J.Y.-T. Leung, On a Dual Version of the One-Dimensional Bin Packing Problem, *Journal of Algorithms* **5**(1984), 502–525.

[7] O.H. Ibarra and C.E. Kim, Fast Approximation Algorithms for the Knapsack and Sum of Subset Problems, *Journal of the Association for Computing Machinery* **22**(1975), 463–468.

[8] E. Lawler, Fast Approximation Algorithms for Knapsack Problems, *Mathematics of Operations Research* **4**(1979), 339–356.

[9] D.R. Karger, A Randomized Fully Polynomial Time Approximation Scheme for the All Terminal Network Reliability Problem, *Proc. of the 27th ACM Symposium on Theory of Computing*, 1995, 11–17. Journal submission available from *http://theory.lcs.mit.edu/karger/papers.html.*

[10] C.H. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1994.

[11] J. Kleinberg, Y. Rabani, and E. Tardos, Allocating Bandwidth for Bursty Connections, *Proc. of the 29th ACM Symposium on Theory of Computing*, 1997, 664–673.

[12] S. Toda and O. Watanabe, Polynomial-time 1-Turing reductions from $\#\mathcal{PH}$ to $\#\mathcal{P}$, *Theoretical Computer Science* **100**(1992), 205–221.

[13] S.A. Cook, The Complexity of Theorem-Proving Procedures, *Proc. of the 3rd IEEE Symposium on the Foundations of Computer Science*, 1971, 151–158.