GEARSHIFT: Guaranteeing Availability Requirements in SLAs using Hybrid Fault Tolerance

Abstract—**The dependability of ICT systems is vital for today's society. However, operational systems are not fault free. Providers and customers have to define clear availability requirements and penalties on the delivered services by using SLAs. Fulfilling the stipulated availability may be expensive. The lack of mechanisms that allow a fine control of the SLA risk may lead to over-dimension the provided resources. Therefore, a relevant question for ICT service providers is: How to guarantee the SLA availability in a cost efficient way? This paper studies how to combine different fault tolerant techniques with different costs and properties, in order to economically fulfill a given SLA requirement. GEARSHIFT is a mechanism that enables ICT providers to set the fault tolerance technique (gear ratio) needed, depending on the current service conditions and requirements. We illustrate how to use the proposed model in a backbone network scenario, using measurements from a production national network. Finally, we show that the total costs of delivering an ICT service follow a simple convex function, which allows an easy selection of the optimal risk by tuning properly the combination of fault tolerant techniques.**

Index Terms—**fault tolerance; SLA; renewal theory; accumulated downtime; network recovery; risk optimization.**

I. Introduction

The importance of ICT services in today's society has increased tremendously, making their appropriate operation a matter of prime interest. However, failures in ICT systems are unavoidable events ([17], [7], [9]) that produce considerable negative consequences. A Service Level Agreement (SLA) is a tool to define among others, the service availability that a provider should deliver. In that sense, customers are aware of the unavoidable existence of failures, but providers are obligated to deliver the resources needed in order to guarantee the signed agreements.

Delivering highly dependable ICT services is expensive (e.g. [21] and [5]). Therefore, finding the right balance between the SLA availability fulfillment, and the cost of the technology to be used is a main concern for providers. One of the main tools to improve the control on the SLA development is the capability of modeling accurately the accumulated downtime over a finite interval. This problem was first addressed by Takács [20]. However, an explicit solution is only given for exponentially distributed up/down times. This work inspired several further works, among them, the work presented in [11], where several fundamental related concepts were proposed, analyzing different time distributions in systems that allow failures and repairs to occur. Finally, the probability that a service provider meet the contracted availability was first studied in [12] by Goyal and Tantawi, making it one of the main references for further SLA availability related works.

There are some previous studies that have addressed the combination of more than one fault tolerance techniques in ICT systems. The work presented in [6] studies a method for combining logical ring protection and mesh restoration mechanism, in order to recover link and node failures in optical networks. In [18], a hybrid model that uses network protection and restoration schemes for two-link failures was proposed. This work considers that in case of a simultaneous failure in the working and backup path, a dynamic rerouting should be applied. These works do not consider SLA scenarios, and they focus on an efficient recovery of a single downtime event rather than the accumulated downtime. The work presented in [10] proposes the combination of two different virtual machine restoration processes, assuming negatively exponentially distributed processes, but without considering financial consequences. Yallouz et al. propose a tunable network survivability approach in [23]. This work provides flexibility in the routing path selection in order to target a specific survivability level. However, it performs static decisions based on steady state failure probabilities, without considering the dynamics of the transient solution.

GEARSHIFT is a work that proposes how to build a *risk map* that leads fault tolerance shifts, considering: i) The accumulated downtime under generally distributed fault tolerance processes. ii) Optimal shifts based on the minimum financial cost, allowing ICT service providers to operate their resources in a cost efficient way.

This paper is organized as follows. Section II provides a general framework on the SLA context addressed in this paper, including the definition of the distribution of the accumulated down time and the SLA risk. In Section III, we define the SLA risk target, and we describe how to build the GEARSHIFT risk map. Section IV shows how to use GEARSHIFT in a backbone network scenario, based on real measurements obtained from a production network operator. In Section V, we evaluate the performance of GEARSHIFT using network recovery as a case study. Section VI illustrates how to select the optimal SLA risk target, that minimizes the total costs of delivering a service. Finally, Section VII concludes the paper.

II. SLA Risk and the accumulated down time

A definition of the *SLA Risk* and the *SLA success probability* is provided in this section. First, a general formulation is given in terms of the time distribution of up and down periods. Since the complexity of this formulation may be impractical in operational scenarios, the second part of this section presents an approximation that makes the computation of the distribution of the accumulated downtime more tractable, using conventional renewal theory concepts.

A. SLA Risk and the Distribution of the Accumulated Down Time

One of the most relevant parameters in SLAs is the availability guarantee α_A, and the duration of the contract period τ. In this paper, we will work with the SLA maximum allowed accumulated downtime α, which is obtained by mapping directly α_A (standard SLA parameter) into the time domain as $\alpha = (1 - \alpha_A)\tau$.

The state of an ICT service can be modeled as a function of the SLA running time t with a random process $O(t)$, which is equal to 1 if the service is working at t, or 0 otherwise.

The duration of each down period ($O(t)$=0) is assumed independent and identically distributed $h(t)$, and the duration of each up period ($O(t)$=1) is also assumed *i.i.d.* $g(t)$.

The accumulated downtime of an ICT service at time t $D(t)$ is a random variable that measures the total time that the service has been down during the interval $[0, t]$. At the end of the SLA ($t = \tau$), it may be defined as $D(\tau) = \tau - \int_0^\tau O(t)\,dt$.

The cumulative distribution function (CDF) and the probability density function (PDF) of $D(t)$ will be defined as $\Omega(\tau, t)$ and $\omega(\tau, t)$ respectively.

The *SLA Success Probability* is the probability that the accumulated downtime will be equal or smaller than α, and it may be defined as:

$$S(\tau, \alpha) \;=\; \int_0^\alpha \omega(\tau, t)\,dt \;=\; \Omega(\tau, \alpha). \tag{1}$$

The complementary function $(1 - S(\tau, \alpha))$ will be defined as the *SLA risk* ($P[\omega(\tau, t) > \alpha]$), and it represents the probability that the SLA availability requirement will not be met.

The evaluation of equation (1) demands the definition of $\omega(\tau, t)$ or $\Omega(\tau, t)$ alternatively. A general expression for $\Omega(\tau, t)$ was derived by Takács in [20] as follows:

$$\Omega(\tau, t) = \sum_{n=0}^{\infty} H_n(t)[G_n(\tau - t) - G_{n+1}(\tau - t)] \tag{2}$$

where the failure and repair processes are described by *i.i.d.* up and down times with CDF $G(t)$ and $H(t)$ respectively, and the subindex n represents the n convolution of a given function.

Equation (2) characterizes a problem with general distributions. However, it is difficult to compute for specific failure and repair processes due to the complexity posed by the convolution of generally distributed CDFs.

For the case of failure and repair processes being negatively exponentially distributed *n.e.d.*, a complete result was obtained by Takács [20].

Previous studies (e.g., [17], [7]) have shown that Weibull and gamma distributions are representative to model the behavior of real failure and repair processes (the results presented in Section IV also agree with this observation). In the literature there is no explicit expression that describes $\Omega(\tau, t)$ for these two distributions. Therefore, the next section shows an approximation for $\Omega(\tau, t)$ and $\omega(\tau, t)$ that makes their computation more tractable.

B. Approximation of the Distribution of the Accumulated Down Time

Assuming a deterministic number of down events n when $t = \tau$, the realization of the accumulated downtime of a service $\overline{D(\tau)}$ is simply the addition of the realization of each individual downtime, i.e., $\overline{D(\tau)} = h_1 + h_2 + ... + h_n$. Using this logic, the PDF of $D(t)$ is given by the convolution $\omega_n^*(t) = h^1(t) * h^2(t) * ... * h^n(t)$. Therefore, if the number of down events is known, $\omega(\tau, t)$ can be obtained. However, this quantity (n) is also a random variable with a range from zero to infinity.

In order to compute $\omega(\tau, t)$ considering the stochastic properties of the number of down events, one can assume that the downtime duration is very small compared to the duration of uptimes (realistic assumption in most ICT systems). Therefore, the probability of n down events during τ $P(N(\tau) = n)$ may be approximated considering only the number of renewals of the failure process which is ruled by the distribution of $g(t)$, i.e., $D(t)$ is not considered for the computation of $P(N(\tau) = n)$. This approximation represents an upper bound, since $P(N(\tau) = n) \;\geq\; P(N(\tau - D(t)) = n)$, and hence a conservative and safer way to estimate the SLA risk.

This approximation overcomes the complexity of Equation (2) by dividing the problem in two. First by making the convolution of only the downtime distribution. Second by obtaining $P(N(\tau) = n)$, which is ruled only by the uptime distribution, where renewal theory and counting models may be used. The approximated PDF of the total accumulated downtime is given as

$$\omega^{'}(\tau, t) = \sum_{n=0}^{\infty} P(N(\tau) = n) h_n^*(t) \tag{3}$$

The approximation proposed in Expression (3) makes easier the computation of the distribution of the accumulated downtime. For instance, the probability $P(N(\tau) = n)$ of n renewals during τ for *n.e.d.* uptimes follows the Poisson distribution. The convolution of exponential functions can be easily obtained applying Laplace transform. In [22], Winkelmann defines a count-data model that computes $P(N(\tau) = n)$ when the times are gamma distributed. The convolution of gamma distributed downtimes is obtained directly by using Laplace transform. For the case of Weibull distributed uptimes, $P(N(\tau) = n)$ can be obtained by expanding the Weibull function in Taylor series as presented in [16], and the convolution of Weibull distributed downtimes may be approximated using the Saddle Point Approximation [14].

III. Hybrid Model Formulation

There is a tradeoff between fulfilling the SLA availability, and the amount of resources needed in order to do that. This section shows how to address this issue, through the

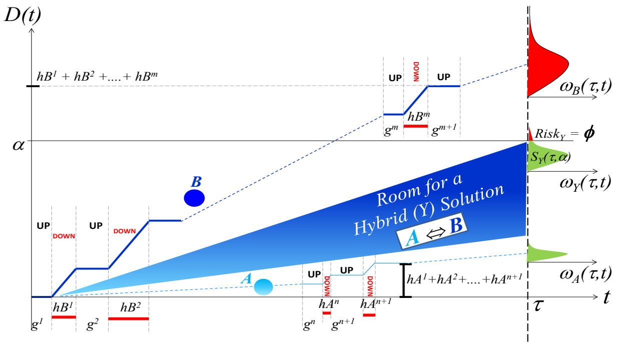

Fig. 1. $\omega_A(\tau, t)$, $\omega_B(\tau, t)$, and the SLA requirement α.

proposal of a hybrid model. Figure 1 illustrates the evolution of the accumulated downtime under two different fault tolerance techniques named here technologies A and B. In general, our model assumes that technology A has better recovery properties than B. The distribution of the duration of individual downtimes using technology A or B will be defined as $h_A(t)$ or $h_B(t)$ respectively. In addition, we assume in general that the operation of technology A is more expensive and demands more resources than technology B. Otherwise, the selection of the fault tolerance technique to be used would be trivial for the provider. Without loosing generality, we will keep the same notation $g(t)$ for the uptime distribution associated with the failure process on both cases (technology A and B).

The distribution of the accumulated downtime using technology A or B ($\omega_A(\tau, t)$ or $\omega_B(\tau, t)$) can be calculated in advance, using the mechanisms provided in Section II-B. Due to the better dependability properties of technology A, the density $\omega_A(\tau, t)$ will be concentrated in values of $D(t)$ much lower than $\omega_B(\tau, t)$, as Figure 1 illustrates.

Section II also shows that the accumulated downtime is a continuous random variable with a range from zero to infinity. Therefore, having a SLA risk equal to zero is simply not possible. Based on this fact, here we define the *SLA risk target* ϕ as the maximum value that the service provider allows for $1-\Omega(\tau, \alpha)$. The selection of ϕ allows a better planning on the SLA needs and consequences, but it is an open value that the provider should select based on a proper financial assessment. Section VI will present how such assessment may be done, in order to chose the optimal value of ϕ.

The maximum allowed accumulated downtime α is the parameter that defines if $\omega_A(\tau, t)$ or $\omega_B(\tau, t)$ may produce an acceptable SLA risk, according to ϕ. In this context, there are three possible scenarios: **i)** $1-\Omega_B(\tau, \alpha)$ is shorter than the risk target. In this case, there is no reason to use technology A, since technology B is able to handle the risk alone. **ii)** $1 - \Omega_A(\tau, \alpha)$ is larger than the SLA risk target. In this case, reaching ϕ is not possible. Therefore, the SLA should not be signed, the risk target should be relaxed (with the commercial implications that it may bring), or new and better technologies than A should be used (if available). **iii)** $1-\Omega_B(\tau, \alpha)$ is larger, and $1 - \Omega_A(\tau, \alpha)$ is shorter than the risk target respectively. In this case, only technology A can handle the desired risk. However, using this solution during the entire SLA period may not be the most efficient solutions in terms of cost efficiency.

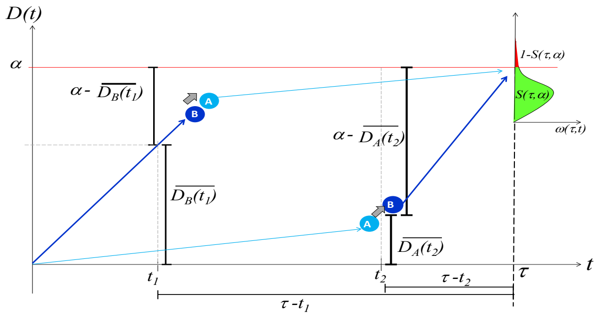

Fig. 2. Two potential hybrid solution approaches.

This paper is focused on the mentioned third case, where there is room for a hybrid solution that through the combination of technology A and B may deliver precisely the desired risk target ϕ, as Figure 1 illustrates.

In this paper, two intuitive ways to combine the use of fault tolerance technologies A and B are studied. **i)** *Spend and Save*. **ii)** *Save and Spend*. In addition, at the end of this section, a generalization of the hybrid model will be proposed.

A. Approach 1: Spend and Save

Spend and Save is a hybrid fault tolerance approach that uses technology B at the beginning of the SLA. This initial phase is called *spend*, since the probability of having a large increase in the accumulated downtime after a failure is much higher with technology B than with technology A. In this way, one can say that the allowed downtime before paying penalty is spent quickly.

In order to have a tight control on the risk of violating the SLA availability parameter, the provider has the opportunity to switch to technology A at any time, starting the *save* phase. However, it is desired to delay this shift as much as possible, due to the high cost and resource utilization that this technology may imply.

The main challenge of this approach is to find the *appropriate* point W to switch to technology A. The term *appropriate* here is based on two criteria:

- Criteria 1: The decision has to avoid a late transition that increases the SLA risk beyond ϕ.
- Criteria 2: The switching decision has to be delayed as much as possible, in order to have a more cost efficient operation.

In the *spend and save* approach, the switching decision will be made only after the recovery process due to: **i)** If the connection is working correctly, $D(t)$ moves horizontally (no increase in the accumulated downtime) as Figure 1 illustrates, and hence, the SLA risk is not affected negatively. **ii)** When a failure event happens, the service has to face the downtime associated with the currently fault tolerance technique in use.

Based on that, when the recovery process successfully ends at time t_1, the accumulated downtime for that service will be

$\overline{D_B(t_1)}$, as Figure 2 shows. At this point in time, the service provider has the following options:

1) Option 1: The connection immediately switches to technology A.
2) Option 2.1: The connection continues using technology B, but after the next coming failure there is a transition to A (only one additional down event under technology B).
3) Option 2.2: The connection continues using technology B for a number of n additional down events.

In order to evaluate the option to be selected, we need to define a transition frontier W, based on a SLA risk assessment as described in Section II.

Computing the SLA risk for Option 1, means to consider only downtime events with a distribution $h_A(t)$. However, the obtained SLA risk may go against the Criteria 2 previously mentioned, since it may be yet space for the use of technology B.

In order to know if there is still room for the use of technology B at the service state $(t_1, \overline{D_B(t_1)})$, we need to make a SLA risk assessment considering the scenario with two different kind of downtime distributions: $h_B(t)$ and $h_A(t)$, as defined by Option 2.

Due to the downtime duration under technology B is a stochastic variable, finding a transition frontier W is challenging because several failure events under technology B may happen without putting in danger the SLA risk target selected. In other words, it is unknown the number of downtime events (renewals) with distribution $h_B(t)$ before the transition to technology A. In order to solve this problem, we use renewal theory, following the model illustrated in Figure 3.

A common scenario in Option 2.1 and 2.2 is the existence of a single last down time with distribution $h_B(t)$ before the transition to technology A. We are interested in finding the distribution of the downtime after the transition point W $h_B^{'}(x)$.

Following the rationale explained in [4], we can define that the last down time with distribution $h_B(t)$ lies in the interval $(x\ ,\ x + \Delta x)$ if:

- The recovery from the first down event l finishes in the interval $(W + x\ ,\ W + x + \Delta x)$ (Option 2.1).
- For a given U, a failure event (renewal) occurs in the interval $(W - U\ ,\ W - U + \delta U)$, and the recovery finishes in the interval $(U + x\ ,\ U + x + \Delta x)$ (Option 2.2)

Considering the previous conditions, the probability density function of $h_B^{'}(x)$ will be defined as:

$$h_B^{'}(x) \;=\; h_B^l(W + x) + \int_0^W r(W - U) h_B(U + x)\, du \quad (4)$$

Where $r(W - U)$ is the renewal density function.

Using renewal theory, we can assume that when $W \to \infty$, $h_B^l(W) \to 0$, and the renewal density tends to the expected value of the technology B downtime duration ($r(W - U) \to 1/E[h_B]$]), as shown in [4]. Therefore, we can define the asymptotic behavior of $h_B^{'}(x)$ as:

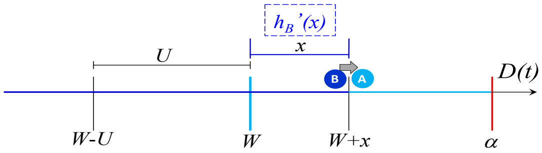

Fig. 3. Distribution of the last accumulated downtime before the frontier W (Forwarding Recurrence Time)

$$\lim_{W \to \infty} h_B^{'}(x) \;=\; \frac{1}{E[h_B]} \int_0^{\infty} h_B(U + x)\, du \;=\; \frac{1}{E[h_B]} \int_x^{\infty} h_B(v)\, dv \;=\; \frac{1 - H_b(x)}{E[h_B]} \quad (5)$$

This results agrees with the function of the limiting distribution forward recurrence time [4]. The approximation of the non-asymptotic distribution of the forward recurrence time has been widely studied, where several accurate proposals have been formulated by the use of numerical methods [8]. Among them one of the most common approaches consist in the discretization of $h_B(t)$ ($hB[n]$), followed by the formulation of a system of linear equations that may be solved using recursive methods ([8], [2]). For instance, in [2] $h_B^{'}(x)$ may be approximated using the expression:

$$hB_{W+1}^{'}[n-1] \;=\; hB_W^{'}[n] + \frac{hB_W^{'}[0] \cdot hB[n]}{1 - hB[0]} \quad (6)$$

where $hB_1^{'}[n] = hB[n+1]/(1 - hB[0])$.

Now, the total accumulated downtime after the transition point W will be the time accumulated under the process ruled by $h_B^{'}(t)$, plus the downtime of the next coming failures under technology A. Based on the results obtained in Section II-B, the accumulated downtime and the SLA success probability after W for the approach 1 proposed in this section (spend and save) may be obtained as follows:

$$\omega_1^{'}(\tau - t_1, t) = P(N(\tau - t_1) = 0) + \quad (7)$$
$$P(N(\tau - t_1) = 1) h_B^{'}(t) + \sum_{n=2}^{\infty} P(N(\tau - t_1) = n) hY_n^*(t)$$

$$S_1(\tau - t_1, \alpha - \overline{D_B(t_1)}) \;=\; \int_0^{\alpha - \overline{D_B(t_1)}} \omega_1^{'}(\tau - t_1, t)\, dt. \quad (8)$$

where $hY_n^*(t)$ is the convolution of one distribution of the downtime under the process ruled by $h_B^{'}(t)$, with the $n-1$ convolution of the distributions of the downtime under technology A, i.e., $hY_n^*(t) \;=\; h_B^{'}(t) * h_A^1(t) * ... * h_A^{n-1}(t)$.

Finally, the transition to technology A will be made if:

$$1 - S_1(\tau - t_1, \alpha - \overline{D_B(t_1)}) > \phi \quad (9)$$

Equation (9) will be used from now on to compute the transition frontier when the spend and save approach will be used during the rest of the paper.

B. Approach 2: Save and Spend

Save and Spend is a solution where the network connection uses technology A at the beginning of the SLA period. This initial phase is called *save*, since there is a very high probability of having a shorter downtime duration when a failure occurs. On the other hand, the use of technology A implies more costs and resources. Therefore, switching to technology B as early as possible is something of interest. We will define the time when transition from A to B happens as t_2.

In order to make a responsible transition to B, the mechanism has to verify that the *saving* at t_2 is enough, i.e., the SLA risk posed by using solution B from time t_2 is equal or smaller than the targeted risk ϕ. This case is much simpler than the one explained previously, since there is no condition that encourages a late transition, allowing its execution at any time. The only condition to switch technology will be based on the SLA risk assessment under technology B.

Using Figure 2 as reference, one can see that at point t_2, the accumulated downtime of the service is $\overline{D_A(t_2)}$. The remaining SLA time before the contract finishes is $\tau - t_2$, and the remaining allowed accumulated downtime will be $\alpha - \overline{D_A(t_2)}$. With this information, the accumulated downtime and the SLA success probability after t_2 may be obtained as follows:

$$\omega_2^{'}(\tau - t_2, t) = \sum_{n=0}^{\infty} P(N(\tau - t_2) = n) hZ_n^*(t) \qquad (10)$$

$$S_2(\tau - t_2, \alpha - \overline{D_A(t_2)}) = \int_0^{\alpha - \overline{D_A(t_2)}} \omega_2^{'}(\tau - t_2, t)\, dt. \qquad (11)$$

where $hZ_n^*(t)$ this time is the convolution of the distribution of n downtime events under technology B, i.e., $hZ_n^*(t) = h_B^1(t) * h_B^2(t) * ... * h_B^n(t)$.

Finally, the transition to technology B will be made at any time t_2 if:

$$1 - S_2(\tau - t_2, \alpha - \overline{D_A(t_2)}) > \phi \qquad (12)$$

C. Hybrid Fault Tolerance - A General Approach

Equations (9) and (12) define how to assess a potential transition. Assuming that the provider is able to obtain $g(t)$, $h_A(t)$ and $h_B(t)$, it is possible to obtain the transition frontiers for a shift in fault tolerance technology. Figure 4 present a *Risk Map* that contains the defined risk areas and the transition frontiers, using the models and data presented in Section IV.

The red line represent the transition frontier from B to A, and the green line represents the transition frontier when the provider is allowed to switch from A to B.

Using the information presented in Figure 4, we can generalize our hybrid approach as follows:

- The hybrid concept applies for any ICT service where: **i)** Specific availability constraints need to be fulfilled. **ii)** Different fault tolerance techniques are available. **iii)** Cost efficient operations is a matter of interest.

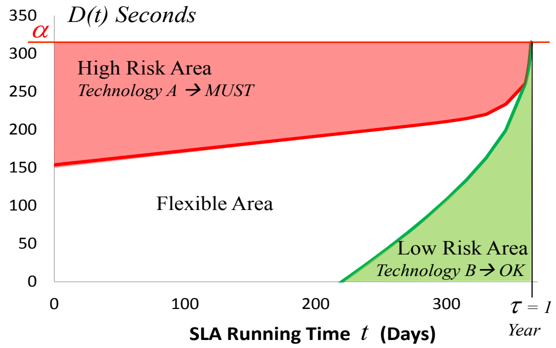

Fig. 4. GEARSHIFT Risk Map: Risk areas and transition frontiers in a Hybrid Fault Tolerant approach.

- Any ICT service will have specific coordinates $[D(t_x)\ ,\ t_x]$ on the risk map that will allow the identification of the current service risk area, and hence the fault tolerance technology to be used.
- If an ICT service is at the high risk area, the ICT provider must deliver the service using the best fault tolerance technology available.
- The low risk area allows the use of the lowest cost technology, without loosing control on the SLA risk.
- The flexible area has two implications: **i)** There is not strict requirements for the use of a specific fault tolerance technique. **ii)** If three or more fault tolerance techniques are involved, this area may be subdivided accordingly.

IV. Network Recovery

Backbone networks are the example of a classical scenario where SLAs are important and the existence of different fault tolerance techniques is a fact. In order to illustrate the implementation of the concepts previously presented in this scenario, we consider two commonly used network recovery mechanisms: path protection and path restoration. They are mechanism that react to failures by redirecting the traffic to alternative failure-free paths, and can be linked with technology A and B respectively.

A. Path Restoration

In path restoration, when a connection is interrupted by a failure, the packets are dynamically rerouted over a backup path which is computed on demand when the failure is detected. Restoration is considered a resource efficient mechanism, given that backup resources are only reserved when they are needed. We are aware of the resource utilization benefits of restoration, but we are also aware that safer and faster network recovery polices exist, e.g., path protection.

For a better illustration of the performance of path restoration mechanisms, in this paper we performed measurements on a production national backbone network on end-to-end connections between different cities.

Active measurements were performed between several servers located at different cities. End-to-end downtime statistics were continuously collected during one year, from January

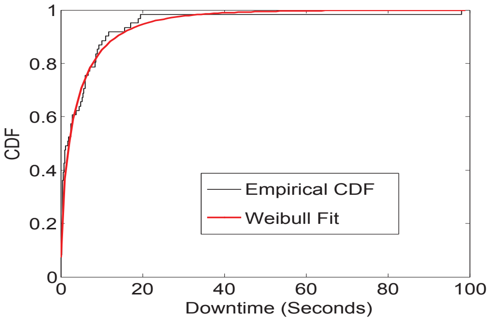

Fig. 5. Measured downtime durations using path restoration

to December 2012. In addition, traceroute was used in order to identify the path between the end-points.

The measurements were performed by sending an evenly distributed stream of packets between server nodes representing end-to-end paths. The interval between probe packets was chosen to be 10 ms to have an acceptable accuracy in the duration of the downtime. The end systems are synchronized using NTP protocol. The programs used for the measurement streams were rude/crude [1]. Each packet is timestamped both on sending and receiving and also has a sequence number. Hence jitter, loss and reordering can be obtained.

We monitor the network load of the network and perform the measurements in areas where congestion is not significant (lightly loaded), in order to assure than service downtimes were consistently due to network element failures and network miss-operation of various kinds rather than congestion. In spite of this, certain down periods may be caused by temporary congestion. In this case, those events where detected by observing a continuous increase in the packet delay, and disregarded as failure events.

The analyzed operational backbone network is currently running IS-IS link state routing [13], i.e., only path restoration is applied for link (and router) failures. Fault detection is based on periodic "hello"-messages on layer 3 (IP) exchanged between all neighboring routers. The current message period is 1000ms. Lack of two consecutive incoming messages results in a fault condition for a link. Link state announcements (LSA) are then flooded via (still connected) neighbor routers to all other routers in the backbone network. The maximum one-way delay between two routers in the backbone is around 17ms. On reception of a LSA, routers update their routing information database (RIB), start recalculating shortest paths, and finally update their forwarding information base (FIB). A full recalculation including FIB update may take up to 250ms, but partial recalculations are more common and result in approximately 50ms RIB+FIB update time. Summarized, any single link failure should in theory be restored in $2 * 1000 + 17 + 250 = 2267ms$.

Figure 5 shows the cumulative distribution of the single down time durations observed in one of the end-to-end measurements performed. We select this specific case due to its rich stochastic behavior (wide downtime range) that may be generated among others because: detection, new path calculation and route propagation times, routing protocol convergence delays, simultaneous and/or reiterative failures, miss-operations, failure detection issues, and other unidentified potential threats. An important remark here is that this behavior motivates a proper SLA risk assessment, instead of using directly the expected values, usually mentioned on the protocol specifications. The results obtained from the measurements can be associated with technology B, as defined in Section III. For this specific example, one can say that the distribution $h_B(t)$ presented in Section III may be modeled using a Weibull distribution with scale and shape parameters of 5079.97 ms and 0.54 respectively.

B. Dedicated Path Protection

Path protection is a fault tolerance mechanism designed for telecommunications networks, where backup resources are pre-computed and reserved across the network topology. If any failure occurred at any point in a working path, the end nodes will move the traffic to or from the alternate route.

In Dedicated Backup Path Protection (DBPP) each working path has its own dedicated backup path. In case of failure, there is no need of additional signalling between the source and the destination node in order to establish the backup path. Therefore, the source node only needs to detect the failure and switch the traffic over to the backup path. The studies made in [19] and [21] (among others) have proved that DBPP can provide recovery times below 50 milliseconds. It is important to mention that this performance does not only depend on the use of redundant pre-reserved resources, but in addition, it also relies on the use of sophisticated failure detection mechanisms such as Loss of Signal (LOS) and Bidirectional Forwarding Detection (BDF) [15].

Proving dedicated backup path protection to all users is however very demanding for the network provider in terms of bandwidth usage and costs. This concept fits with the features of technology A, and it motivates the use of a hybrid approach.

Some case studies made in [19] and [21] found that the downtime distribution of dedicated backup path protection (DBPP) may oscillate between 40 to 50 millisecond. These kind of results may be optimistic, since they assume a perfect transition. In real operations, DBPP does not always perform a perfect switching, being failure correlation one of its main threats. Several studies (e.g., [9] and [17]) show that failures in routers and links in an IP backbone network may be simultaneous and/or correlated, and they can be modeled with a conditional probability P_s. To cope with such situations, a connection using DBPP should also get access to additional resources available in case that the pre-configured backup path does not work, i.e., DBPP should perform path restoration in those extreme cases. For instance the work developed in [18] suggests that in case of a simultaneous failure in the working and backup path, an online rerouteing (restoration) should be applied to restore the traffic.

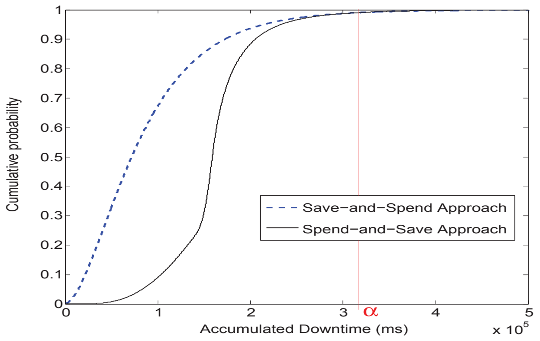

Fig. 6. Simulation results of the accumulated downtime under SPEND-SAVE and SAVE-SPEND approaches

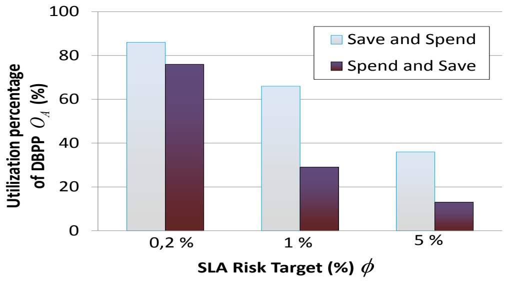

Fig. 7. Utilization Percentage of DBPP under Save-Spend and Spend-Save approaches for a given ϕ

In order to address this issue, here we propose a model that is able to capture better the risk implications by implementing DBPP.

To calculate the probability density of the downtime duration with a DBPP scheme, two scenarios are considered: **i).** Perfect condition: No simultaneous failures in the working and backup path. **ii).** Flawed condition: Simultaneous failures that force DBPP behave as a path restoration scheme.

We can use the density $h_B(t)$ obtained in Section IV-A as the distribution of the downtime duration under restoration. On the other hand, based on the results presented in [19], the density of DBPP under perfect conditions can be modeled using a uniform distribution between 40 to 50 millisecond, and it can be notated as $h^{'}_A(t)$. Thus, a more risk aware model for $h_A(t)$ can be obtained as follows:

$$h_A(t) = (1 - P_s) \cdot h^{'}_A(t) + P_s \cdot h_B(t) \tag{13}$$

Alternatively, $h_A(t)$ can be obtained directly from network measurements, if there is enough data that allow a realistic model.

Using the presented models, we found a huge gap between the accumulated downtime densities of path protection and path restoration, being this an additional motivation for implementing the proposed hybrid approach in a network recovery scenario.

V. Hybrid Model Evaluation

Figure 4 presented the transition lines and risk areas of a hybrid fault tolerance model that combines DBPP with path restoration, based on the concepts and results presented in Section IV, with a SLA risk target ϕ = 1%.

In order to see how to implement the guidelines provided by this risk map (Figure 4), and see the accuracy on targeting the selected ϕ, we use discrete event simulation. We consider end-to-end network connections where failures arrive at unexpected times according to a *n.e.d* with expected value of 15 days. As presented in Section IV, $h_B(t)$ is modeled using a Weibull distribution with scale and shape parameters 5079.97 ms and 0.54 respectively, and $h_A(t)$ follows Equation (13) with a measured value of P_s equal to 0.01. Those failures make the network connection state move around the risk map, generating at some point a shift on network recovery mechanism.

In Figure 6, one can see the CDFs of the accumulated downtime under the two approaches presented in Section III. The Save-and-Spend approach presents an early increase in the cumulative probability, which means that very short accumulated downtime values have statistical significance. On the other hand, the probability of having very short values of $D(t)$ is much smaller in the Spend-and-Save approach, which means that its density is more concentrated in intermediate values, but smaller than α.

A very important remark is that the value of $\Omega(\tau, \tau)$ is the same on both approaches (99%), which fits with the SLA risk target ϕ selected. Here, there are two important observations: **i).** If the service provider wrongly considers $h_B(t)$ instead of $h^{'}_B(t)$ for the derivation of $hY^*_n(t)$ in Equation (7), the SLA risk obtained will be dangerously higher than ϕ, depending on the failure/recovery process and the value of ϕ. For instance, for the case study illustrated in Figure 6 ($\phi = 1\%$), the obtained SLA risk was 2.1%. This approach would only be valid if the density $h_B(t)$ is *n.e.d.* **ii).** If $h^{'}_B(t)$ is computed using the asymptotic value presented in Equation (5), the SLA risk obtained will be conservatively lower than ϕ, depending on the failure/recovery process and the value of ϕ. For instance, for the case study illustrated in Figure 6 ($\phi = 1\%$), the obtained SLA risk is 0.92%. Although this difference is safe for the SLA, the risk target can be precisely achieved using the numerical methods mentioned in Section III-A.

We are interested in evaluating the difference in the resources needed by both approaches, and the influence of ϕ on that. For this reason, in Figure 7, we present the utilization percentage of DBPP O_A during the SLA period, considering the two hybrid approaches and different ϕ values. The lower utilization of DBPP obtained by the Spend-and-Save approach is an interesting results that justifies its preference in an implementation scenario. For instance, the difference of approximately 36% obtained when ϕ= 1% may represent a huge saving in network resources and operational costs, without loosing control on the SLA risk.

Finally, we evaluate the influence of the burstiness of the recovery process (downtime duration). For this we variate the shape parameter in the Weibull function, keeping the same expected value in all cases by tuning the scale parameter.

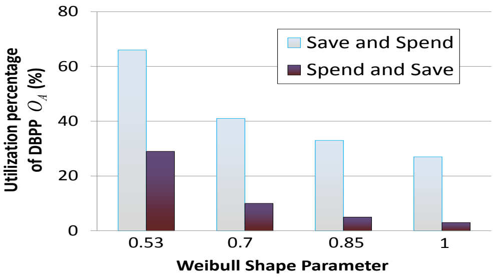

Fig. 8. Utilization Percentage of DBPP under Save-Spend and Spend-Save approaches for different recovery burstiness (Weibull shape parameter) with same expected value by tuning the scale parameter.

Figure 8 illustrates this evaluation by fixing the SLA risk target in 1%. We observe that a recovery process with smaller burstiness (e.g., larger Weibull shape parameter) allows the utilization of less resources, and hence we can assume a less demanding SLA availability fulfillment. On the other hand, the decrease of the burstiness (e.g., increase in the Weibull shape parameter) also reduces slightly the difference between the utilization of DBPP of the two hybrid approaches studied. For instance, with a shape parameter equal to 1 (*n.e.d.*), the difference is approximately 24%.

VI. Optimal SLA Risk

As mentioned in previous sections, by combining a lower-cost/higher-risk technology such as restoration with a higher-cost/lower-risk technology such as protection, the probability of failing the SLA can be controlled economically. One of the main issues in the implementation of this combination (hybrid model) is the selection of the SLA risk target ϕ. In this section, we present a procedure that can be used to determine the optimal value of ϕ, using network recovery as a case study. More particularly, we seek a trade-off between the higher costs incurred by the protection, and the higher expected penalty costs in case of using restoration. The financial target is to minimize the total expected costs C_T, which consist of the expected penalty costs C_P, plus the expected operational costs C_O, by using protection during a period $\tau \cdot O_A$ and restoration during a period $\tau \cdot (1 - O_A)$.

Given that the Spend and Save approach provides a better resource utilization by delivering the same risk, this approach should be used in any scenario where financial targets are important.

The network provider is confronted with the decision problem to define the SLA risk target ϕ. The value of ϕ has a direct implication on an earlier or later switching from protection to restoration, by moving up or down the transition frontier in the GEARSHIFT risk map. This affects directly the utilization percentage time of network protection O_A, and the expected penalty costs generated by the violation of the SLA.

We assume that after exceeding the SLA maximum allowed accumulated downtime α, the penalty costs increases linearly with a gradient p. The penalty cost $C_P(\phi, D(\tau))$ will depend on the SLA risk target ϕ and the realization of the accumulated downtime $\overline{D(\tau)}$ at time τ (end of the SLA). It will be defined as follows:

$$C_P(\phi, D(\tau)) = \begin{cases} p \cdot (\overline{D(\tau)} - \alpha) & \text{if } \overline{D(\tau)} \geq \alpha. \\ 0 & \text{if } \overline{D(\tau)} < \alpha. \end{cases} \quad (14)$$

We presented in Section V the impact of ϕ on the utilization percentage of network protection O_A, and the remaining utilization percentage of network restoration $(1 - O_A)$ (see Figure 7). In addition, we define the operational costs of network protection as C_A and the operational costs of network restoration as C_B. Based on this, we can define the operational costs of our hybrid model as:

$$C_O(\phi, D(\tau)) = C_B \cdot (1 - O_A) + C_A \cdot O_A \quad (15)$$

Summarizing the decision problem can now be stated as:

$$\underset{\phi}{\text{minimize}} \quad \mathbf{E}_\omega[C_P(\phi, D(\tau)) + C_O(\phi, D(\tau))] \quad (16)$$

There are some important considerations regarding the problem stated in (16). When ϕ increases, there is a point ϕ_B where it becomes very flexible, allowing the SLA risk target to be fulfilled by using only restoration, i.e., $1 - \Omega_B(\tau, \alpha) < \phi$. Considering ϕ values bigger than ϕ_B does not have any sense for the proposed hybrid model (see Section III). On the other hand, when ϕ decreases, there is a point ϕ_A where it makes compulsory the use of protection from the very beginning of the SLA, i.e., the transition frontier is equal to zero at $t = 0$. Therefore, in this case considering ϕ values shorter than ϕ_A does not have any sense for the proposed hybrid model.

Using discrete event simulation, the expected total costs presented in (16) can be evaluated for fixed values of ϕ_A and ϕ_B respectively, by generating a large number of stochastic replications that allow the evaluation of $\mathbf{E}_\omega[C_T]$ with negligible error, as described in [3].

Following a simple local search heuristic approach, we generate iterative steps by increasing and reducing ϕ_A and ϕ_B respectively. We found that the C_T follows a convex function with a single global minimum which makes feasible and simple the successful implementation of our heuristic approach.

In order to illustrate this, we will implement the proposed local search approach in a scenario where the penalty gradient p is equal to $0.2X$ cost units per unit of time. In addition, in our case study the number of links in the backup path is the same than in the working path. Therefore, we assume a simplified scenario with double cost for C_A. In this case C_A is equal to $2X$ cost units, and a cost C_B equal to X cost units. The cost behavior of this scenario is presented in Figure 9. We can observe that for short values of ϕ, the operational cost is very high, since this high requirement demands the use of protection a large percent of the SLA time, but at the same time the expected penalty costs are very small. The gradient of the function described by the expected operational costs is negative and its absolute value decreases quickly with the

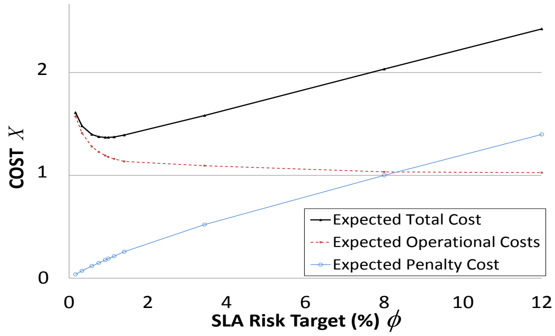

Fig. 9. Optimal Selection of the SLA Risk Target ϕ.

increment of ϕ. On the other hand, the gradient of the function described by the expected penalty costs is positive, and it decreases slightly with the increment of of ϕ. For the specific case illustrated in this section, we found that the minimum total cost is obtained when the SLA risk target is equal to 1%. A value that was found after 8 iterations of the local search heuristic approach previously described.

VII. Conclusion

This paper describes how to build a risk map that can be used to lead the use of different fault tolerance techniques, depending on the current conditions and requirements of a service. It considers the behavior of generally distributed failure and recover processes, and the financial impact generated by the selection of a given risk. Simulation results show that by following the transition frontiers defined in the risk map, the provider is able to define a specific SLA risk target, having a tighter control on the SLA. We show how to implement GEARSHIFT in a network recovery scenario, using real measurements from an operational network. In addition, those measurements show the importance of analyzing the operational behavior of fault tolerance techniques, considering the entire stochastic behavior, instead of using directly the expected values mentioned in protocol specifications. We show that the burstiness of the fault tolerance processes has a strong influence on the transition frontiers defined in the risk map, and hence on the amount of resources to be delivered to a service. Finally, we show that the total cost of the studied hybrid fault tolerance scenario follows a convex function that allows the optimal selection of the SLA risk target to be chosen.

References

[1] Rude/crude real-time UDP data emitter/collector. [Online]. Available: http://rude.sourceforge.net/.

[2] Baganha, M. P., Ferrer, G., and Pyke, D. F. The residual life of the renewal process: A simple algorithm. *Naval Research Logistics (NRL) 46*, 4 (1999), 435–443.

[3] Banks, J., Carson, J. S., Nelson, B. L., and Nicol, D. M. *Discrete-Event System Simulation (4rd Edition)*, 4 ed. Prentice Hall, 2005.

[4] Cox, D. R. *Renewal theory*. Methuen's monographs on applied probability and statistics. Methuen, 1967.

[5] DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., and Vogels, W. Dynamo: amazon's highly available key-value store. *SIGOPS Oper. Syst. Rev. 41*, 6 (Oct. 2007), 205–220.

[6] Felske, K., Gruber, J., and Wang, G. Two stage, hybrid logical ring protection with rapid path restoration over mesh networks, Dec. 10 2003. EP Patent App. EP20,010,302,630.

[7] Ford, D., Labelle, F., Popovici, F. I., Stokely, M., Truong, V.-A., Barroso, L., Grimes, C., and Quinlan, S. Availability in globally distributed storage systems. In *9th USENIX conference on Operating systems design and implementation* (Berkeley, CA, USA, 2010), OSDI'10, USENIX Association, pp. 1–7.

[8] From, S. G. Some new approximations for the renewal function. *Communications in Statistics - Simulation and Computation 30*, 1 (2001), 113–128.

[9] Gill, P., Jain, N., and Nagappan, N. Understanding network failures in data centers: measurement, analysis, and implications. In *Proceedings of the ACM SIGCOMM 2011 conference* (New York, NY, USA, 2011), ACM, pp. 350–361.

[10] Gonzalez, A., and Helvik, B. Hybrid cloud management to comply efficiently with sla availability guarantees. *12th IEEE International Symposium on Network Computing and Applications (NCA)* (Aug 2013), 127–134.

[11] Goyal, A., Nicola, V. F., Tantawi, A., and Trivedi, K. Reliability of systems with limited repairs. *Reliability, IEEE Transactions on R-36*, 2 (June 1987), 202–207.

[12] Goyal, A., and Tantawi, A. A measure of guaranteed availability and its numerical evaluation. *IEEE Transactions on Computers Volume 37, Issue 1* (1988), 25 – 32.

[13] Gredler, H., and Goralski, W. The complete IS-IS routing protocol. *SpringerVerlag* (2004).

[14] Huzurbazar, S., and Huzurbazar, A. V. Survival and hazard functions for progressive diseases using saddlepoint approximations. *International Biometric Society Journal of Biometrics 55*, 1 (1999), pp. 198–203.

[15] Katz, D., and Ward, D. Bidirectional Forwarding Detection BFD. rfc-5880. *IETF* (2010).

[16] Lomnicki, Z. A. A note on the Weibull renewal process. *Biometrika 53*, 3/4 (1966), pp. 375–381.

[17] Markopoulou, A., Iannaccone, G., Bhattacharyya, S., Chuah, C. N., Ganjali, Y., and Diot, C. Characterization of failures in an operational IP backbone network. *IEEE/ACM Transactions on Networking 16*, 4 (Aug. 2008), 749–762.

[18] Ruan, L., and Feng, T. A hybrid protection/restoration scheme for two-link failure in WDM mesh networks. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE* (2010), IEEE, pp. 1–5.

[19] Sharma, S., Staessens, D., Colle, D., Pickavet, M., and Demeester, P. Openflow: Meeting carrier-grade recovery requirements. *Computer Communications 36*, 6 (2013), 656–665.

[20] Takacs, L. On certain sojourn time problems in the theory of stochastic processes. *Acta Mathematica Hungarica 8* (1957), 169–191.

[21] Vasseur, J.-P., Pickavet, M., and Demeester, P. *Network Recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[22] Winkelmann, R. Duration dependence and dispersion in count-data models. *Journal of Business & Economic Statistics 13*, 4 (1995), pp. 467–474.

[23] Yallouz, J., and Orda, A. Tunable QoS-aware network survivability. *INFOCOM, 2013 Proceedings IEEE* (April 2013), 944–952.