Greening Reliability of Virtual Network Functions via Online Optimization

Xiaojun Shang, Yu Liu, Yingling Mao, Zhenhua Liu, Yuanyuan Yang
Stony Brook University, Stony Brook, NY 11794

***Abstract*—The fast development of virtual network functions (VNFs) brings new challenges to providing reliability. The widely adopted approach of deploying backups incurs financial costs and environmental impacts. On the other hand, the recent trend of incorporating renewable energy into computing systems provides great potentials, yet the volatility of renewable energy generation presents significant operational challenges. In this paper, we optimize availability of VNFs under a limited backup budget and renewable energy using a dynamic strategy GVB. GVB applies a novel online algorithm to solve the VNF reliability optimization problem with non-stationary energy generation and VNF failures. Both theoretical bound and extensive simulation results highlight that GVB provides higher reliability compared with existing baselines.**

I. Introduction

The emerging of network function virtualization allows software-based VNFs running on commodity servers to achieve the same functionality to that of middleboxes built on dedicated hardware. To realize the potential of VNFs in improving elasticity, flexibility, and scalability, much research has been conducted [1]–[6]. Despite potential benefits, software-based VNFs are more vulnerable to failures compared with traditional middleboxes. This is because that VNF platforms often have additional virtual layers between VNFs and the underlying hardware [7], [8]. Any misconfiguration of these layers may fail VNFs running over them. In addition, the reliability of commodity servers is inherently inferior to dedicated hardware made for middleboxes [9]. Therefore, guaranteeing availability of VNFs, which is essential to provide reliable network services, brings new challenges.

Deploying backups is a generic and robust method to improve availability of VNFs [9]–[13]. Yet, it improves availability at the expense of extra costs. Thus, service providers often limit the budget for deploying backups for profit consideration [14]. This is especially the case for systems with limited resources or energy, e.g., edge servers with small-scale CPU and memory [15], [16] or wireless devices with limited power supply [17]–[19]. Therefore, the dilemma between high availability of VNFs and the limited backup budget becomes an urgent and challenging problem.

Existing methods proposed to improve availability in previous work are insufficient due to the following two reasons. First, to our best knowledge, no previous work introduces mechanisms to leverage volatile renewable energy generation in real time to deploy VNF backups. Nowadays, both large data centers and edge servers start to take advantage of renewable energy for cost and greenhouse gas emission reduction [20]–[24]. Due to the volatility of renewable energy [25], it may not be reliable enough for applications that need stable energy supply. However, renewable energy is suitable for VNF backups which can scale up and down based on the sufficiency of the budget and operate only when the original ones fail. Thus, it is profitable for a VNF backup scheme to utilize renewable energy effectively if present. Second, it is essential for any backup strategy to consider the fact that failures are time-varying and hard to predict [26]. On one hand, offline VNF backup schemes giving a fixed number of backups may not work well with varying failure rates at different time slots. This is because, with a fixed budget, deploying more backups at a time with higher failure rates may achieve higher overall availability compared to distributing backups evenly over the time span. On the other hand, the hardness of predicting VNF failures becomes an obstacle for any online backup method relying on the prediction accuracy.

978-1-7281-6887-6/20/$31.00 ©2020 IEEE

Therefore, it is desirable for a scheme to deploy VNF backups in an online manner with full consideration of renewable energy and unpredictable VNF failures. Online mechanisms have been explored in related work for efficient VNF placement, e.g., [27], [28], or renewable energy utilization, e.g., [29]. As far as we are concerned, no online algorithm has been proposed to improve availability of VNFs utilizing renewable energy under non-stationary states.

In this paper, we propose a green energy aware VNF backup scheme GVB to improve availability of VNFs with a limited backup budget. GVB leverages renewable energy generated at each time slot for higher availability. It maintains a theoretical performance guarantee to the offline optimal solution with the presence of arbitrary green energy generation patterns and non-stationary VNF failure rates.

Our main contributions are summarized as follows.

- We design an efficient backup scheme GVB to improve availability of VNF systems with renewable energy and unpredictable VNF failures.
- We formulate the VNF backup deployment problem as an online integer optimization problem which is both computational complex and lacks future information. To solve this problem, we propose an online algorithm and prove its competitive ratio to the offline optimal solution.
- For GVB in the long run, we propose a parameter adjustment method to tune the online algorithm to achieve better performance than the theoretical bound.

- We conduct real-world trace-driven numerical simulations. The results highlight that our proposed scheme achieves higher availability of VNFs with the same backup budget compared to the baselines.

The remainder of this paper is organized as follows. Section II presents an overview of the related work and background. Section III formulates the online model of the problem while Section IV continues with the details of the online algorithm and the parameter adjustment method. Numerical results are presented and evaluated in Section V. Section VI concludes this paper and briefly describes future work.

II. Related Work and Background

Network function virtualization has the potential to reduce opening and operating costs, improve service agility and enhance utilization of existing network. It has drawn much research attention and many promising schemes have been proposed for various objectives. Recently, Zhang et al. propose an adaptive interference-aware heuristic approach to place VNFs in 5G service-customized network slices in [1]. Luo et al. propose a deep learning-based framework for scaling of the geo-distributed VNF chains in [2]. Kiji et al. construct a virtual network function placement and routing model for multicast service chaining based on merging multiple service paths in [3]. Bao et al. [4] investigate parallelism of VNFs for acceleration and propose a polynomial-time solution. Zheng et al. [5] present a two-stage optimization framework to minimize the cost while balancing the CPU processing capability. Considering fluctuations of SFC traffics and VNF migration costs, Shang et al. [6] propose an online joint SFC placement and routing scheme to reduce cost and network congestion.

Despite such benefits, the relatively complex configuration of a VNF platform may bring new challenges in guaranteeing availability of VNFs running over it. Taleb et al. [14] point out that VNFs are often more vulnerable to failures due to hardware failures, software bugs, hypervisor misconfigurations, and malicious attacks. Wang et al. [26] further illustrate that failures are not uniformly random at different time scales, and sometimes not even uniformly random at different spaces. Many failures are highly correlated with one another and failure spikes may happen at any time.

To protect VNFs from failures, Fan et al. [9] proposes schemes to minimize the number of backups with fixed availability requirements. Zhang et al. [13] further consider various resource demands of VNFs in the same problem and propose the corresponding solution. In scenarios where the backup budget is constrained due to resource or energy limitation, Kanizo et al. [11] use a backup-sharing method to maximize availability of VNFs with a prefixed number of backups. Dinh et al. [16] present a cost-efficient VNF backup scheme for VNFs deployed on the edge, which makes backup decisions based on evaluating availability improvement potential of each VNF. Taleb et al. [14] apply an early failure detection technology to place VNF backups in a reactive way. To save unnecessary costs, a VNF in their scheme is only backed up when its failure is detected in advance.

Instead of only saving backup resources for the most beneficial VNFs, GVB proposed in this paper introduces renewable energy generated over time to deploy more backups with the same budget. In addition, GVB is designed not only to handle the uncertainty of renewable energy but also to deal with non-stationary distribution of VNF failures. Methods in previous work either consider failure rates as stationary states or achieve the upcoming VNF failures using prediction mechanisms. However, as we mentioned in Section I, VNF failure distributions are often non-stationary and hard to predict. Any deviation of failure rates may result in unsatisfactory performance. Different from these methods, GVB formulates and solves an online integer optimization problem under non-stationary states, i.e., renewable energy generation and failure rates.

Online algorithms, e.g., [30]–[33], have been widely applied to solve problems in data centers. For online convex optimization problems with the presence of non-stationarity, Yu et al. [30] propose an algorithm with a theoretical performance guarantee which incorporates the estimations of average future states into the stochastic optimization framework for decision making. Lin et al. [31] propose an algorithmic framework called CR-Pursuit which preserves a competitive ratio of $\beta \cdot (\log(\theta) + 1)$, where β and θ are parameters depending on properties of the objective function. Comden et al. [32] studied how to jointly select prediction and control algorithms for online resource provisioning. Nevertheless, the formulated integer problem in this paper could not be solved directly using the proposed convex optimization methods. Thus, we propose a novel online algorithm solving the problem with a theoretical competitive ratio to the offline optimal solution.

III. Problem Formulation

In this section, we formulate and analyze the model of the green VNF backup problem dealt by GVB. Important notations used in this paper are summarized in Table I.

TABLE I

Notation	Definition
T	Time interval with a limited backup budget
C	Backup budget can be spent in time interval T
I	Set of VNFs in the system, $I = \{1, ..., t, ..., \|I\|\}$
α_i	Cost of deploying one backup for VNF i
$\sigma(t)$	Maximal backup cost can be saved by renewable energy at time t
$f_i(t)$	Failure rate of VNF i at time t
$x_i(t)$	Number of backups of VNF i deployed at time t
$\widehat{x}_i(t)$	Relaxed solution of $x_i(t)$
$\widehat{y}_i(t)$, $\widehat{z}_i(t)$	Determining variables for P2 and P3, the sub-problems of $\widehat{P1}$
G, g	Maximal and minimal gradient of the objective function of P1 at the time slot $t = 0$
θ	Ratio between G and g
p	Determining parameter of the online algorithm
L_i	Maximal number of backups of VNF i can be deployed at time t
C_t	Backup budget can be spent by the online algorithm at time t
B_t	Rounding budget can be spent by the rounding method at time t
r_t	Budget can be spent at time t inherited from the previous time slot $t-1$

Denote by C the maximal backup budget affordable by a VNF service provider during a time interval T. Here, C is given ahead of time and T is divided into multiple time slots, each of which is represented by t. Denote the set of different types of VNFs provided by the service provider by $I = \{1, ..., i, ..., |I|\}$. For each VNF i, denote by α_i the cost of deploying one backup for it in each time slot. α_i is the cost of computational resources and operating energy needed by the VNF backup within each time slot. To simplify our model without losing generality, we omit the opening cost of a VNF backup. We can do this because, with the rapid development of software-based VNF platforms built on virtual machines (VMs) or containers, e.g., ClickOS, new VNF instances can be booted within seconds. Moreover, since a VNF backup only works when the original one is down, the overhead of data transmission does not affect the performance of a backup in most cases. In this way, the opening cost of a backup is much smaller and negligible compared to its operating cost with long enough time slot t.

As mentioned in Section I, GVB considers renewable energy generated at each time slot t that can be consumed to deploy VNF backups. Denote by $\sigma(t)$ the maximal backup cost can be saved by utilizing renewable energy at time t. The weight of $\sigma(t)$ compared to the backup cost may vary in different cases and will be discussed in detail in Section V. Due to the uncertain generation pattern of renewable energy, any $\sigma(\tau)$ with $\tau > t$ is unknown at time t. Since green energy is often not stored for the next time slot, $\sigma(t)$ can only be consumed at t. Suppose the failure rate of each VNF is the same within one time slot. We denote the failure rate of VNF i at t by $f_i(t)$, which is non-stationary over time. Similar with $\sigma(t)$, all $f_i(\tau)$ are unknown with $\tau > t$. We then define an integer decision variable $x_i(t)$ representing how many backup copies are assigned to VNF i at time slot t. We thus maximize the total availability gained by VNFs backups within time interval T. The detailed problem denoted by P1 is formulated as follows.

$$\begin{aligned}\max_{x_i(t)} \quad & \sum_{t\in T}\sum_{i\in I} f_i(t)\cdot\Big(1-f_i(t)^{x_i(t)}\Big)\\ \text{s.t.} \quad & \sum_{t\in T}\left[\sum_{i\in I}\alpha_i\cdot x_i(t)-\sigma(t)\right]^+ \le C,\\ & x_i(t)\in\{0,1,2,...,L_i\},\ \forall t\in T, i\in I.\end{aligned} \tag{P1}$$

Availability of a VNF is the summation of its success rates at each time slot over time interval T. Thus, the objective function of P1 sums up availability gained by deploying all types of VNF backups at every time slot t. The first constraint makes sure that the total backup cost spent in T is within the prefixed budget C. Since the applied renewable energy can offset at most $\sigma(t)$ backup cost at time t, it is subtracted from the left side of the constraint. Due to the volatility of renewable energy, the constraint also makes sure that the unused part of $\sigma(t)$ is not inherited to future time slots. The second constraint ensures that the variable $x_i(t)$ is a non-negative integer constrained by L_i, since the number of backups deployed for VNF i cannot be fractional or exceed the maximal quantity affordable by the provider.

P1 is formulated as an online integer optimization problem which is computational complex and lacks future information. So it cannot be solved directly using offline algorithms. In the following section, we solve this problem using an online algorithm which reduces the computational complexity while preserving a competitive ratio to the offline optimal solution.

IV. Green VNF Backup Strategy

In this section, we present details of the GVB scheme. For time interval T with a fixed VNF backup budget C, we propose an online algorithm shown in Algorithm 1 which decides the number of backups for each VNF at every time slot t. We prove the competitive ratio of Algorithm 1 to the offline optimal solution which knows all information in advance, i.e., VNF failure rates and available renewable energy at every t.

When the total running time is longer than T, it is common that the available backup cost is replenished to C at the end of each T. For instance, the cost budget of a network service is renewed at the start of a new payment cycle and the battery of a wireless device is recharged when used up. In these cases, we further propose a parameter adjustment method in GVB to improve its performance in the long run.

A. Online Algorithm Design

The theoretical challenges of the problem formulated in Section III mainly come from online optimization and integral variables. In Algorithm 1, we first convert the problem P1 into an online convex optimization problem $\widehat{\text{P1}}$ by relaxing the decision variable from $x_i(t) \in \{0, 1, 2, ..., L_i\}$ to $\widehat{x}_i(t) \in [0, L_i]$. We first solve $\widehat{\text{P1}}$ using a novel online convex optimization algorithm with a smaller competitive ratio compared with existing related work. Based on the solution of $\widehat{\text{P1}}$, we then design a tailor-made rounding method to get integral solutions for P1 which preserves a competitive ratio to the optimal integral solution.

To solve $\widehat{\text{P1}}$, Algorithm 1 first divides the problem into two sub-problems P2 and P3 with decision variables $\widehat{y}_i(t) \in [0, L_i]$, $\widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)]$ and $\widehat{x}_i(t) = \widehat{y}_i(t) + \widehat{z}_i(t)$. Since $\widehat{\text{P1}}$, P2 and P3 are all concave and non-decreasing, it is clear that the optimal solution of $\widehat{\text{P1}}$ is achieved when P2 and P3 reach their maximal objective values.

$$\begin{aligned}\max_{\widehat{y}_i(t)} \quad & \sum_{t\in T}\sum_{i\in I} f_i(t)\cdot\Big(1-f_i(t)^{\widehat{y}_i(t)}\Big)\\ \text{s.t.} \quad & \sum_{i\in I}\alpha_i\cdot\widehat{y}_i(t)\le\sigma(t),\ \forall t\in T,\\ & \widehat{y}_i(t)\in[0,L_i],\ \forall t\in T, i\in I.\end{aligned} \tag{P2}$$

$$\begin{aligned}\max_{z_i(t)} \quad & \sum_{t\in T}\sum_{i\in I} f_i(t)^{1+\widehat{y}_i(t)}\cdot\Big(1-f_i(t)^{\widehat{z}_i(t)}\Big)\\ \text{s.t.} \quad & \sum_{t\in T}\sum_{i\in I}\alpha_i\cdot\widehat{z}_i(t)\le C,\\ & \widehat{z}_i(t)\in[0,L_i-\widehat{y}_i(t)],\ \forall t\in T, i\in I.\end{aligned} \tag{P3}$$

Here, P2 is solved before P3. Since $\sigma(t)$ is the backup cost saved by utilizing renewable energy and is only consumable in time t, the optimal solution of P2 can be achieved by solving an offline convex optimization problem at each time slot t. In this paper, we apply the convex problem solver CVXPY in [34] to solve P2. With $\widehat{y}_i(t)$ achieved, solving the online optimization problem $\widehat{\text{P1}}$ is thus converted to solving the online problem P3. Algorithm 1 solves P3 by formulating and solving the problem P4 at each t as follows.

$$\begin{aligned} \min_{\widehat{z}_i(t)} \quad & \sum_{i \in I} \alpha_i \cdot \widehat{z}_i(t) \\ \text{s.t.} \quad & \sum_{i \in I} f_i(t)^{1+\widehat{y}_i(t)} \cdot \left(1 - f_i(t)^{\widehat{z}_i(t)}\right) \\ & \geq \frac{1}{p} \cdot \left(\eta_{opt}(t) - \eta_{opt}(t-1)\right), \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)], \ \forall i \in I. \end{aligned} \tag{P4}$$

Here, $\eta_{opt}(t)$ is the optimal objective value of the following problem P5 and p is the key parameter affecting the performance of our algorithm which will be discussed in detail in the following sections.

$$\begin{aligned} \max_{\widehat{z}_i(\tau)} \quad & \sum_{\tau=1}^{t} \sum_{i \in I} f_i(\tau)^{1+\widehat{y}_i(\tau)} \cdot \left(1 - f_i(\tau)^{\widehat{z}_i(\tau)}\right) \\ \text{s.t.} \quad & \sum_{\tau=1}^{t} \sum_{i \in I} \alpha_i \cdot \widehat{z}_i(\tau) \leq C, \\ & \widehat{z}_i(\tau) \in [0, L_i - \widehat{y}_i(\tau)], \ \forall i \in I, \tau \in [0, t]. \end{aligned} \tag{P5}$$

Algorithm 1 Online Green VNF Backup Algorithm

```
Relax the problem P1 to the problem P1^, r_0 = 0.
for all t ∈ T do
    Solve the offline problem P2 at t to get y^_i(t).
    Solve P5 to get η^t_opt.
    Solve P4 to get z^_i(t) and further get x^_i(t) = y^_i(t) + z^_i(t).
    Calculate C_t = r_t + Σ_{i∈I} α_i · x^_i(t).
    Calculate B_t = C_t − Σ_{i∈I} α_i · ⌊x^_i(t)⌋.
    Sort I in the decreasing order of α_i to get I'.
    for all i ∈ I' do
        if B_t − α_i ≥ 0 then
            B_t = B_t − α_i, x_i(t) = ⌈x^_i(t)⌉
        else
            x_i(t) = ⌊x^_i(t)⌋
        end if
    end for
    r_{t+1} = B_t
end for
```

At each time slot t, with $\widehat{z}_i(t)$ from P4 and $\widehat{y}_i(t)$ from P2, we can get $\widehat{x}_i(t) = \widehat{y}_i(t) + \widehat{z}_i(t)$ which is a feasible solution of $\widehat{\text{P1}}$. Algorithm 1 further determines integer solution $x_i(t)$ of P1 based on $\widehat{x}_i(t)$ using a novel rounding method.

At each time slot t, we calculate the backup cost can be spent by Algorithm 1 at this time slot, $C_t = r_t + \sum_{i \in I} \alpha_i \cdot \widehat{x}_i(t)$. It consists of $\sum_{i \in I} \alpha_i \cdot \widehat{x}_i(t)$ (the backup cost spent by the solution of $\widehat{\text{P1}}$) and r_t (the unused rounding budget from the previous time slot). We denote the smallest (largest) integer larger (smaller) than $\{\widehat{x}_i(t)\}$ by $\lceil \widehat{x}_i(t) \rceil$ ($\lfloor \widehat{x}_i(t) \rfloor$, respectively). We further denote by $B_t = C_t - \sum_{i \in I} \alpha_i \cdot \lfloor \widehat{x}_i(t) \rfloor$ the rounding budget at time t. The sum of all backup cost caused by rounding $\widehat{x}_i(t)$ to $\lceil \widehat{x}_i(t) \rceil$ should not exceed B_t.

With B_t, we sort VNFs in I in the decreasing order of a_i. Starting from the i with the largest a_i, if $B_t - \alpha_i \geq 0$, $\widehat{x}_i(t)$ is rounded up that $x_i(t) = \lceil \widehat{x}_i(t) \rceil$ and $B_t = B_t - \alpha_i$. If $B_t - \alpha_i < 0$, $x_i(t) = \lfloor \widehat{x}_i(t) \rfloor$. The above step is repeated until I is traversed. At this time, if $B_t > 0$, there exists unused rounding budget. To improve the performance, this remaining budget is inherited to the following time slot that $r_{t+1} = B_t$. At any time slot t, the number of backups to be deployed for each VNF is thus determined after the rounding method. We then prove the competitive ratio of Algorithm 1 in the next section.

B. Theoretical Analysis of the Online Algorithm

We now explain the details of how we achieve the competitive ratio of Algorithm 1 to the offline optimal solution. Since the optimal solution of P2 can be achieved by solving $|T|$ offline optimization problems, the performance of Algorithm 1 in solving the relaxed problem $\widehat{\text{P1}}$ is determined by the performance of solving the online convex optimization problem P3. We denote the part of Algorithm 1 solving P3 by $SV(p)$ which is determined by a parameter p. The solution of $SV(p)$ is feasible if $\sum_{t \in T} \sum_{i \in I} \alpha_i \cdot \widehat{z}_i(t) \leq C$. Denote the objective value of applying $SV(p)$ (the optimal objective value of P3) by η^{SV} (η^{OPT}, respectively). According to the first constraint of P4, if $SV(p)$ gives a feasible solution, $\eta^{SV} \geq \sum_{t \in T} \frac{1}{p} \cdot (\eta_{opt}(t) - \eta_{opt}(t-1)) = \frac{1}{p} \cdot \eta^{OPT}$. Thus, $\frac{1}{p}$ is the competitive ratio between $SV(p)$ and the offline optimum of P3. We then need to determine the range of p for $SV(p)$ to give feasible solutions.

At any time slot t, we first create a set of functions $\{M_t(w) | w \geq 0\}$. $M_t(w)$ is the optimal objective value of the following problem P6.

$$\begin{aligned} \max_{\widehat{z}_i(t)} \quad & \frac{1}{|I|} \cdot \sum_{i \in I} f_i(t)^{1+\widehat{y}_i(t)} \cdot \left(1 - f_i(t)^{\widehat{z}_i(t)}\right) \\ \text{s.t.} \quad & \sum_{i \in I} \alpha_i \cdot \widehat{z}_i(t) \leq |I| \cdot w \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)] \ \forall i \in I \end{aligned} \tag{P6}$$

We first present some important properties of $M_t(w)$ in Lemma 1.

Lemma 1. *$M_t(w)$ for any t is a concave, continuous, increasing, and piecewise differentiable function. In addition, $M_t(0) = 0$ and $M_t \triangleq \lim_{w \to 0^+} \frac{M_t(w)}{w} \in [g, G]$.*

The detailed proof of Lemma 1 is omitted here due to the space limitation. With $M_t(w)$ formulated, we now construct an entire new problem P7 similar to P3 as follows.

$$\begin{aligned} \max_{\widehat{z}_i(t)} \quad & \sum_{t\in T}\sum_{i\in I} M_t(\alpha_i \cdot \widehat{z}_i(t)) \\ \text{s.t.} \quad & \sum_{t\in T}\sum_{i\in I} \alpha_i \cdot \widehat{z}_i(t) \le C, \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)],\ \forall i \in I, t \in T. \end{aligned} \tag{P7}$$

For the formulated P7, we apply a new algorithm $SV^*(p)$ to solve it. The first step is to solve an optimization problem at each time slot t as follows.

$$\begin{aligned} \min_{\widehat{z}_i(t)} \quad & \sum_{i\in I} \alpha_i \cdot \widehat{z}_i(t) \\ \text{s.t.} \quad & \sum_{i\in I} M_t(\alpha_i \cdot \widehat{z}_i(t)) = \frac{1}{p}\cdot\left(\eta^*_{opt}(t) - \eta^*_{opt}(t-1)\right), \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)],\ \forall i \in I. \end{aligned} \tag{P8}$$

Here, $\eta^*_{opt}(t)$ is the optimal objective value of the following Problem P9.

$$\begin{aligned} \max_{\widehat{z}_i(\tau)} \quad & \sum_{\tau=1}^{t}\sum_{i\in I} M_t(\alpha_i \cdot \widehat{z}_i(\tau)) \\ \text{s.t.} \quad & \sum_{\tau=1}^{t}\sum_{i\in I} \alpha_i \cdot \widehat{z}_i(\tau) \le C, \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(\tau)],\ \forall i \in I, \tau \in [0,t]. \end{aligned} \tag{P9}$$

Denote the optimal solution of P8 at time slot t by $\widehat{z}^*_i(t)$, which is chosen as the output of $SV^*(p)$ for time t. It is clear that if $\sum\limits_{t\in T}\sum\limits_{i\in I} \alpha_i \cdot \widehat{z}^*_i(t) \le C$, the solution of $SV^*(p)$ is feasible for P7. We then introduce Lemma 2 which relates the newly constructed problem P7 to P3 which we actually need to solve.

Lemma 2. *For a specific p, if the solution of $SV^*(p)$ is feasible for P7, the solution of $SV(p)$ is feasible for P3.*

Proof. We first prove that $\eta_{opt}(t) = \eta^*_{opt}(t), \forall t \in T$. Suppose $\widehat{z}_i(\tau)^{P5}_{opt}$ is the optimal solution of P5. We define $\widehat{z}_i(\tau)^{P9} = \frac{1}{|I|}\cdot\sum_{i\in I}\widehat{z}_i(\tau)^{P5}_{opt}$, which is a feasible solution of P9. Based on Lemma 1, we have $\sum_{i\in I} M_t(\alpha_i \cdot \widehat{z}_i(\tau)^{P9}) \ge \sum\limits_{i\in I} f_i(\tau)^{1+\widehat{y}_i(\tau)} \cdot (1 - f_i(\tau)^{\widehat{z}_i(\tau)^{P5}_{opt}}), \forall \tau \in [0,t]$. Summing up all τ in $[0,t]$, we further have $\eta^*_{opt}(t) \ge \eta_{opt}(t)$. We then suppose $\widehat{z}_i(\tau)^{P9}_{opt}$ is the optimal solution of P9. According to the definition of M_t, these exist $\widehat{z}_i(\tau)^{P5}, i \in I, t \in [0,t]$. which makes $\sum_{i\in I} M_t(\alpha_i \cdot \widehat{z}_i(\tau)^{P9}_{opt}) = \sum\limits_{i\in I} f_i(\tau)^{1+\widehat{y}_i(\tau)} \cdot (1 - f_i(\tau)^{\widehat{z}_i(\tau)^{P5}}), \forall \tau \in [0,t]$, under the condition that $\sum_{i\in I}\alpha_i \cdot \widehat{z}_i(\tau)^{P5} \le \sum_{i\in I}\alpha_i\cdot\widehat{z}_i(\tau)^{P9}_{opt}$. Sum the equality up in $[0.t]$ and we get $\eta^*_{opt}(t) = \sum\limits_{\tau=1}^{t}\sum\limits_{i\in I}(1 - f_i(\tau)^{\widehat{z}_i(\tau)^{P5}})$. In this way, based on this equation and previous results, we can conclude that $\eta_{opt}(t) \ge \eta^*_{opt}(t)$. Since $\eta_{opt}(t) \ge \eta^*_{opt}(t)$ and $\eta^*_{opt}(t) \ge \eta_{opt}(t+1)$, we have $\eta_{opt}(t) = \eta^*_{opt}(t)$.

Denote the backup cost $SV(p)$ ($SV^*(p)$) spends at time t by $K_t = \sum_{i\in I}\alpha_i\widehat{z}_i(t)$ ($K^*_t = \sum_{i\in I}\alpha_i\widehat{z}^*_i(t)$, respectively). Based on the formulation of M_t, availability gained at time t of P7 is the objective value of the problem P10 formulated as follows.

$$\begin{aligned} \max_{\widehat{z}_i(t)} \quad & \sum_{i\in I} f_i(t)^{1+\widehat{y}_i(t)} \cdot \left(1 - f_i(t)^{\widehat{z}_i(t)}\right) \\ \text{s.t.} \quad & \sum_{i\in I} \alpha_i \cdot \widehat{z}_i(t) \le K^*_t, \\ & \widehat{z}_i(t) \in [0, L_i - \widehat{y}_i(t)]\ \forall i \in I. \end{aligned} \tag{P10}$$

We then prove that $K_t = K^*_t$. On one hand, if $K^*_t < K_t$, there exist $\widehat{z}_i(t)$ that $\sum\limits_{i\in I}\alpha_i \cdot \widehat{z}_i(t) = K^*_t < K_t = \sum\limits_{i\in I}\alpha_i \cdot \widehat{z}_i(t)_{opt}$ and $\sum\limits_{i\in I} f_i(t)^{1+\widehat{y}_i(t)} \cdot \left(1 - f_i(t)^{\widehat{z}_i}(t)\right) \ge \frac{1}{p}\cdot\left(\eta^*_{opt}(t) - \eta^*_{opt}(t-1)\right) = \frac{1}{p}\cdot(\eta_{opt}(t) - \eta_{opt}(t-1))$. This conclusion is contradict to that $\widehat{z}_i(t)_{opt}$ is the optimal solution of P4. On the other hand, if $K^*_t > K_t$, we formulate $\alpha_i \cdot \widehat{z}_i(t) = \frac{1}{|I|}\cdot\sum\limits_{i\in I}\alpha_i\cdot\widehat{z}_i(t)_{opt}$. It is obvious that $\sum\limits_{i\in I}\alpha_i\cdot\widehat{x}_i(t) = K_t < K^*_t = \sum\limits_{i\in I}\alpha_i\cdot\widehat{x}^*_i(t)_{opt}$. Based on the formulation of M_t and $\eta_{opt}(t) = \eta^*_{opt}(t)$, we further have $\sum\limits_{i\in I} M_t(\alpha_i\cdot\widehat{z}_i(t)) \ge \frac{1}{p}\cdot(\eta^*_{opt}(t) - \eta_{opt}(t)) = \frac{1}{p}\cdot(\eta^*_{opt}(t) - \eta^*_{opt}(t)) = \sum\limits_{i\in I} M_t(\alpha_i\cdot\widehat{z}^*_i(t)_{opt})$. This conclusion is contradict with that $\widehat{z}^*_i(t)$ is the optimal solution of P8. Therefore, $K_t = K^*_t$ and if $SV^*(p)$ is feasible for P7, $SV(p)$ is feasible for P3. $\square$

With Lemma 2 proved, we now take advantage of P7, $SV^*(p)$ and the CR-Pursuit(p) algorithm in [31] to find the region of p. We suppose the solution of P7 using CR-Pursuit(p) is $\widehat{z}_i(t)_{CR}$. We then have the following Lemma 3, 4 and 5.

Lemma 3. *For a specific p, if the solution of CR-Pursuit(p) is feasible for P7, the solution of $SV^*(p)$ is feasible for P7.*

Lemma 4. *When applying CR-Pursuit(p) to P7, we have*

$$\alpha_i \cdot \widehat{z}_i(t)_{CR} \le \beta \cdot \frac{M_t(\alpha_i \cdot \widehat{z}_i(t)_{CR})}{u(t,i)}, \forall t \in T, \forall i \in I$$

In Lemma 4, $\beta = \max\limits_{f(t)\in F}\frac{s(t,i)}{f_i(t)^{1+\widehat{y}_i(t)}\cdot(1-f(t)^{L_i(t)})/L_i(t)}$, where $s(t,i) = \lim_{\widehat{z}_i(t)\to 0^+}\frac{f_i(t)^{1+\widehat{y}_i(t)}\cdot(1-f_i(t)^{\widehat{z}_i(t)})}{\alpha_i\cdot\widehat{z}_i(t)} \in [g,G]$. $u(t,i) = \lim_{\widehat{z}_i(t)\to 0^+}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t))}{\alpha_i\cdot\widehat{z}_i(t)} \in [g,G]$.

Lemma 5. *When applying CR-Pursuit(p) to P7, for any threshold $s \in [g,G]$, we have*

$$\sum_{\{t,i:u(t,i)\le s\}} M_t(\alpha_i \cdot \widehat{z}_i(t)_{CR}) \le \frac{s}{p}\cdot C.$$

We can easily get that the budget spent by CR-Pursuit(p) at any t is always larger than that of $SV^*(p)$. So Lemma 3 is easily proved. The proof of Lemma 4 and 5 is similar to that of Lemma 9 and Lemma 10 in [31]. Due to space limitations, we omit the detailed proof of these lemmas here. Based on Lemma 4 and 5, we can conclude an upper bound of the backup cost spent by CR-Pursuit(p) for P7, which is shown in Lemma 6.

Lemma 6. *When applying CR-Pursuit(p) to P7, we have*

$$\sum_{t\in T}\sum_{i\in I}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}\le$$

$$\frac{C}{p}\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right).$$

Proof. Based on the formulation of M_t, all $u(t,i)$ with the same t are equal for different $i\in I$. We assume they take H different values, that are $S_1,...,S_h,...,S_H$ in an increasing order. We further define $v_h=\sum_{\{t,i:u(t,i)=S_h\}}\frac{p}{C}\cdot M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})$ and have the equation as follows.

$$\sum_{t\in T}\sum_{i\in I}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}=\frac{C}{p}\cdot\sum_{h=1}^{H}\frac{v_h}{S_h}$$

.

Considering the following problem P11,

$$\begin{aligned}\max_{v_h}\quad&\sum_{h=1}^{H}\frac{v_h}{S_h}\\ \text{s.t.}\quad&\sum_{j\in[1,h]}v_j\le S_h,\forall h\in[1,H],\\ &v_h\ge 0,\forall h\in[1,H].\end{aligned}\tag{P11}$$

According to Lemma 11 in [31], the objective value of P11 is $\sum_{h=1}^{H}\frac{S_h-S_{h-1}}{S_h}$, Which means

$$\sum_{t\in T}\sum_{i\in I}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}=\frac{C}{p}\cdot\sum_{h=1}^{H}\frac{v_h}{S_h}$$

$$\le\frac{C}{p}\cdot\sum_{h=1}^{H}\frac{S_h-S_{h-1}}{S_h}$$

According to the settings in Lemma 4, we have $g<S_h<G,\forall h\in[1,H]$. We now consider P12 as follows.

$$\max_{g<S_1<...<S_h<...S_H<G}\quad\sum_{h=1}^{H}\frac{S_h-S_{h-1}}{S_h}\tag{P12}$$

This problem is the same as

$$\max_{g<S_1<...<S_h<...S_H<G}\quad H-\sum_{h=2}^{H}\frac{S_h-S_{h-1}}{S_h}\tag{P13}$$

By applying the inequality of arithmetic and geometric means to it, we have

$$Value(P13)\le H-(H-1)\cdot(\frac{S_1}{S_2}\cdot\frac{S_2}{S_3}...\frac{S_{H-1}}{S_H})^{1/(H-1)}$$

The equality stands only when $\frac{S_1}{S_2}=\frac{S_2}{S_3}=...=\frac{S_{H-1}}{S_H}$. In this way, we can get that the optimal objective value of P12 is $H-(H-1)\cdot(\frac{g}{G})^{1/(H-1)}=H-(H-1)\cdot(1-\theta^{-1/(H-1)}$. It is clear that $H\le|T|$, so we have $\sum_{t\in T}\sum_{i\in I}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}\le\frac{C}{p}\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right)$ and Lemma 6 is thus proved. ☐

With the above lemmas and conclusions, we illustrate the performance of $SV(p)$ by the following lemma.

Lemma 7. *The competitive ratio of $SV(p)$ for P3 is $\frac{1}{p_0}$ to the offline optimum, where $p_0=\beta\cdot(1+(|T|-1)(1-\theta^{-\frac{1}{|T|-1}}))$.*

Proof. According to Lemma 4, we have $\sum_{t\in T}\sum_{i\in I}\alpha_i\cdot\widehat{z}_i(t)_{CR}\le\sum_{t\in T}\sum_{i\in I}\beta\cdot\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}$. According to Lemma 6, we have $\sum_{t\in T}\sum_{i\in I}\frac{M_t(\alpha_i\cdot\widehat{z}_i(t)_{CR})}{u(t,i)}\le\frac{C}{p}\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right)$. In this way, we have $\sum_{t\in T}\sum_{i\in I}\alpha_i\cdot\widehat{z}_i(t)_{CR}\le\beta\cdot\frac{C}{p}\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right)$. Therefore, when $p\ge\beta\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right)$, $\sum_{t\in T}\sum_{i\in I}\alpha_i\cdot\widehat{z}_i(t)_{CR}\le C$. Then CR-Pursuit($p$) is feasible for P7. Then, based on Lemma 2 and 3, we finally have $SV(p)$ is feasible for P3. ☐

With Lemma 7 proved, we can achieve the distance from the online solution of $\widehat{\text{P1}}$ to the offline optimal solution of P1. In the following theorem, we denote the offline optimal solution (the solution of Algorithm 1) of a problem P by $OPT(P)$ (ALG1(P), respectively).

Theorem 1. *The competitive ratio between $ALG1(\widehat{P1})$ and $OPT(P1)$ is $\frac{1}{p_0}$, where $p_0=\beta\cdot(1+(|T|-1)(1-\theta^{-\frac{1}{|T|-1}}))$.*

Proof. As mentioned in Section IV-A, the objective value of $\widehat{\text{P1}}$ is the summation of objective values of P2 and P3. We also know that $OPT(P2)$ can be achieved by solving an offline optimization problem at each t. Therefore, we have $\frac{ALG1(\widehat{P1})}{OPT(P1)}\ge\frac{ALG1(\widehat{P1})}{OPT(\widehat{P1})}\ge\frac{1+OPT(P2)/ALG1(P3)}{p_0+OPT(P2)/ALG1(P3)}\ge\frac{1}{p_0}$. ☐

We claim that the competitive ratio achieved in Theorem 1 is strictly smaller than $\mathcal{O}(\log(\theta))$ in the CR-Pursuit algorithm.

Proof. We know that $\theta^k>1+k\cdot\log(\theta)$ with $|k|<1$. Since $|T|>1$, we have $1-\theta^{-\frac{1}{|T|-1}}<1-1+\frac{1}{|T|-1}\log(\theta)$. Then, $(|T|-1)\cdot(1-\theta^{-\frac{1}{|T|-1}})<\log(\theta)$ and we thus have $\beta\cdot(\log(\theta)+1)>\beta\cdot\left(1+(|T|-1)\left(1-\theta^{-\frac{1}{|T|-1}}\right)\right)$. ☐

We then focus on the distance from the integer solution of the rounding method to the online solution of $\widehat{\text{P1}}$, i.e., $\frac{ALG1(P1)}{ALG1(\widehat{P1})}$. To get this ratio, we need Lemma 8 first.

Lemma 8. *After applying the rounding method at each time t, we have $\sum_{i\in I}\alpha_i\cdot x_i(t)-\sum_{i\in I}\alpha_i\cdot\lfloor\widehat{x}_i(t)\rfloor\geq\frac{B_t}{2}$, as long as $B_t\geq\max_i\{\alpha_i\}$.*

Proof. Assume VNF j is a VNF which is rounded to $\lceil\widehat{x}_j(t)\rceil$ and the cost of its following VNF $j+1$ exceeds the remaining backup cost. $j+1$ always exists, otherwise all $\widehat{x}_j(t)$ can be rounded up and $B_t=0$. It is clear that $\alpha_j\leq B_t-B_t'$, where B_t' is the remaining backup cost after the rounding method. If $B_t'>\frac{B_t}{2}$, we can get that $\widehat{x}_{j+1}(t)$ with $\alpha_{j+1}\leq\alpha_j$ will have $\alpha_{j+1}>\frac{B_t}{2}$, since a_{j+1} exceeds the remaining budget larger or equal to B_t'. However, we then have $\alpha_{j+1}>\frac{B_t}{2}\geq\alpha_j$, Which is contradict with the assumption that $\alpha_{j+1}\leq\alpha_j$. Therefore, we have $B_t'\leq\frac{B_t}{2}$ and thus $\sum_{i\in I}\alpha_i\cdot\widehat{x}_i(t)-\sum_{i\in I}\alpha_i\cdot\lfloor\widehat{x}_i(t)\rfloor\geq\frac{B_t}{2}$. □

For simplicity of the proof, we suppose $\bar{x}_i(t)=x_i(t)-\lfloor\widehat{x}_i(t)\rfloor$ and $\tilde{x}_i(t)=\widehat{x}_i(t)-\lfloor\widehat{x}_i(t)\rfloor$. At each time slot t, we also suppose the objective value of $ALG1(P1)$ ($ALG1(\widehat{P1})$) is $ALG1(P1)_t$ ($ALG1(\widehat{P1})_t$, respectively). With Lemma 8, we then claim the following lemma which presents the bound between $ALG1(P1)_t$ and $ALG1(\widehat{P1})_t$.

Lemma 9. *We have $\frac{ALG1(P1)_t}{ALG1(\widehat{P1})_t}\geq\frac{Q_1(t)}{2Q_2(t)}$, where $Q_1(t)=\min_i\{\frac{f_i(t)-\gamma_i(t)}{\alpha_i}\}$, $Q_2(t)=\max_i\{\frac{f_i(t)-\gamma_i(t)\cdot f_i(t)^{\tilde{x}_i(t)}}{\alpha_i\cdot\tilde{x}_i(t)}\}$ and $\gamma_i(t)=f_i(t)^{1+\lfloor\widehat{x}_i(t)\rfloor}$.*

Proof. It is clear that we have $f_i(t)-\gamma_i(t)\cdot f_i(t)^{\bar{x}_i(t)}\geq Q_1(t)\cdot\alpha_i\cdot\bar{x}_i(t)$. In this way, we have $\sum_i\left(f_i(t)-\gamma_i(t)\cdot f_i(t)^{\bar{x}_i(t)}\right)\geq Q_1(t)\cdot\sum_i\alpha_i\cdot\bar{x}_i(t)$. According to Lemma 8, we have $\sum_i\alpha_i\cdot\bar{x}_i(t)\geq\frac{B_t}{2}$. In this way, we further have $\sum_i\left(f_i(t)-\gamma_i(t)\cdot f_i(t)^{\bar{x}_i(t)}\right)\geq\frac{Q_1(t)\cdot B_t}{2}$. We also have that $\sum_i(f_i(t)-f_i(t)^{1+\widehat{x}_i(t)})\leq Q_2(t)\cdot\sum_i\alpha_i\cdot\tilde{x}_i(t)\leq Q_2(t)\cdot B_t$. In this way, we can conclude that $\frac{ALG1(P1)_t}{ALG1(\widehat{P1})_t}\geq\frac{Q_1(t)}{2Q_2(t)}$. □

The assumption of Lemma 8 is often easily satisfied, since there are multiple types of VNFs in the system and the rounding budget is often larger than the cost of one VNF backup. When the assumption is not satisfied, we have $B_t<\max_i\{\alpha_i\}$. Since the backup cost of different VNF is within the same scale, B_t is rather small within the cost of one backup copy. Then the ratio between $ALG1(P1)_t$ and $ALG1(\widehat{P1})_t$ is very close to 1. Therefore, the conclusion of Lemma 9 stands in most cases. With the conclusion of Lemma 9 and Theorem 1, we can finally prove the competitive ratio between $ALG1(P1)$ and $OPT(P1)$ in the following theorem.

Theorem 2. *We have $\frac{ALG1(P1)}{OPT(P1)}\geq\frac{Q_{1min}}{2Q_{2max}\cdot P_0}$. Here, $Q_{1min}=\min_{t\in T}\{Q_1(t)\}$, $Q_{2max}=\max_{t\in T}\{Q_2(t)\}$ and $p_0=\beta\cdot(1+(|T|-1)(1-\theta^{-\frac{1}{|T|-1}}))$.*

Proof. Based on Lemma 9, we can easily get the conlusion that $\frac{ALG1(P1)}{ALG1(\widehat{P1})}=\frac{\sum_{t\in T}ALG1(P1)_t}{\sum_{t\in T}ALG1(\widehat{P1})_t}\geq\frac{Q_{1min}}{2Q_{2max}}$. Knowing $\frac{ALG1(\widehat{P1})}{OPT(P1)}\geq\frac{1}{p_0}$ from Theorem 1, the conclusion is thus obvious. □

With Theorem 2 proved, we know the theoretical competitive ratio of our GVB scheme to the offline optimal solution in any time interval T with a fixed backup budget C. The parameter p_0 can be calculated with detailed β and θ. However, we can pick smaller p for Algorithm 1 to improve the performance while still satisfying the budget constraint. In the following section, we illustrate this in detail and propose a novel p adjustment method for better performance of GVB in the long run.

C. Parameter Adjustment for Better Performance

As discussed in Section IV-A, the rounding method in Algorithm 1 may not spend all of the budget, $C_t=r_t+\sum_{i\in I}\alpha_i\cdot\widehat{x}_i(t)$, at each time slot t. So we can be more aggressive and pick $p\leq p_0$ to further improve the performance of GVB without violating the budget. When GVB is applied over a long time span containing multiple budget period T, we can decide p for the current period based on performance and remaining budgets of previous periods through methods such as online learning. In this paper, we apply the most straightforward strategy which starts from p_0 and makes $p=p-\delta$ if there is a remaining budget and $p=p+\delta$ if there is a deficit. The value of δ is tuned based on the proportion of the remaining budget after each budget period T. Simulation results in Section V shows that the adjustment method can achieve higher availability compared to the baselines in long time spans.

V. Performance Evaluation

A. Simulation Setup

For the simulation setup, we consider the VNF set I contains 20 different types of VNFs and each VNF can maintain at most four backups at one time. Since the backup cost of different VNF may be different due to multiple factors, we suppose the relative operating cost of a VNF varies in the range $[1,5]$. We suppose the time interval T is evenly divided into 40 time slots. For the distribution of renewable energy, we apply the real-world trace of solar energy distribution in [35]. The settings of other parameters, e.g., C and $f_i(t)$, vary in different simulations and will be illustrated in detail in the following sections.

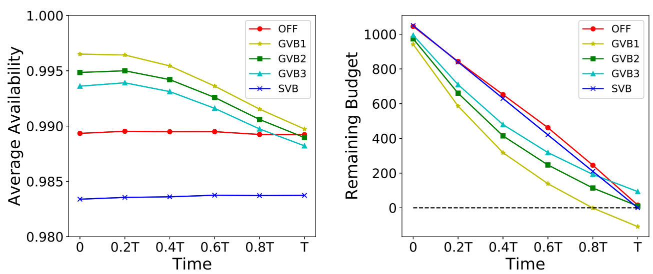
B. Baselines

Since GVB is an online scheme, we set an offline solution of the online integer optimization problem as a baseline. In the offline solution, we first solve the convex optimization problem $\widehat{\mathrm{P1}}$ directly with all failure rates and renewable energy known in advance. We then apply the rounding method in Algorithm 1 to get integral solutions. We are doing this because it is computationally inefficient to solve the original integer problem P1 directly.

To evaluate the advantages of GVB by considering renewable energy and non-stationary failure rates, we also compare it with an ideal static VNF backup scheme (SVB) which knows the accurate average failure rate of each VNF over T. Since SVB does not consider the online patterns of renewable energy and failure rates, the number of backups for each VNF is the same at different time slots. Similar to the offline solution, We first solve $\widehat{\text{P1}}$ with accurately predicted average failure rates, then apply the rounding method in Algorithm 1 to get integral solutions.

C. Performance of GVB

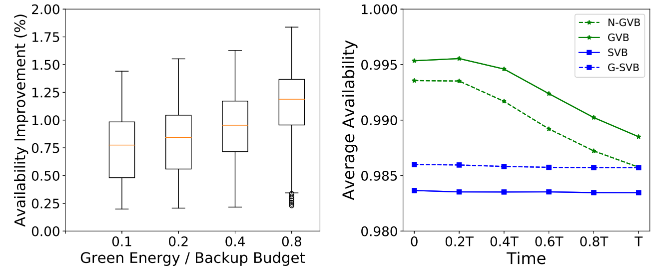
In this section, we evaluate the performance of GVB under the influences of different factors. All results presented are the average value of 50 independent simulations. In Fig. 1, we present average VNF availability achieved by GVB with different p and compare it with the offline solution and SVB. Here, the backup budget C is set to 1200, which is around half of the total energy cost to provide every VNF one backup copy at each time slot. The total backup cost saved by renewable energy over T is 30% of the backup budget. The VNF failure rates are uniformly distributed in $[0, 5\%]$.



(a) (b)

Fig. 1. Performance of the offline solution, GVB with different p and SVB when time increases from 0 to T. (a) Average availability of VNFs achieved by the offline solution, GVB1, GVB2, GVB3 with $p = 1.04, 1.06, 1.08$ and SVB. (b) The average remaining budget of each algorithm at different time.

Fig. 1 (a) shows average availability of all VNFs from the offline solution, SVB with different p, and the baseline GVB at different time slots. We observe that the value of p affects the performance of Algorithm 1 directly and smaller p leads to higher availability. Combining Fig. 1 (b) which shows the corresponding remaining budget, we learn that Algoirhtm 1 in GVB is feasible when $p \geq 1.06$. When Algorithm 1 is feasible, average availability of it is much higher than that of SVB over the entire time span. Furthermore, we also observe that GVB outperforms the offline solution for more than 90% of the time. Although inferior at time $t = T$, final average availability of GVB is still very close to the offline solution. In this way, we can conclude that, in general cases, the VNF availability provided by the GVB scheme is much higher than that of the baseline scheme SVB and very close to the offline solution.

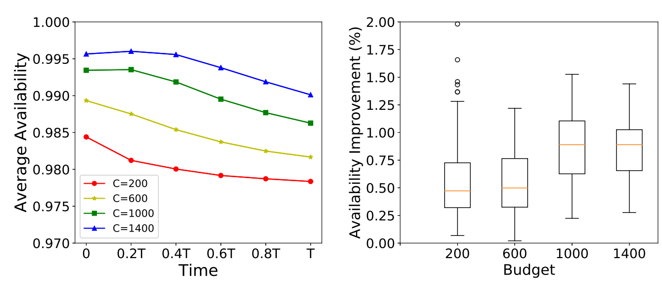


(a) (b)

Fig. 2. Availability improvement by utilizing renewable energy. (a) Availability improvement of GVB compared to SVB with an increasing amount of renewable energy. (b) Availability with and without the consideration of renewable energy in GVB (shown by GVB and N-GVB) and SVB (shown by G-SVB and SVB).

As mentioned in Section I, the reason why GVB can further improve availability is that it utilizes renewable energy generated at each time slot and handles arbitrary VNF failures in an online manner. We use Fig. 2 (a) to illustrate detailed availability improvement when the proportion of renewable energy increases compared to the backup budget. The simulation setting is the same as that of Fig. 1 except the weight of renewable energy. It is obvious that the more renewable energy available, the better GVB will perform compared to schemes which do not consider renewable energy.

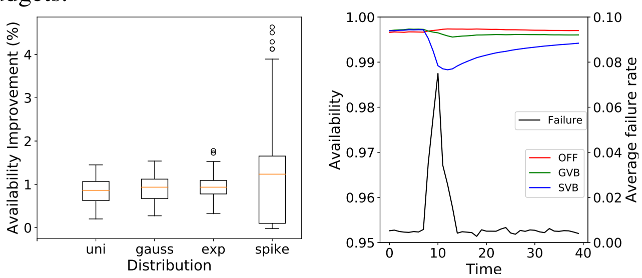
We further use Fig. 2 (b) to show availability of GVB and SVB with and without using renewable energy. availability of GVB without utilizing renewable energy is denoted by N-GVB in a dashed line. Similarly, the dashed line marked G-SVB represents availability of SVB which can perfectly predict the generation of renewable energy and use it for backups. It is clear that both algorithms achieve higher availability over the entire time span when utilizing renewable energy. In addition, even utilizing renewable energy by predicting its generation accurately, performance of SVB is inferior to that of GVB. The reason is that GVB considers online patterns of both renewable energy and failure rates and dynamically adjusts the number of backups for better performance. Thus, the utilization of renewable energy and the online manner of handling VNF failures in GVB both help to improve availability of VNFs when the backup budget is limited.



(a) (b)

Fig. 3. Availability improvement of GVB compared to the baseline with different backup budgets. (a) VNF availability of GVB with different backup budgets when time increases from 0 to T. (b) The box plot of availability difference between GVB and SVB with an increasing backup budget.

Since the amount of backup budget is the determining factor of availability, We then analyze how GVB performs with different adequacy of the backup budget. Fig. 3 (a) shows the change of VNF availability when time increases to T with different backup budgets. As expected, GVB performs better with a larger backup budget. Fig. 3 (b) further shows availability difference between GVB and SVB with an increasing budget. We observe that availability improvement does not increase monotonically with the growth of C. This is because availability of VNFs is relatively higher for all backup schemes with a larger backup budget. Then, there is less room for GVB to further optimize availability. According to Fig. 3, we can thus conclude that GVB always provides much higher availability than the baseline with rather limited backup budgets.



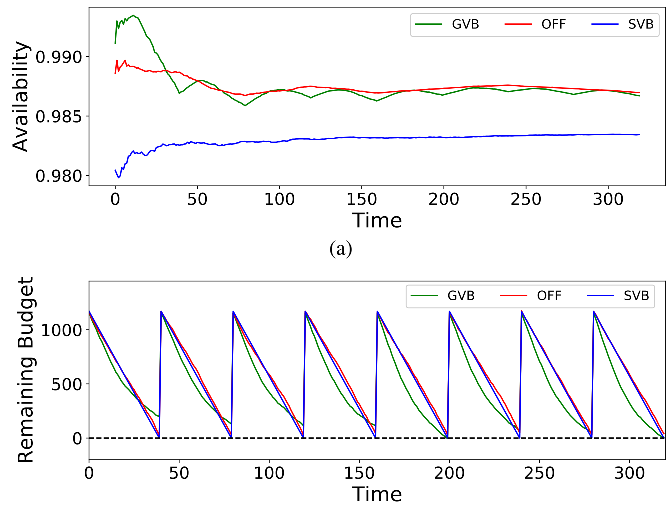
(a) (b)

Fig. 4. Performance of GVB under different VNF failure distributions. (a) The box plot of availability improvement between GVB and the baseline SVB under failure rates following uniform, Gauss, exponential and non-stationary distributions. (b) An example of performance of different algorithms when a failure spike happens.

We further evaluates performance of GVB with different failure distributions in Fig. 4. Since VNFs failures could be arbitrary, we apply GVB and the baseline under failure rate distributions of uniform, exponential, Gauss and non-stationary with random spikes in Fig. 4 (a). The mean of failure rates in all distributions is set to 2.5%. It is obvious that GVB always maintains higher VNF availability with different failure rate distributions. In addition, GVB performs much better than the static baseline when failure spikes happen. A failure spike is a period of time when the failure rates are much higher than those under normal circumstances. In such cases, multiple VNFs could failure at the same time and these failures may be relevant [26]. GVB performs better under failure spikes because it can flexibly deploy more backups during the failure spikes while leaving fewer backups at the rest of the time with lower failure rates. Fig. 4 (b) shows an example of different algorithms operating under a severe failure spike. During the time interval when failure rates of VNFs increase sharply, availability of the static baseline SVB decreases obviously. On the contrary, GVB reserves high availability comparable to the offline solution. Based on such observations and analysis, we conclude that GVB can achieve higher availability compared to the static baseline when non-stationary VNF failures happen.

D. GVB in the Long Run

In this section, we evaluate the performance of the GVB scheme when applied in longer time spans consisting of multiple budget cycles. Fig. 5 illustrates how the parameter adjustment method works for better performance with the increase of time.



(a)

(b)

Fig. 5. Performance of the GVB scheme in the long run compared with the offline solution and SVB. (a) Average availability of VNFs of different schemes in a long time span of 320 time slots, i.e., $8T$. (b) The backup budget expenditure of different schemes in the long time span.

Fig. 5 (a) shows the trend of average availability of all VNFs in the process of time when applying GVB, the offline solution, and the static baseline SVB. Fig. 5 (b) shows the corresponding backup budget expenditure over time. From figure (a), we observe that average availability of GVB fluctuates over time and departs from the offline solution at the end of each backup budget cycle. However, as time goes by, the amplitude of the fluctuation decreases and availability of GVB converges to the offline solution. In this process, the parameter adjustment method in GVB keeps finding better p for each budget cycle which achieves high availability without violating the budget limitation. The non-stationary states, i.e., failure rates and renewable energy, in each T vary and make the best value of p different for each cycle. Yet, with the adjustment of p, GVB always trends to the best performance. See the time interval $[200, 320]$ as an example. In this way, we can conclude that, with efficient parameter adjustment methods to find proper p for each budget cycle, GVB achieves high VNF availability close to the offline solution and much better than the static baseline in the long run.

VI. Conclusion and Future Work

The virtualization of network functions brings more flexibility and scalability at the expense of low reliability. In this paper, we propose a novel dynamic backup strategy GVB to improve availability of VNFs under a limited budget and renewable energy. GVB leverages renewable energy for higher availability while handling the non-stationary VNF failures. At the core of GVB is an online algorithm designed to solve the formulated VNF reliability optimization problem. The online algorithm is proven to achieve a competitive ratio compared to the offline optimum. Simulation results highlight that GVB significantly improves availability of VNFs compared to the baselines while respecting the budget.

As illustrated in Section IV, we can pick p smaller than the theoretical value p_0 for better performance in the long run. Due to space limitation, we only provide the most straight forward method to find proper p for each budget cycle. The value of p could be adjusted dynamically for more tighten theoretical bound and better performance based on various online feedback, e.g., the remaining rounding budget r_t at each time slot t. We will address this issue in our future work.

ACKNOWLEDGEMENTS

This research work was supported in part by the U.S. National Science Foundation under grant numbers CCF-1526162, CCF-1717731, CNS-1617698, CNS-1730128, and CNS-1717588.

REFERENCES

[1] Q. Zhang, F. Liu, and C. Zeng, "Adaptive interference-aware vnf placement for service-customized 5g network slices," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2449–2457.

[2] Z. Luo, C. Wu, Z. Li, and W. Zhou, "Scaling geo-distributed network function chains: A prediction and learning framework," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1838–1850, 2019.

[3] N. Kiji, T. Sato, R. Shinkuma, and E. Oki, "Virtual network function placement and routing model for multicast service chaining based on merging multiple service paths," in *2019 IEEE 20th International Conference on High Performance Switching and Routing (HPSR)*. IEEE, 2019, pp. 1–6.

[4] W. Bao, D. Yuan, B. B. Zhou, and A. Zomaya, "Prune and plant: Efficient placement and parallelism of virtual network functions," *IEEE Transactions on Computers*, 2020.

[5] J. Zheng, C. Tian, H. Dai, Q. Ma, W. Zhang, G. Chen, and G. Zhang, "Optimizing nfv chain deployment in software-defined cellular core," *IEEE Journal on Selected Areas in Communications*, 2019.

[6] X. Shang, Z. Liu, and Y. Yang, "Network congestion-aware online service function chain placement and load balancing," in *Proceedings of the 48th International Conference on Parallel Processing*, 2019, pp. 1–10.

[7] J. Martins, M. Ahmed, C. Raiciu, V. Olteanu, M. Honda, R. Bifulco, and F. Huici, "Clickos and the art of network function virtualization," in *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*, 2014, pp. 459–473.

[8] Z. Ni, G. Liu, D. Afanasev, T. Wood, and J. Hwang, "Advancing network function virtualization platforms with programmable nics," in *2019 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*. IEEE, 2019, pp. 1–6.

[9] J. Fan, M. Jiang, O. Rottenstreich, Y. Zhao, T. Guan, R. Ramesh, S. Das, and C. Qiao, "A framework for provisioning availability of nfv in data center networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 10, pp. 2246–2259, 2018.

[10] X. Shang, Z. Li, and Y. Yang, "Rerouting strategies for highly available virtual network functions," *IEEE Transactions on Cloud Computing*, 2019.

[11] Y. Kanizo, O. Rottenstreich, I. Segall, and J. Yallouz, "Optimizing virtual backup allocation for middleboxes," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2759–2772, 2017.

[12] X. Shang, Y. Huang, Z. Liu, and Y. Yang, "Reducing the service function chain backup cost over the edge and cloud by a self-adapting scheme," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020.

[13] J. Zhang, Z. Wang, C. Peng, L. Zhang, T. Huang, and Y. Liu, "Raba: Resource-aware backup allocation for a chain of virtual network functions," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 1918–1926.

[14] T. Taleb, A. Ksentini, and B. Sericola, "On service resilience in cloud-native 5g mobile systems," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 483–496, 2016.

[15] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE internet of things journal*, vol. 3, no. 5, pp. 637–646, 2016.

[16] N.-T. Dinh and Y. Kim, "An efficient availability guaranteed deployment scheme for iot service chains over fog-core cloud networks," *Sensors*, vol. 18, no. 11, p. 3970, 2018.

[17] C. Wang, J. Li, F. Ye, and Y. Yang, "Netwrap: An ndn based real-timewireless recharging framework for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1283–1297, 2014.

[18] Y. Gao, W. Dong, L. Deng, C. Chen, and J. Bu, "Cope: Improving energy efficiency with coded preambles in low-power sensor networks," *IEEE transactions on industrial informatics*, vol. 11, no. 6, pp. 1621–1630, 2015.

[19] M. Harris, "Tech giants race to build orbital internet [news]," *IEEE Spectrum*, vol. 55, no. 6, pp. 10–11, 2018.

[20] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew, "Greening geographical load balancing," *ACM SIGMETRICS Performance Evaluation Review*, vol. 39, no. 1, pp. 193–204, 2011.

[21] C. Gao, W. Zhang, J. Tang, C. Wang, S. Zou, and S. Su, "Relax, but do not sleep: A new perspective on green wireless networking," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, 2014, pp. 907–915.

[22] F. Kong and X. Liu, "Greenplanning: Optimal energy source selection and capacity planning for green datacenters," in *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2016, pp. 1–10.

[23] T. N. Le, Z. Liu, Y. Chen, and C. Bash, "Joint capacity planning and operational management for sustainable data centers and demand response," in *Proceedings of the Seventh International Conference on Future Energy Systems*, 2016, pp. 1–12.

[24] M. Marwah, A. A. McReynolds, A. J. Shah, Z. Wang, C. Patel, D. J. Gmach, C. D. Hyser, N. Kumari, Z. Liu, C. E. Bash *et al.*, "Managing a facility," Sep. 18 2014, uS Patent App. 14/353,607.

[25] H. Yi, M. H. Hajiesmaili, Y. Zhang, M. Chen, and X. Lin, "Impact of the uncertainty of distributed renewable generation on deregulated electricity supply chain," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6183–6193, 2017.

[26] G. Wang, L. Zhang, and W. Xu, "What can we learn from four years of data center hardware failures?" in *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2017, pp. 25–36.

[27] Y. Jia, C. Wu, Z. Li, F. Le, and A. Liu, "Online scaling of nfv service chains across geo-distributed datacenters," *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 699–710, 2018.

[28] X. Chen, W. Ni, I. B. Collings, X. Wang, and S. Xu, "Automated function placement and online optimization of network functions virtualization," *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1225–1237, 2018.

[29] T. Liu, Y. Zhu, H. Zhu, J. Yu, Y. Yang, and F. Ye, "Online pricing for efficient renewable energy sharing in a sustainable microgrid," *The Computer Journal*, vol. 60, no. 2, pp. 268–284, 2017.

[30] Y. Liu, Z. Liu, and Y. Yang, "Non-stationary stochastic network optimization with imperfect estimations," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 431–441.

[31] Q. Lin, H. Yi, J. Pang, M. Chen, A. Wierman, M. Honig, and Y. Xiao, "Competitive online optimization under inventory constraints," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 3, no. 1, pp. 1–28, 2019.

[32] J. Comden, S. Yao, N. Chen, H. Xing, and Z. Liu, "Online optimization in cloud resource provisioning: Predictions, regrets, and algorithms," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 3, no. 1, pp. 1–30, 2019.

[33] Y. Zeng, Y. Huang, Z. Liu, and Y. Yang, "Joint online edge caching and load balancing for mobile data offloading in 5g networks," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 923–933.

[34] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[35] "Solar power data for integration studies," https://www.nrel.gov/grid/solar-power-data.html, accessed January 10, 2020.