# A Mathematical Framework for Measuring Network Flexibility

Péter Babarczi[a,b], Markus Klügel[a], Alberto Martínez Alba[a], Mu He[a], Johannes Zerwas[a], Patrick Kalmbach[a], Andreas Blenk[a,c], Wolfgang Kellerer[a]

[a]Chair of Communication Networks, Technical University of Munich, Germany
[b]MTA-BME Future Internet Research Group, Dept. of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Hungary
[c]Communication Technologies, Faculty of Computer Science, University of Vienna, Austria

**Abstract**

In the field of networking research, increased flexibility of new system architecture proposals, protocols, or algorithms is often stated to be a competitive advantage over its existing counterparts. However, this advantage is usually claimed only on an argumentative level and neither formally supported nor thoroughly investigated due to the lack of a unified flexibility framework. As we will show in this paper, the flexibility achieved by a system implementation can be measured, which consequently can be used to make different networking solutions quantitatively comparable with each other. The idea behind our mathematical model is to relate network flexibility to the achievable subset of the set of all possible demand changes, and to use measure theory to quantify it. As increased flexibility might come with additional system complexity and cost, our framework provides a cost model which measures how expensive it is to operate a flexible system. The introduced flexibility framework contains different normalization strategies to provide intuitive meaning to the network flexibility value as well, and also provides guidelines for generating demand changes with (non-)uniform demand utilities. Finally, our network flexibility framework is applied on two different use-cases, and the benefits of a quantitative flexibility analysis compared to pure intuitive arguments are demonstrated.

*Key words:* network flexibility, cost of flexibility, measure theory, demand changes, softwarized networks

## 1. Introduction

In the last decades, communication networks have interweaved with all areas of our society, influencing segments as different as social media, industrial production and health-care. New application requirements create a need for dynamic changes of the working resources, e.g., to react to social events like flash crowds or to shifts of communication demands like in day-night patterns. It is commonly accepted that the ossification of existing networks – in particular of the Internet – constitutes a lack of flexibility to adapt themselves to changing requirements efficiently on a sufficiently small time-scale.

In recent years, several concepts have emerged to increase flexibility in networks through virtualization, reconfigurable data center topologies and control-plane programmability [1]. The split between data plane and control plane, proposed by software defined networking (SDN) [2], is regarded as the basic concept to provide flexible network adaptation. In addition, network virtualization (NV) [3] allows the sharing of physical network resources by different, independent networks. Furthermore, the softwarization of network functions via network function virtualization (NFV) [4], replacing previously used middleboxes, enables dynamic adaptations that add to flexibility. Altogether, these paradigms act as an enabler towards more flexible network operation.

General in most research, increased flexibility is often claimed only on an argumentative level and not analyzed quantitatively, mainly because the lack of a unified flexibility framework and a common understanding of network flexibility. However, a quantitative analysis might reveal weaknesses, unintuitive findings and trade-offs required to achieve the promised flexibility. For example, in [5] the authors demonstrate that while reconfigurable data center topologies are obviously *more flexible in adapting the topology* to changing traffic matrices, their performance in terms of serving traffic matrices with low traffic demands is not automatically better, as these demands do not require adaptation and can be served cheaper and faster by a static configuration. Hence, they conclude that static data center topologies can be *more flexible in terms of supporting throughput* than re-configurable ones depending on the traffic demands. We argue that a mathematical framework of network flexibility is required to help revealing these (and similar) claims, and make flexibility measurable.

Network flexibility is often investigated purely from an adaptability perspective and neglects possible trade-offs that might be induced by realizing the potential flexibility, e.g., over-provisioning or buying extra hardware is costly and might increase the complexity of network operation. In order to formally analyze the impact of flexibility on communication networks and enable meaningful trade-off analysis, flexibility needs to be quanti-

*Email addresses:* peter.babarczi@tum.de (Péter Babarczi), markus.kluegel@tum.de (Markus Klügel), alberto.martinez-alba@tum.de (Alberto Martínez Alba), mu.he@tum.de (Mu He), johannes.zerwas@tum.de (Johannes Zerwas), patrick.kalmbach@tum.de (Patrick Kalmbach), andreas.blenk@tum.de (Andreas Blenk), wolfgang.kellerer@tum.de (Wolfgang Kellerer)

fied in a formally clean fashion. Clearly, when such a quantification is possible, this enables follow-up arguments: (i) it enables the relative comparison of systems and their ordering in the sense of being more or less flexible; (ii) it enables the establishment of flexibility scaling-laws, similar to algorithmic complexity discussions; (iii) it enables the explicit design of systems for increased flexibility; (iv) and most importantly, it enables a tractable trade-off analysis with respect to introduced cost, time, traffic demands and system complexity.

As already mentioned, increased flexibility is often claimed in the context of SDN, NV and NFV. However, a formal argument to support these findings is mostly missing in the respective literature [1, 5, 6] – in these times when flexibility is one of the key promises given by network researchers. Although different fields have their respective flexibility metrics [1], they usually consider flexibility purely as the number of available options to change the system. However, the effort and time to perform these changes – which we think has utmost importance in networking – is missing from these definitions [1]. Hence, in this paper we aim at starting the formal analysis of network flexibility by proposing the use of mathematical measure functions on the size of appropriately defined set of demands to capture it [7]. Obviously, there is no such thing as "the" definition of flexibility, such that there also is no single way to assess it. However, we aim at providing a clean flexibility notion that (i) is applicable to communication networks, (ii) is consistent with the intuitive usage of the term, that (iii) does not lead to mathematical inconsistencies and (iv) can be assessed analytically as well as with an empirical measurement procedure. Compared to our initial measure definition fulfilling these requirements [7], in the current manuscript we further refine these notions and improve our definitions. Furthermore, in order to provide a whole network flexibility framework we discuss different approaches for demand set generation and introduce a basic model for measuring the cost of flexibility.

The rest of the paper is organized as follows. In Section 2 we enumerate basic research results on flexibility in general and also in the context of networking. The main contributions of the paper are introduced in Section 3 and Section 4, which contain the theoretical and empirical results on our network flexibility measure. We discuss how to generate demands in a meaningful way for the analysis in Section 5, and analyze the cost of the system providing a given level of flexibility in Section 6. The numerical evaluations for two concrete use cases are presented in Section 7, and finally the paper is concluded in Section 8.

## 2. Background and Related Work

The notion of flexibility has been defined in several contexts and in several different ways. In order to clarify what we mean by measuring flexibility, in this section we highlight the most important works related to our novel network flexibility framework. For a full survey of flexibility research the interested reader can refer to [1].

### *2.1. Flexibility in Communication Networks*

Flexibility has emerged recently as a core target for communication network designs and is explicitly or implicitly touched by many works on SDN, NV and NFV, but even the understanding of the word "flexible" strongly differs among different papers. The works that directly discuss or target flexibility itself, and that we are aware of, can be reduced to [5, 6, 8, 9]. In [5, 6], the authors discuss the flexibility of traffic engineering solutions in data centers, arguing against the assumption that wired connections are inflexible. To support their argument, the used flexibility metric is the throughput performance for increasing traffic, compared to a "throughput proportional" behavior. Works [8, 9] propose to measure flexibility by using an acceptance ratio over a set of induced change requests. While this measure is compatible with the notion we develop here, the authors do not introduce any formal argument of why it reflects the flexibility of a system.

### *2.2. Flexibility in Different Fields*

While not commonly discussed in networking, flexibility analysis is a tool that has been used in other scientific contexts [10–24], such as manufacturing systems, management science (where an increased interest exists already for more than four decades), software engineering, and power systems.

A flexible manufacturing system (FMS) [10–15, 21] can be reconfigured to match changing requirements, e.g., changed (or new) production volume, production flow or produced product. As shown in the survey [11], many works follow the naive approach of defining a flexibility function and demanding different intuitive properties to be satisfied, e.g., it should increase with production volume but decrease with required production time. As argued in [11], each function falls into one out of five main streams, which are generalized in [13], however without resulting in a single, consistent metric. This is criticized in [15], as some of the introduced metrics are even shown to produce inconsistent relative orderings of more or less flexible manufacturing systems. Exceptions are the works [10, 14], which define flexibility more formally as the weighted efficiency of machines over a possible task set [10] and the distance that can be traversed by a flexible manufacturing systems (FMS) in a given state space [14].

In contrast to FMS, decision theory considers the impact of decision flexibility [16–20], which is consistently defined and treated throughout the literature. Here, the focus is on companies that need to make decisions, influencing an unknown or only partly known future. Decision flexibility is then defined either as the amount or as the revenue of future options enabled by a current decision. For instance, in [22] the authors address the definition and measurement of flexibility of financial decisions.

In the field of software engineering, flexibility has been studied by some authors as an indicator of software quality in a similar fashion to software complexity. In [23], software flexiblity is formally defined as the *evolution complexity* of the software, which measures the effort – e.g., in lines of code that have to be modified – of adjusting a software implementation to respond to a change in the addressed problem.

In addition, flexibility is also a well studied concept in the field of power systems. This is motivated by the increased variability in the energy generation and demand that appeared in recent years, originated by the evergrowing use of renewable energy sources, changes in the market, and appearence of new technologies such as the smart grid. In [24], the authors quantify the flexibility of a power system by combining three indicators that measures how much, for how long, and how fast a power system can change to adapt to a changing demand. Combining the views of FMS, decision theory and power systems, *flexibility is in general related to an option set*, i.e., a set of possible tasks, achievable system states or available decisions.

### 2.3. Flexibility versus Other Metrics

As we have seen, flexibility refers to be able to react to possible future challenges. Preferring system (or network) states with a large option set to possible future challenges is often referred to as intelligence, and can be modelled both as a force that drives the system toward states with higher entropy [25], or as an information-theoretic metric that can be used to calculate the influence an agent has on its environment in a given state [26] (called empowerment). Using the achievable option set size as a performance metric is identifiable in networking context as well. For example, in minimum interference routing [27] we use the least congested links instead of shortest paths in order to maximize the future communication requests that can be served, while a robust communication network is prepared to function correctly in the presence of unknown perturbations in the future [28, 29].

However, as we have seen [1], in addition to a large option set, network flexibility should involve a dynamic component as well: *are we able to realize this potential in a timely and cost-efficient manner?* For example, although minimum interference routing keeps future options open, calculating an optimal detour path around the network might take excessive time and the delay (can be considered as cost) of this path might be unacceptable for the applications. Furthermore, a robust network might be prepared for failures, but the cost of the redundant resources and the adaptation cost to the new condition might be too expensive. We believe that time and cost constraints should be included in a useful network flexibility metric; however, to the best of our knowledge, no such metric exists.

## 3. Network Flexibility

Just as most high-level terms, such as "creativity", "intelligence" or "fairness", the meaning of the term flexibility is not easily defined in a clear manner, nor can any short definition capture the full meaning of the term. However, from our literature review, we realize that the main features of interest for communication networks are threefold, referring to (i) the variety of adaptation possibilities, (ii) the speed of adaptation and (iii) the overhead/cost of adaptation. For example, pro-active dedicated protection approaches [30] can recover from single link failures instantaneously for the price of using backup resources (see Section 7.2). On the other hand, re-active restoration can respond to multiple link failures as well without extra

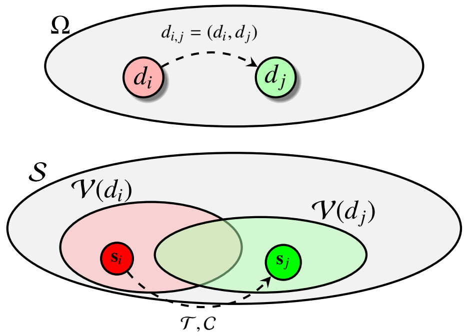



Figure 1: Our system model with demand set $\Omega$ in the upper part containing demands $d_i, d_j$ and possible demand change $d_{i,j} = (d_i, d_j)$, and with the system states $\mathcal{S}$ in the lower part depicting possible valid states $\mathcal{V}(d_i)$ and $\mathcal{V}(d_j)$ for the corresponding demands. Demand change $d_{i,j}$ induces system state change $\mathbf{s}_i \mapsto \mathbf{s}_j$, which requires action time $\mathcal{T}$ and action cost $C$.

resources, but takes excessive time to respond to the network failure [7]. Based on these observations, without claiming completeness of this definition, we refer to flexibility in the following fashion:

**Definition 1.** *Given the demands the communication network has to respond, network flexibility is the ability of the network to adapt its state to satisfy the new demands* promptly *and with* little effort.

Here, a state can be, e.g., the routes used by communication flows or network resource usage patterns that satisfy certain demands. The effort can be related to any cost metric, such as overhead (e.g., control plane message or bandwidth cost), system complexity and monetary cost (i.e., relating to capital expenditures and operational expenses), and the conditions can be externally given by technological constraints or actively demanded by the users through service level agreements. In an abstract sense, different states can be realized by a given system implementation, which is bound to specific protocols, hardware and software modules. A change of state can then happen on different time-scales and with differing effort and lead or not lead to a proper adaptation to changing conditions. For example, in restoration new paths for the disrupted connections are calculated after a failure occurs. However, if the algorithm cannot find new paths owing to the lack of resources, the network cannot be adapted to the new state.

It is the above interaction of changing conditions, state change, speed of change and needed effort to change that impact our view on flexibility. Hence, in our framework besides introducing a measure for network flexibility based on adaptation time and cost, we will also investigate the effect of changing conditions (i.e., demands in Section 5) and analyse the cost of operating the network and changing states (in Section 6).

### 3.1. System Model

The notations of our model are summarized in Table 1.

**System State.** We consider a communication system that can be described by a *system state* $\mathbf{s} \in \mathcal{S}$, where $\mathcal{S}$ contains all possible states that the system can realize. The state can

Table 1: Notation list for the network flexibility framework

| Notations | Description |
|---|---|
| $\mathbf{s} \in \mathcal{S}$ | set of all states $\mathbf{s}$ of the communication system |
| $d_i \in \Omega$ | set of demands $d_i$ (e.g., connection requests) that can be posed to the network |
| $\mathcal{V}(d_i)$ | set of valid states $\mathcal{V}(d_i) \subseteq \mathcal{S}$ in which demand $d_i$ is satisfied |
| $X \in \mathcal{X}$ | set of implementations (e.g., algorithms) $X$ which realizes states from $\mathcal{S}_X \subseteq \mathcal{S}$ |
| $\mathcal{V}_X(d_i)$ | valid system state(s) $\mathcal{S}_X \cap \mathcal{V}(d_i)$ which can be achieved by $X$ for demand $d_i$ |
| $d_{i,j} \in \Omega^2$ | set of demand changes $d_{i,j}$ with initial demand $d_i$ and new demand $d_j$ |
| $\mathcal{T}(d_{i,j}), C(d_{i,j})$ | action time and cost (vector) required to perform demand change $d_{i,j}$ |
| $\mathcal{A}_X(T, \mathbf{C})$ | set of achievable demand changes by $X$ within constraints $\mathcal{T}(d_{i,j}) \leq T; C(d_{i,j}) \leq \mathbf{C}$ |
| $\mu(\mathcal{A}_X(T, \mathbf{C}))$ | network flexibility of $X$ wrt. $T$ and $\mathbf{C}$, where $\mu$ is an appropriate measure on $\Omega^2$ |
| $\mathcal{D}$ | demand change sequence the network is challenged with $\{d_{i_1,j_1}, d_{i_2,j_2}, d_{i_3,j_3}, \ldots\}$ |
| $\hat{\mu}_{\mathcal{D}}(\mathcal{A})$ | estimated network flexibility with respect to demand change sequence $\mathcal{D}$ |

reflect, e.g., the routes used by communication flows, end-to-end connectivity patterns, network resource usage patterns or the positions of network functions or controllers. We assume it to be given in an appropriate descriptive form, which can be a set, tuple, matrix or vector but is in principle not limited to such, e.g., in Fig. 2 the system states are the shortest paths realizing the corresponding demand.

**Demand Set.** Furthermore, we define a *demand set* $\Omega$ that captures demands posed to the network. Demands are requirements on the network state and can be used to model, e.g., connection requests, rate or Quality of Service (QoS) demands. Each demand $d_i \in \Omega$ is associated with a set of valid states $\mathcal{V}(d_i) \subseteq \mathcal{S}$, in which the demand is satisfied. For instance, an end-to-end connection request can satisfied by a state where all links/switches are set-up properly in a network. Over time, the demands will vary (e.g., flows will leave a network, users will change their locations whatever) and the system will adapt its state in order to satisfy the demand changes. A *demand change* is an event denoted by the tuple of initial demand $d_i$ and new demand $d_j$, i.e., $d_{i,j} = (d_i, d_j)$ for $d_i, d_j \in \Omega$. We will use the notations $\Omega \times \Omega$ and $\Omega^2$ to denote the base set of demand changes interchangeably. Each change $d_{i,j} \in \Omega^2$ demands that the system is adapted from $\mathbf{s}_i \mapsto \mathbf{s}_j$, where $\mathbf{s}_i \in \mathcal{V}(d_i)$, $\mathbf{s}_j \in \mathcal{V}(d_j)$, respectively. In Fig. 1 we depicted a possible relationship of the corresponding sets and adaptation of the system states in response to demand change $d_{i,j}$. One can observe that the sets $\mathcal{V}(d_i)$ and $\mathcal{V}(d_j)$ might contain common states (i.e., where both $d_i$ and $d_j$ are satisfied); thus, a demand change does not necessarily induce a state change if the actual realization is part of both sets.

**System Implementation.** Assume a *system implementation* $X \in \mathcal{X}$, where $\mathcal{X}$ is the set of possible implementations, which is bound to specific algorithms, protocols, hardware and software modules. Due to its nature, $X$ can realize any system state out of a set $\mathcal{S}_X \subseteq \mathcal{S}$. Consequently, $\forall d_i \in \Omega$ there is a set of valid states, $\mathcal{V}_X(d_i) = \mathcal{S}_X \cap \mathcal{V}(d_i)$, that can be achieved by $X$. An example set-up is shown in Fig. 2, which consists of a network where in each state $\mathbf{s}_i$ a single node-pair can communicate with each other on a simple path. The demands $d_i$ are the connection requests between node-pairs, and the valid system states $\mathcal{V}(d_i)$ for demand $d_i$ are the possible realizations (i.e., simple paths or routings) of the connection.

In Fig. 2, the implementation $X$ realizes each connection according to a shortest path routing algorithm and hence will deterministically select a realization with the minimum number of hops from $\mathcal{V}(d_i)$ for every demand. Therefore, $\mathcal{V}_X(d_i)$ contains the single system state corresponding to the chosen shortest path. For example, valid system states for $d_1$ are $\mathcal{V}(d_1) = \{v_1 - v_2, v_1 - v_3 - v_2, v_1 - v_3 - v_4 - v_2\}$, while $\mathcal{V}_X(d_1) = \{\mathbf{s}_1\} = \{v_1 - v_2\}$. A demand change $d_{1,4} = (d_1, d_4)$ in Fig. 2(a) will induce a system state change from $\mathbf{s}_1 \mapsto \mathbf{s}_4$, the realization of which requires a certain amount of time and comes at a certain cost.

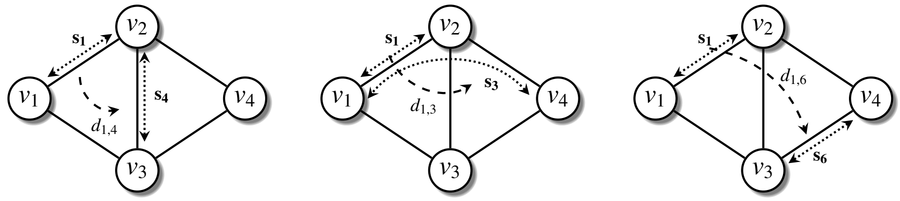
**Action Time & Cost.** In general, the time needed for the system to adapt (including all the required processing such as algorithm execution time and time due to implementation artifacts) is described by the *action time* $\mathcal{T} : \Omega^2 \mapsto \mathbb{R}^+$, which maps each demand change to its appropriate time value (e.g., time elapsed until the new flow is established in Fig. 2). Furthermore, each demand change is associated with an *action cost*, which is described by the mapping $C : \Omega^2 \mapsto \mathbb{R}^+$ and reflects the effort of adapting to the demand change (e.g., the number of forwarding table changes). As the action cost might have multiple components with different restrictions[1], we will denote it as a vector $\mathbf{C}$.

For the sake of simplicity and readability, throughout this paper we restrict ourselves to the simple model introduced above. In particular, we consider the case that $\mathcal{V}_X(d_i) \in \mathcal{S}_X$ is either empty or a single element, i.e., the demands are formulated in a way that only a single state will satisfy it. This applies, e.g., to systems where the demand itself is a system state or to systems that adapt themselves towards a unique optimum state, as in the given example in Fig. 2. In this model $\mathcal{T}$ and $C$ become deterministic, memoryless mappings on $\Omega \times \Omega \mapsto \mathbb{R}^+$, respectively: whether a system can adapt to changing demands only depends on the current system state and never on the past system states. However, we note that the introduced framework can be extended towards probabilistic state changes and valid sets $\mathcal{V}_X(d_i)$ containing multiple states [7].

### 3.2. Formal Network Flexibility Definition

A quantifier for flexibility should exhibit at least the following three properties to match the way we use the terminology. As we want to be able to sort systems and implementations with respect to flexibility, its quantifier should *allow an*

[1] Please, refer to Section 6.1 for a detailed cost of flexibility analysis.




(a) Demand change $d_{1,4}$ is realizable with $\tau_X = 1$ and $\mathbf{c}_X = 3$ (16 such $d_{i,j}$ demand changes).

(b) Demand change $d_{1,3}$ is realizable with $\tau_X = 2$ and $\mathbf{c}_X = 3$ (4 such $d_{i,j}$ demand changes).

(c) Demand change $d_{1,6}$ is realizable with $\tau_X = 1$ and $\mathbf{c}_X = 4$ (10 such $d_{i,j}$ demand changes).

| $(T, \mathbf{C})$ | $\mu(\mathcal{A})$ | $\hat{\mu}_{\mathcal{D}}(\mathcal{A})$ |
|---|---|---|
| (1,3) | 16 | 0.53 |
| (2,3) | 20 | 0.67 |
| (1,4) | 26 | 0.87 |
| (2,4) | 30 | 1.0 |

(d) Time and cost constraints, achievable demand changes by $X$, and corresponding flexibility values.

Figure 2: A 4-node example network with unit edge capacities and with demand set $\Omega = \{d_1 = (v_1, v_2), d_2 = (v_1, v_3), d_3 = (v_1, v_4), d_4 = (v_2, v_3), d_5 = (v_2, v_4), d_6 = (v_3, v_4)\}$ (i.e., 30 possible demand changes in $\mathcal{D}$). A shortest path routing algorithm $X$ realizes the path with the least number of hops from $\mathcal{V}(d_i)$, e.g., $\mathcal{V}_X(d_1) = \{\mathbf{s}_1\} = \{v_1 - v_2\}$ for $d_1$. The adaptation cost $\mathbf{c}_X$ is the number of reconfigured nodes (switches), and the action time $\tau_X$ corresponds to the number of steps the demand change can be realized (i.e., in 1 step if the edge(s) for the new demand are not occupied, and in 2 if the capacity of the edge was used).

*ordering* (i.e., a "≤" operation), in which more flexible systems should achieve a larger flexibility value. Furthermore, it should be possible to *have a "totally inflexible" system*, which intuitively should be assigned value zero. Because such an inflexible system is less flexible than any other system, *flexibility should have non-negative values*.

From our previous discussion, flexibility is related to the amount of demand changes that a system can support, to the time scale at which it can serve a demand and to the effort associated with it. Using the system model described in Section 3.1, we introduce the following key definition of our framework:

**Definition 2.** *The set of* **achievable demand changes** *by the considered system implementation X under given action time constraint T and action cost constraint* $\mathbf{C}$ *is defined as:*

$$\mathcal{A}_X(T, \mathbf{C}) = \Big\{d_{i,j} \in \Omega \times \Omega : i \neq j; \mathcal{V}_X(d_i), \mathcal{V}_X(d_j) \neq \emptyset; \mathcal{T}(d_{i,j}) \leq T; C(d_{i,j}) \leq \mathbf{C}\Big\}.$$

The first line only ensures that $d_{i,j}$ is a valid demand change, i.e., it can be performed by implementation $X$. Next, we assess the flexibility of a system implementation as the "size" of the set[2] $\mathcal{A}_X(T, \mathbf{C})$. Armed with this fact, we are ready to present our network flexibility definition:

**Definition 3.** *Given a network implementation X with the set of achievable demand changes* $\mathcal{A}_X(T, \mathbf{C})$ *with respect to time and cost constraints T and* $\mathbf{C}$. *The* **network flexibility** *of X is defined as* $\mu(\mathcal{A}_X(T, \mathbf{C}))$*, where* $\mu$ *is an appropriate measure on* $\Omega \times \Omega$.

For ease of notation, we will drop inputs $X$, $T$, $\mathbf{C}$ when they do not explicitly contribute to the understanding and write $\mu(\mathcal{A})$ instead of $\mu(\mathcal{A}_X(T, \mathbf{C}))$. An example is given in Fig. 2(d), where $\mu$ is the *counting measure*, i.e., $\mu(\mathcal{A})$ is simply the number of achievable demand changes within $T$ and $\mathbf{C}$.

[2] Although for set size usually operator $|\cdot|$ is used, we will use set function $\mu(\cdot)$ instead to emphasize the variety of measures which can be applied.

### *3.3. Properties of the Network Flexibility Measure*

The size of achievable demand changes indeed meets our intuitions about flexibility. This we show with the following observations, which follow directly from Definition 3 and the properties of mathematical measures:

**Observation 1.** *Implementation X that cannot react to any demand change has zero flexibility, i.e., is inflexible, as* $\mathcal{A}_X(T, \mathbf{C}) = \emptyset$ *and hence* $\mu(\mathcal{A}_X(T, \mathbf{C})) = 0$.

The strictness of Observation 1, demanding that the system cannot react to any demand change, in fact implies that only few systems can be claimed to be completely inflexible.

**Observation 2.** *Implementation X is more flexible than Y if it can react to more demand changes under time and cost constraints T,* $\mathbf{C}$*, indicated by* $\mu(\mathcal{A}_Y(T, \mathbf{C})) \leq \mu(\mathcal{A}_X(T, \mathbf{C}))$*. Trivially, this is the case when* $\mathcal{A}_Y(T, \mathbf{C}) \subseteq \mathcal{A}_X(T, \mathbf{C})$.

**Observation 3.** *If implementation X and Y can realize different demand changes, an implementation* $Z = X \cup Y$ *can be constructed, that selects among X and Y the one that can realize a given demand change within the constraints, with ties broken arbitrarily. It holds* $\forall T, \mathbf{C}$ *that*

$$\mu(\mathcal{A}_Z(T, \mathbf{C})) \geq \max\{\mu(\mathcal{A}_X(T, \mathbf{C})), \mu(\mathcal{A}_Y(T, \mathbf{C}))\}.$$

Proof. This follows from $\mathcal{A}_Z(T, \mathbf{C}) = \mathcal{A}_X(T, \mathbf{C}) \cup \mathcal{A}_Y(T, \mathbf{C})$ which induces $\mu(\mathcal{A}_Z(T, \mathbf{C})) = \mu(\mathcal{A}_X(T, \mathbf{C})) + \mu(\mathcal{A}_Y(T, \mathbf{C})) - \mu(\mathcal{A}_X(T, \mathbf{C}) \cap \mathcal{A}_Y(T, \mathbf{C}))$ [31] and hence

$$\mu(\mathcal{A}_Z(T, \mathbf{C})) \geq \max\{\mu(\mathcal{A}_X(T, \mathbf{C})), \mu(\mathcal{A}_Y(T, \mathbf{C}))\}.$$

The arguments given in Observations 2 and 3 in fact reflect the perspective that is found often in literature when flexibility is claimed in the context of SDN, NFV, or NV: By enabling reconfiguration of the network, its functions, flows or similar, the resulting system is one that can do anything it could do before, and more; thus, it must be more flexible. From a formal perspective, this argument is only partly applicable to real systems, as it assumes that the reconfiguration itself does not induce any substantially increased delay or cost. As soon as the reconfiguration induces a non-negligible delay or cost overhead, *there are* *time $T$ or cost $\mathbf{C}$ constraints under which the re-configuration itself is a drawback that can actually make the network less flexible*. When network operation requires reaction times or cost efficiency corresponding to these tight constraints, the impact of this decrease needs to be investigated, otherwise increased flexibility can be an empty promise.

### *3.4. Dimensions of Network Flexibility*

While the evaluation over a size of achievable demand changes has an intuitive relation to flexibility, the introduction of time and cost constraints is at first a little counter-intuitive. However, their use is in accordance with the multi-dimensionality of flexibility has been observed already in the discussed literature. Formally, we establish the "dimensions" adaptability, reactivity and cost-efficiency. Consider two system implementations $X$, $Y$ with associated sets $\mathcal{A}_X(T, \mathbf{C})$ and $\mathcal{A}_Y(T, \mathbf{C})$, respectively[3]. Then, we define the following:

**Definition 4.** *We say that implementation $X$ is at least as reactive as $Y$ if*

$$\forall T: \quad \mu(\mathcal{A}_Y(T, \infty)) \leq \mu(\mathcal{A}_X(T, \infty)), \tag{1}$$

*that $X$ is at least as cost-efficient as $Y$ if*

$$\forall C: \quad \mu(\mathcal{A}_Y(\infty, \mathbf{C}) \leq \mu(\mathcal{A}_X(\infty, \mathbf{C})), \tag{2}$$

*and that $X$ is at least as adaptive as $Y$ if:*

$$\mu(\mathcal{A}_Y(\infty, \infty)) \leq \mu(\mathcal{A}_X(\infty, \infty)). \tag{3}$$

The reactivity property states that, disregarding cost, implementation $X$ can react to demand changes at least as fast as implementation $Y$. Analogy holds for cost-efficiency with respect to cost. The notions of reactivity and cost-efficiency can be interpreted as Pareto-superiority in the time and cost dimensions. Finally, adaptability states that, independent of time and cost, implementation $X$ can react to at least as many demand changes as implementation $Y$.

One can argue that $X$ is more flexible than $Y$ if it is better in any of these properties. Hence, each constraint combination $(T, \mathbf{C})$ of the flexibility evaluation can be interpreted as an emphasis on the adaptability, reactivity and cost-efficiency property, with more stringent constraints enforcing better reactivity and cost-efficiency. However, we note that *relaxing one or both of these constraints might result that flexibility is equivalent to existing metrics* (e.g., acceptance ratio in Section 7, or network robustness), which can be calculated with alternative tools as well. Although flexibility in these cases might not have any additional meaning, in order to keep our definition as general as possible, we do not want to exclude these cases.

Hence, in the rest of the paper, we will mainly focus on the general flexibility definition, but keep in mind that the findings are applicable to all of these dimensions with an appropriate constraint selection.

[3] We denote with "$\infty$" if a constraint is infinitely relaxed on that parameter.

### *3.5. Normalization of the Flexibility Measure*

We have motivated that flexibility can be quantified by a measure $\mu$ on $\mathcal{A}_X(T, \mathbf{C})$, which reflects the size of the set of achievable demand changes. Note that, with this definition flexibility is by no means a unique metric. For example, the use of different measure types, such as the Dirac measure versus the counting measure, will result in different flexibility values and even might lead to different relative orderings of systems. However, if $\mu$ is a valid measure, then $\mu'(\mathcal{A}) = c\mu(\mathcal{A})$ for any constant $c > 0$ fulfills all properties of a measure, too [31]. In this case, although $\mu'$ and $\mu$ will have different absolute values, they will induce the same ordering of implementations with respect to network flexibility. Therefore, we can normalize the measure with respect to an arbitrary base set $\mathcal{B} \subseteq \Omega \times \Omega$ that satisfies $0 < \mu(\mathcal{B}) < \infty$ in order to provide intuitive meaning for our flexibility value by defining:

$$\mu_{\mathcal{B}}(\mathcal{A}) := \frac{1}{\mu(\mathcal{B})}\mu(\mathcal{A}). \tag{4}$$

It is easy to see that any scaling factor cancels out for $\mu_{\mathcal{B}}$:

$$\mu'_{\mathcal{B}}(\mathcal{A}) = \frac{1}{c\mu(\mathcal{B})}c\mu(\mathcal{A}) = \frac{1}{\mu(\mathcal{B})}\mu(\mathcal{A}) = \mu_{\mathcal{B}}(\mathcal{A}). \tag{5}$$

Different choices for $\mathcal{B}$ can in general lead to different intuitive interpretations.

- The direct "amount" of achievable demands in $\mathcal{A}_X(T, \mathbf{C})$ can be represented as the choice of a set with $\mu(\mathcal{B}) = 1$, and then has the following intuition:

  $$\mu(\mathcal{A}) = \mu_{\mathcal{B}}(\mathcal{A}) = \text{\# achievable demand changes.} \tag{6}$$

  Although we consider Eq. (6) as a measure for flexibility, the outcome will be a number between $[0, \mu(\Omega^2)]$, which is a rather arbitrary output, as we have seen in Fig. 2(d).

- Another option is the normalization by the maximum number of achievable demand changes over all possible implementations $X \in \mathcal{X}$, i.e, $\mathcal{B} := \mathcal{A}^*$, where

  $$\mathcal{A}^* = \arg\max_{X} \mu(\mathcal{A}_X(T, \mathbf{C})).$$

  This measure has the intuitive meaning of *flexibility degree* $\varphi_X(T, \mathbf{C}) := \mu_{\mathcal{A}^*}(\mathcal{A})$, which is

  $$\varphi_X(T, \mathbf{C}) = \frac{\mu(\mathcal{A})}{\mu(\mathcal{A}^*)} = \frac{\text{\# achievable changes by } X}{\text{\# max. achievable changes over } \mathcal{X}}.$$

  Obviously $\varphi_X(T, \mathbf{C}) \in [0, 1]$, which motivates the term degree. A flexibility degree of one corresponds to a system being "100%", i.e., maximum flexible. For example, if we can select any available path in the network in Fig. 2 instead of only the shortest path, then $\mu(\mathcal{A}^*_X(1, 4)) = 30$ (as we can satisfy demands in Fig. 2(b) in 1 step as well), hence $\varphi_X(T, \mathbf{C}) = 0.86$ for shortest path routing algorithm $X$ in our network. However, we note that it is not easy to find $\mu(\mathcal{A}^*)$ in general.

Although both the number of achievable demand changes and the flexibility degree are valid metrics, they do not have an intuitive meaning or they are hard to calculate in practice, respectively. In Section 4 we will discuss normalizations that can be achieved empirically, which conforms the intuitive meaning of flexibility.

## 4. Flexibility Evaluation

By the introduced theoretical framework in Section 3, measuring the flexibility of system implementation $X$ requires the identification of set $\mathcal{A}_X(T, \mathbf{C})$ – i.e., the set of possible demand changes under time constraint $T$ and cost constraint $\mathbf{C}$. In this section, we discuss how to empirically measure and compare the flexibility of different system implementations. Furthermore, we discuss how the results measured in different set-ups can be combined into an overall metric.

### *4.1. Exact Flexibility*

An intuitive way to measure $\mathcal{A}_X(T, \mathbf{C})$ for given $(T, \mathbf{C})$ constraints is introduced in the following. Given a system under test whose flexibility shall be calculated, we select an *infinite length demand change sequence*[4] $\mathcal{D} = \{d_{i_1,j_1}, d_{i_2,j_2}, d_{i_3,j_3}, \ldots\}$, which may contain arbitrary demand changes[5]. We argue that challenging the system with this sequence and observing its reaction for each demand change, the flexibility measure can be evaluated. First, for each demand change $d_{i,j}$, we assume that it occurs in $\mathcal{D}$ with a relative frequency

$$\nu(d_{i,j}) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{1}\{d_{i,j} = d_{i_k,j_k}\}. \qquad (7)$$

$\mathbb{1}\{\cdot\}$ therein is the indicator function, which is one if the logical statement is true and zero otherwise. We refer to $\nu$ as the *challenge profile*, which is imposed onto the system under test by sequence $\mathcal{D}$. By counting the number of changes that are in $\mathcal{A}$, we measure:

$$\mu^{\nu}(\mathcal{A}) = \int\limits_{d_{i,j}\in\Omega^2} \nu(d_{i,j})\mathbb{1}\{d_{i,j} \in \mathcal{A}\}d\mu = \int\limits_{d_{i,j}\in\mathcal{A}} \nu(d_{i,j})d\mu,$$

where the second equality holds because $\mathcal{A} \subseteq \Omega^2$ and the integral $\int \cdot\, d\mu$ denotes Lebesgue integration, which is an abstraction of integration towards arbitrary measurable set systems [31]. Intuitively, the system under test reacts to challenge profile $\nu$ and exhibits a certain flexibility. Now, because $\nu$ has all properties of a normalized density with cumulative measure of

$$\int_{\Omega^2} \nu(d_{i,j})d\mu = 1,$$

it can be interpreted as a probability density function, reflecting the probability of a demand change being challenged by the sequence $\mathcal{D}$. This leads to

$$\begin{aligned}\mu^{\nu}(\mathcal{A}) &= \int\limits_{d_{i,j}\in\Omega^2} \nu(d_{i,j})\mathbb{1}\{d_{i,j} \in \mathcal{A}\}d\mu \\ &= \sum_{d_{i,j}\in\Omega^2} \nu(d_{i,j})\mathbb{1}\{d_{i,j} \in \mathcal{A}\} = \sum_{d_{i,j}\in\mathcal{A}} \nu(d_{i,j}) \\ &= \mathbb{E}\left\{\mathbb{1}\{d_{i,j} \in \mathcal{A}\} \mid \mathcal{D}\right\} = \Pr\{d_{i,j} \in \mathcal{A} \mid \mathcal{D}\}. \qquad (8)\end{aligned}$$

$\mathbb{E}\{\cdot\}$ therein is the expectation value and $\Pr\{\cdot\}$ the probability of an event occurring. That is, flexibility corresponds to the expectation of the demand changes contained in $\mathcal{D}$ being in $\mathcal{A}$ under the specific challenge profile $\nu$. From the definition of the set $\mathcal{A}$, the expectation is nothing but the probability that the system can react to the challenged demand changes in sequence $\mathcal{D}$ within the target time and cost constraints $(T, \mathbf{C})$.

An important design choice for evaluating flexibility is the selection of sequence $\mathcal{D}$. In particular, if it can be chosen such that the challenge profile is uniform, i.e., $\forall d_{i,j} \in \mathcal{D} : \nu(d_{i,j}) = 1/\mu(\mathcal{D})$, then the flexibility is:

$$\begin{aligned}\mu^{\nu}(\mathcal{A}) &= \int\limits_{d_{i,j}\in\Omega^2} \nu(d_{i,j})\mathbb{1}\{d_{i,j} \in \mathcal{A}\}d\mu \qquad (9) \\ &= \frac{1}{\mu(\mathcal{D})} \int\limits_{d_{i,j}\in\mathcal{A}} 1d\mu = \frac{\mu(\mathcal{A})}{\mu(\mathcal{D})} = \mu_{\mathcal{D}}(\mathcal{A}), \qquad (10)\end{aligned}$$

which is proportional to $\mu(\mathcal{A})$. Hence, we get back to a normalized version of the flexibility measure defined in Sec. 3.5, with the selection of $\mathcal{B} = \mathcal{D}$. As we have argued, $\mu_{\mathcal{D}}(\mathcal{A})$ is equivalent to $\mu(\mathcal{A})$ in that sense that it induces the same relative ordering. On the other hand, if $\nu$ is not uniform, we will have a weighted flexibility value with more emphasis on the demand changes that occur more often. This better matches observed system behavior in a running environment, as the system might be challenged with the same demand changes (e.g., in flow demands, connection requests, virtual network requests, etc.) multiple times, while others might not be requested at all. Although such flexibility value will be distorted with respect to our original definition, we can claim it to be the flexibility of the system with respect to the given challenge profile $\nu$.

### *4.2. Estimated Flexibility*

In practice we cannot generate an infinite length demand change sequence to obtain the exact network flexibility value. Thus, we will have only a *finite length demand change sequence*, which distorts the resulting values because the targeted relative frequencies might not be matched precisely. However, the flexibility evaluation boils down to estimating the event probability from Eq. (8). Consider a demand change sequence $\mathcal{D}$ of finite length $N$, with elements randomly chosen out of $\Omega^2$ with uniform distribution. By using the empiric mean as estimator for the expectation, an estimate for $\mu_{\mathcal{D}}(\mathcal{A})$ can be created, which is

$$\hat{\mu}_{\mathcal{D}}(\mathcal{A}) = \frac{\hat{\mu}(\mathcal{A})}{\hat{\mu}(\mathcal{D})} = \frac{\sum_{k=1}^{N} \mathbb{1}\{d_{i_k,j_k} \in \mathcal{A}\}}{N}. \qquad (11)$$

[4] Where appropriate, we will consider sequence $\mathcal{D}$ as a set with unique elements in it.

[5] In the case of continuous demands, first we might need to identify a set of demand changes $\mathcal{D}' \subseteq \Omega^2$ with $\mu(\mathcal{D}') < \infty$ such that $\mathcal{A}_X(T, \mathbf{C}) \subseteq \mathcal{D}'$ holds, from which the demand changes are selected into $\mathcal{D}$.

**Algorithm 1** Estimating Flexibility $\hat{\mu}_{\mathcal{D}}(\mathcal{A})$

Input: $X, \Omega, T, \mathbf{C}, N$
Variables: $\Sigma_{\mathcal{A}} := 0$
Create sequence $\mathcal{D}$ with $N$ elements out of $\Omega^2$
**for** $k \in \{1, ..., N\}$ **do**
  Impose demand change $d_{i_k,j_k}$
  Observe $\delta_X := \begin{cases} 1, & \text{if } d_{i_k,j_k} \text{ is realized} \\ 0, & \text{else} \end{cases}$
  Measure $\tau_X(d_{i_k,j_k})$, $c_X(d_{i_k,j_k})$
  $\Delta := \mathbb{1}\{\delta_X = 1 \wedge \tau_X(d_{i_k,j_k}) \leq T \wedge c_X(d_{i_k,j_k}) \leq \mathbf{C}\}$
  $\Sigma_{\mathcal{A}} := \Sigma_{\mathcal{A}} + \Delta$
**end for**
Output $\hat{\mu}_{\mathcal{D}}(\mathcal{A}) := \Sigma_{\mathcal{A}}/N$

The intuitive meaning of this estimated flexibility can be reduced to

$$\hat{\mu}_{\mathcal{D}}(\mathcal{A}) = \frac{\text{\# of supported changes}}{\text{\# of posed changes}}. \tag{12}$$

Indeed, Eq. (12) takes exactly the form proposed in [8, 9], such that we are able to re-motivate it. The given results lead to an overall estimation flow of flexibility as is given in Algorithm 1. Due to the behavior of the empiric mean, the estimate becomes arbitrarily precise for $N \to \infty$:

$$\lim_{N\to\infty} \hat{\mu}_{\mathcal{D}}(\mathcal{A}) = \mathbb{E}\left\{\mathbb{1}\{d_{i,j} \in \mathcal{A}\}\right\} = \mu_{\mathcal{D}}(\mathcal{A}). \tag{13}$$

A measure defined this way will always be out of the interval $[0, 1]$, which follows the intuition of the flexibility degree defined in Section 3.5. However, in this case it is not guaranteed that 100% flexibility is reachable, because $\mathcal{D}$ might contain non-achievable demand changes in contrast with $\mathcal{A}^*$. In our running example in Fig. 2, the estimated flexibility values are shown in Fig. 2(d), where $\mathcal{D}$ contains the 30 possible demand changes from $\Omega^2$. One can observe that with constraints $(2, 4)$ all 30 demand changes can be achieved, hence, 100% network flexibility can be reached. Furthermore, the estimated flexibility value decreases as the constraints getting tighter, which meets our expectation.

### *4.3. Compound Network Flexibility*

Our metric so far measures and compares the flexibility of implementations $X$ and $Y$ with respect to a given aspect and parameter setting of the network. However, different settings might result in different relative ordering of $X$ and $Y$ in terms of their flexibility, questioning which one is more flexible. Here, we discuss how can we extend our measure to combine the flexibility value of different system set-ups into an overall metric, obtained e.g., with multiple $T$ and $\mathbf{C}$ constraints, in different network topologies, using different challenge profiles $\nu$ or considering different cost components in $\mathbf{C}$. Such a compound flexibility measure would allow us to draw better conclusions about the overall flexibility of different implementations. Note that, calculating the flexibility values for several challenge profiles or for a wide range of $(T, \mathbf{C})$ constraint combinations might be a tedious job, especially if we consider time as a continuous parameter. Furthermore, selecting meaningful constraint values assumes deep knowledge of network traffic characteristics and service level agreements, which is only obtainable through costly measurements or might not be accessible, respectively. Therefore, these calculations urge the need of benchmark flexibility values for given system set-ups, which can support the calculation of compound flexibility from a manageable number of relevant network scenarios. However, providing such benchmark settings is far from trivial.

Consider a set $\mathcal{K}$ of system set-ups, in which we want to measure the flexibility of two system implementations $X$ and $Y$, respectively. For example, the different set-ups could correspond to different network graphs[6] on which two flow embedding algorithms are evaluated, different time and cost constraints, or different aspects of the network [9] (e.g., resource scaling or topolgoy adaptation). Intuitively, we want to evaluate the overall performance, e.g., by averaging flexibility over set-ups, summing them up or considering the maximum and/or minimum achieved flexibility. For a consistent comparison over different system set-ups, it must be ensured that $\forall k, k' \in \mathcal{K}$ : $\hat{\mu}(\mathcal{D}_k) = \hat{\mu}(\mathcal{D}_{k'})$, i.e., that the same normalization is used. This is intuitive at first glance but often needs to be explicitly ensured, e.g., if $\mathcal{D}_k$ depends on the currently used set-up, such as the underlying topology (a 20-node network has more node-pairs – demand changes – than the 4-node topology in Fig. 2). When the normalization sets differ, the used measures will in fact vary for different set-ups $k$, $k'$, which can lead to inconsistent orderings. Obviously, averaging over such different measures might not give any insightful results. Therefore, in order to obtain meaningful results, we use the demand change set $\mathcal{D}_{\text{tot}} = \cup_k \mathcal{D}_k$ for normalization purposes, where each system set-up $k$ is challenged with its corresponding demand changes from $\mathcal{D}_{\text{tot}}$. As a result, flexibility can be evaluated with a consistent measure over all different set-ups, leading to the outcome of

$$\begin{aligned} \hat{\mu}_{\mathcal{D}_{\text{tot}}}(\mathcal{A}) &= \sum_{k=1}^{|\mathcal{K}|} \frac{\hat{\mu}(\mathcal{A}_k)}{\hat{\mu}(\mathcal{D}_{\text{tot}})} = \sum_{k=1}^{|\mathcal{K}|} \frac{\hat{\mu}(\mathcal{A}_k)}{\hat{\mu}(\mathcal{D}_k)} \frac{\mu(\mathcal{D}_k)}{\hat{\mu}(\mathcal{D}_{\text{tot}})}, \\ &= \sum_{k=1}^{|\mathcal{K}|} w_k \hat{\mu}_{\mathcal{D}_k}(\mathcal{A}_k), \\ \text{where } w_k &= \frac{\hat{\mu}(\mathcal{D}_k)}{\hat{\mu}(\mathcal{D}_{\text{tot}})}; \quad \sum_k w_k = 1. \end{aligned} \tag{14}$$

In words, the overall flexibility measure can be obtained as the weighted sum over the measures of each set-up, where the weights denote the relative sizes of the normalization sets $\mathcal{D}_k$. This conforms our intuitive definition, as a set-up which has to adapt to more demand changes should have a larger importance.

[6] Note that, depending on the model the network topology might be part of the system states, the implementation, or it can be considered as a demand as well.

## 5. Demand Changes

We defined flexibility of a communication network as the size of achievable demand changes from a given demand change sequence $\mathcal{D}$. Hence, the measured flexibility highly depends on the generated sequence $\mathcal{D}$ in Algorithm 1. As sampling sets in a meaningful way has a huge literature [32] and selecting demand changes from $\Omega^2$ is not different at all, here we only discuss the most important considerations on $\mathcal{D}$ regarding network flexibility.

### 5.1. Different Types of Demand Changes

In Sec 3.1 we briefly defined demands as external requirements that have to be met by the network. These requirements can lead to multiple types of different demand changes, i.e., new flow requests, topology extensions, link failures, etc. These demand changes may be based upon system parameters that can be modified by end users (such as source-destination pairs of embedded flows), or they can be unpredictable external changes (such as link failures). In general, the requirements leading to demand changes can be regarded as the instantaneous state of the external environment, in which the system operates, and can be modeled as a set of (possibly many) different parameters (e.g., modified by end users or through unpredictable events). This leads to a potentially high variability in the demand space $\Omega$ when performing flexibility analysis. Flexibility analysis applies for multiple demand change types without explicit mentioning.

### 5.2. Single-Shot Demand Changes Versus Sequence of Changes

We defined the demand change sequence as $\mathcal{D} = \{d_{i_1,j_1}, d_{i_2,j_2}, d_{i_3,j_3}, \dots, d_{i_N,j_N}\}$, where each $d_{i,j}$ is a change from an arbitrary demand $d_i$ to $d_j$, and the different elements of the sequence are independent from each other (e.g., if they were sampled uniformly random). However, there might be some scenarios where only the ability to serve a given *ordered sequence of demand changes* (e.g., adapting the network continuously to traffic load changes during a week including daily and weekly peeks) has importance rather than the ratio of achievable single-shot demand changes. In these scenarios the elements of the demand change sequence can be defined as an ordered sequence of demand changes $\mathbf{d}_{i_1,j_1} = (d_{i_1,i_2}, d_{i_2,i_3}, d_{i_3,i_4}, \dots, d_{i_{M-1},i_M=j_1})$, where the system adapts from state to state from $\mathcal{V}(d_{i_1})$, $\mathcal{V}(d_{i_2})$, $\mathcal{V}(d_{i_3}), \dots, \mathcal{V}(d_{i_M})$. When the network can perform all of these changes we say it is flexible, while the inability to adapt to only a single demand change within constraints $T$ and $\mathbf{C}$ from the sequence $M$ can be considered as a failure to serve $\mathbf{d}_{i_1,j_1}$ and shows the inflexibility of the system[7]. As with this definition the fulfilment of each $\mathbf{d}_{i,j}$ can be judged similarly to $d_{i,j}$, the network flexibility of $\mathcal{D} = \{\mathbf{d}_{i_1,j_1}, \mathbf{d}_{i_2,j_2}, \mathbf{d}_{i_3,j_3}, \dots, \mathbf{d}_{i_N,j_N}\}$ can be calculated in the same way as we have shown for the single-shot demand changes.

[7] Note that considering the sequence in $\mathbf{d}_{i_1,j_1}$ as single-shot demand changes, our original network flexibility metric would give an $M - 1/M$ flexibility value for $\mathbf{d}_{i_1,j_1}$ instead of 0.

### 5.3. How to Generate Demand Changes?

In most cases, the demand or state spaces of the system are infinitely large rendering *exact* flexibility measurement not feasible. Flexibility can only be estimated based on a selected, finite-length demand change sequence $\mathcal{D} \subset \Omega^2$, cf. Sec. 4.2. For some systems, there are well established benchmarking procedures, e.g. specified by IETF [33] or ETSI [34], which propose sets of specific tests (with specific numbers) to evaluate the devices under test. While such specific procedures can provide hints on how to generate demands or demand changes for estimating flexibility, they often focus on a limited part of the demand space only [35]. The question is how a meaningful demand change sequence can be generated so that the contained requests challenge the system to cover a significant part of the state space. Moreover, for a large fraction of systems such benchmarks do not exist. Thus, a more general procedure for generation of a meaningful demand change sequence is required. We suggest the following procedure:

1. Start with common input attributes and models for the use case, e.g., for a routing scenario, there are arrival times of flows, the flow sizes, and the sources and destinations of the flows.
2. Generate a list of demands by uniformly sampling the attributes from 1. and derive the subset of demands that are per se feasible in the system, i.e., in the empty system. Note that depending on the system the feasibility check might be an expensive operation. In case you compare multiple systems, select the feasible demands for every system and build the union of the sets.
3. Generate requests, i.e., demand transitions, by uniformly sampling from the set of feasible demands.
4. Incorporate additional knowledge about future behavior or particular interest in demand changes by means of utility (Sec. 6.3).

The example in Fig. 2 has a finite demand set of size 6 resulting in 30 possible demand changes. Thus, flexibility can be exactly measured. However, this does not hold for the case-studies presented in Sec.7 where we provide a more detailed example of the request generation process.

## 6. Cost of Flexibility Analysis

Flexibility is intuitively related to cost, yet in a contradictory manner. On the one hand, a more flexible system may lead to a better performance, hence increasing earnings or reducing cost. On the other hand, increasing flexibility requires additional resource usage, which can lead to higher costs. In this section we formally define the different flexibility-related cost components, which can be used by the operator in order to calculate the cost of a flexible network.

### 6.1. Action Cost

In Sec. 3.1 we introduced the concept of action cost as the effort of adapting to the demand change. A maximum value $\mathbf{C}$

of this cost is used in Eq. (1) to define the set of achievable demand changes $\mathcal{A}_X(T, \mathbf{C})$, upon which we define our flexibility measure. With the intention of providing a detailed analysis of the relation between cost and flexibility, we formally defined the *action cost* $C$ as a function mapping each demand change to a cost value:

$$C(d_{i,j}) : \Omega^2 \mapsto \mathbb{R}^+. \quad (15)$$

The cost associated to a demand change reflects all the additional resource usage which the network incurs during the action time, i.e., during adaptation. We can therefore distinguish two action cost subcomponents, which might be considered together or as different elements of cost vector $\mathbf{C}$:

- **Proaction cost** $C^P$: This is the cost of *deciding* how to adapt when there is a new demand change. In other words, it reflects the cost of interpreting the demand change and running the appropriate adaptation or optimization algorithms to select a new network state. As a result, we can express it as a function of the demand change $d_{i,j}$:

  $$C^P(d_{i,j}) : \Omega^2 \mapsto \mathbb{R}^+. \quad (16)$$

  Although it may be trivial in some cases, this proaction cost is always present after a demand change and it can be highly relevant when the adaptation implies solving a hard optimization problem.

- **Reaction cost** $C^R$: This is the cost of *performing* the selected adaptation, that is, the selected state change after a demand change. It reflects the additional resource usage required to change the state. As a result, we can express it as a function of the current and future states $\mathbf{s}_i$ and $\mathbf{s}_j$, respectively:

  $$C^R(\mathbf{s}_i, \mathbf{s}_j) : \mathcal{S}^2 \mapsto \mathbb{R}^+. \quad (17)$$

  Note that this cost component is absent if the demand change $d_{i,j}$ leads to no adaptation, either when $\mathbf{s}_i$ already fulfills $d_j$ (that is, $\mathbf{s}_i \in \mathcal{V}(d_j)$) or because no state fulfilling $d_j$ could be selected.

As a result, we can describe the action cost $C(d_{i,j})$ required by demand change $d_{i,j}$ as the combination of the proaction and the reaction cost:

$$C(d_{i,j}) = C^P(d_{i,j}) + C^R(\mathbf{s}_i, \mathbf{s}_j). \quad (18)$$

If we assume that demand changes occur randomly, $C(d_{i,j})$ is directly related to the probability distribution of the cost of the demand changes. Thus, once we have set a maximum action cost $\mathbf{C}$ to define our flexibility degree $\varphi_X(T, \mathbf{C})$, there will be, in general, a negative correlation between the average action cost and the flexibility of our system. Simply put, the higher the action cost, the lower the flexibility. This follows the intuition behind our measure, which penalizes adaptations that are too costly.

### 6.2. Preparation Cost

In general, the trend between flexibility and action cost cannot be extrapolated to the total cost of a system. For instance, in a full-mesh network with high capacity links, on which we could easily embed new flow demands, the action cost would be very low, but the cost of deploying and operating such an over-provisioned network will clearly surpass the action cost reduction in many practical cases. Therefore, apart from the action cost, which reflects the resource usage when a demand changes, we have to consider another cost component: the *preparation cost* $\mathcal{K}$. This is the cost of deploying and operating a flexible system, even if no demand changes occur. Intuitively, we expect that a flexible system, that is, a system that can dynamically adapt to multiple demands, may lead to higher deployment and operation costs. We can further distinguish between two cost subcomponents:

- **Provisioning cost** $\mathcal{K}^P$: This is the cost of building and deploying a flexible network, i.e., *the capital expenditures* (CAPEX). Depending on whether the cost analysis is being used to compare systems or not, this subcomponent can be defined as relative to a baseline CAPEX that is required to deploy a non-flexible network, or it can be the full CAPEX.

- **Readiness cost** $\mathcal{K}^R$: This is the cost of operating a flexible network, i.e., the *operating expenses* (OPEX). Following from this definition, the readiness cost $\mathcal{K}^R(\mathbf{s}, d)$ is a function of the system state $\mathbf{s}$ and reflects how well the network is adapted to the current demand $d$:

  $$\mathcal{K}^R(\mathbf{s}, d) : \mathcal{S} \times \Omega \mapsto \mathbb{R}^+. \quad (19)$$

  As a result, each possible state $\mathbf{s} \in \mathcal{V}(d)$ has an associated cost $\mathcal{K}^R(\mathbf{s}, d)$, such that it can be taken into account when selecting the state. It follows from this definition that the readiness cost is directly related to the instantaneous resource consumption of the network. For example, a state with a high energy consumption owing to frequent network monitoring, high link utilization, etc. features a larger readiness cost than that of a state with low energy consumption.

Fig. 3 shows a depiction of the different cost components. It can be seen that demand changes $d_{i,j}$ and $d_{j,k}$ (upper part of the figure) both incur proaction costs $C^P(d_{i,j})$ and $C^P(d_{j,k})$, respectively, as they have to decide whether adaptation is required or not. However, only $d_{j,k}$ causes a state change, which is charged by the reaction cost $C^R(\mathbf{s}_j, \mathbf{s}_k)$. Additionally, the readiness costs of operating the network in states $\mathbf{s}_j$ and $\mathbf{s}_k$ are $\mathcal{K}^R(\mathbf{s}_j)$ and $\mathcal{K}^R(\mathbf{s}_j)$, respectively. These components can be considered either different elements in $\mathbf{C}$ for the flexibility analysis with a different constraint on each component, or can be incorporated into a scalar with a single cost constraint (if it is possible to find a common basis of the different cost values).

### 6.3. Cost of Flexibility and Demand Utilities

In the flexibility evaluation discussed so far, every demand change was considered equally important. However, this might

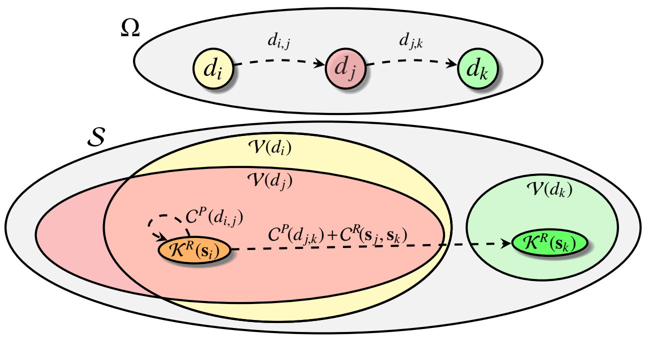



Figure 3: Illustration of readiness cost ($\mathcal{K}^R$), proaction cost ($C^P$), and reaction cost ($C^R$) for two demand changes. The system state $\mathbf{s}_i$ satisfies demands $d_i$ and $d_j$, that is, $\mathbf{s}_i = \mathcal{V}_X(d_i) = \mathcal{V}_X(d_j)$ for implementation $X$. Consequently, there is no reaction cost for demand change $d_{i,j}$.

not be true in real scenarios: fulfilling some demands may be crucial, whereas fulfilling others might have no effect. We can intuitively measure this "importance" with an utility function. For example, consider a utility function where only two demand changes $d_{i,j}$ and $d_{j,i}$ are actually assigned a non-zero value (e.g., we are only switching back and forth between demands $d_1$ and $d_6$ in Fig. 2(c)). Hence, any two system implementation $X$ and $Y$ capable of reacting to these two changes will have the same utility. In contrast, their flexibility can very well differ if $X$ is able to handle only these two demand changes while $Y$ is flexible enough to react within constraints $(T, \mathbf{C})$ to any possible demand change (flexibility of 0.06 and 0.87 in Fig. 2 for constraints $(1, 4)$, respectively). This intuitive idea of utility can be formally defined by using the aforementioned cost components, since the importance of fulfilling a demand can be matched with the cost of running in an unadapted state under that demand. Concretely, we consider that the utility function $u : \Omega\times\Omega \mapsto \mathbb{R}_+$ of demand change $d_{i,j}$ is

$$u(d_{i,j}) = \sum_{\mathbf{s}\notin\mathcal{V}(d_j)} \mathcal{K}^R(\mathbf{s}, d_j) - \sum_{\mathbf{s}'\in\mathcal{V}(d_j)} \mathcal{K}^R(\mathbf{s}', d_j), \tag{20}$$

that is, the difference of the accumulated readiness cost of not fulfilling the demand and fulfilling the demand. Note that the initial demand $d_i$ does not play any role in Eq. (20), but it is included as an input variable to better merge it with the previous formulation, as will become clear in the following.

We will now show how the utility of different demands can be incorporated in our flexibility metric. Given a utility function $u$ that reflects how valuable the ability to change a certain demand from one to another is assumed. In many cases, the utility of a demand change is mostly impacted by the utility of its target demand, which can be appropriately modeled. Then, for given constraints $(T, \mathbf{C})$, we define the utility of flexibility as:

$$\overline{\mu}(\mathcal{A}) = \int_{d_{i,j}\in\mathcal{A}} u(d_{i,j}) d\mu. \tag{21}$$

Eq. (21) reflects the total utility enabled by allowing the switching between demands.

Estimation of utility can be established analog to estimation of flexibility, by designing a demand change sequence $\mathcal{D}$ with profile $\nu$ and defining utility of flexiblity as

$$\begin{aligned} \overline{\mu}^{\nu}(\mathcal{A}) &= \int_{d_{i,j}\in\Omega^2} u(d_{i,j})\nu(d_{i,j})\mathbb{1}\{d_{i,j}\in\mathcal{A}\}d\mu \\ &= \mathbb{E}\left\{u(d_{i,j}) \mid d_{i,j}\in\mathcal{A},\mathcal{D}\right\}. \end{aligned} \tag{22}$$

When $\nu$ is uniform, $\overline{\mu}^{\nu}(\mathcal{A}) = \overline{\mu}(\mathcal{A})/\mu(\mathcal{D})$ holds, which again allows comparison among different implementations. An estimation process can then be created by replacing Line 9 in Algorithm 1 with $\Sigma_{\mathcal{A}} := \Sigma_{\mathcal{A}} + u(d_{i_k,j_k})\cdot\Delta$.

Finally, note that the exact flexibility for non-uniform challenge profiles in Section 4.1 can be expressed as utility if $u(d_{i,j})$ is selected as the relative frequency $\nu(d_{i,j})$ of occurrence in the demand change sequence $\mathcal{D}$.

## 7. Experimental Results

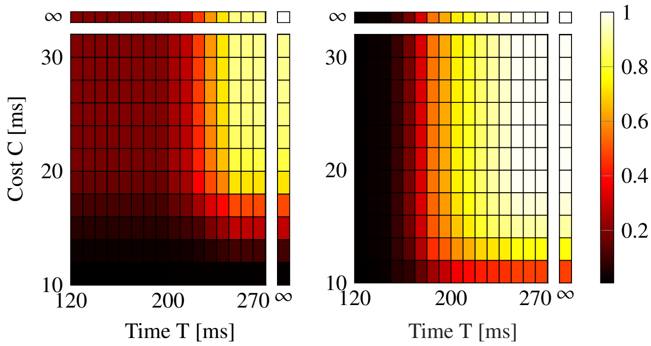
In this section we show how the proposed network flexibility framework can be applied for the use-cases of controller placement and resilience, and discuss some additional insights a quantitative flexibility analysis can bring compared to qualitative claims.

### *7.1. Dynamic Controller Placement in SDN*

Consider an SDN network described by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, that is managed by $1, 2, .., |\mathcal{N}|$ controllers. The SDN network faces constantly changing flow demand, i.e., flows are entering and leaving the network over time. In order to react to the changing flow demands, the controllers program the network switches in a centralized fashion to serve the flow demands in the network, i.e., to establish connections between the sources and the destinations of the flow demands. Generally, SDN is claimed as flexible in literature due to its ability to control data plane devices via a standardized interface from a logically centralized control plane. However, the separation between control and data plane can introduce additional latencies, i.e., flow setup times; the controllers need to send control messages over the network to the data plane devices [8].

We model this set-up as follows. The network state is given by the current paths on which the flows are embedded, as well as the positions of the controllers in the network. We assume that the controller positions are optimized towards a minimum average flow set-up time at each instant. Although someone might claim that this comparison is not fair in terms of achievable control plane latency (existing studies have already shown that multiple controllers might achieve lower control plane latencies), we argue that it can still be a realistic scenario. First, it should be noted that optimization targets and cost aspects can always be use case specific. Hence, we argue that minimizing control plane latencies for one and multiple controllers trades-off other cost aspects when comparing those architectures: for instance, multiple controllers might induce additional control plane messages or operators need to pay more due to additional control software licenses.

In order to fully exploit the flexibility of softwarized networks, the network might react with a controller migration, e.g.,




(a) Implementation $X_1$ (b) Implementation $X_3$

Figure 4: Flexibility comparison of different system implementations (1 or 3 controllers) with varying constraints $T$ and $\mathbf{C}$. Each square represents the average flexibility of 50 simulation runs on 100 size demand change sequence. If the time or cost constraint is relaxed to infinity, then the cost-efficiency and reactivity property of flexibility is emphasized, respectively (see Def. 4).

a simple virtual machine migration when the flows are changing. The idea is to migrate the controllers towards locations that are closer to the current flow demands, i.e., their source and destination nodes.

Formally, a demand $d_i \in \Omega$ represents a set of flows, specified by a set of triplets $(s, t, r)$, each indicating source $s \in \mathcal{N}$, target $t \in \mathcal{N}$ and the requested number of flows $r \in \mathbb{N}_+$ between $s$ and $t$. The set of all possible new flows is defined as $\Omega = \mathcal{N} \times \mathcal{N} \times \mathbb{N}_+$. Assume a time-slotted network operation, such that in each slot a set of new flows $d_i$ are in the network and their forwarding paths need to be set up by the control plane. Because the network optimizes the controller positions, the associated state $\mathcal{V}_X(d_i)$ is the optimal controller placement that minimizes the average flow setup time given the demand $d_i$, as well as the implemented paths. Because the used paths and controller positions are uniquely defined, $\mathcal{V}_X(d_i)$ contains a single element $\forall d_i \in \Omega$. We can model different numbers of controllers as different system implementations $X_n$ and realize that $\mathcal{V}_{X_n}(d_i)$ of different system implementations are completely different, respectively. According to our flexibility definition, they can be compared due to the same demand sets.

To react to a demand change $d_{i,j}$, the control plane first needs to find the optimal placement $\mathcal{V}_X(d_j)$ that induces a certain cost $\mathbf{c}_X(d_{i,j}) = \mathbf{c}_X(d_j)$, defined as the optimal average flow setup time with $\mathcal{V}_X(d_j)$. If the current optimal placement $\mathcal{V}_X(d_j)$ is different from the previous optimal placement $\mathcal{V}_X(d_i)$, a certain delay $\tau_X(d_{i,j})$ is induced to record the control plane adaptation time [9]. The adaptation is counted as a success if and only if $\mathbf{c}_X(d_{i,j}) < \mathbf{C}$ and $\tau_X(d_{i,j}) < T$.

The numerical results of this use case are generated using the flow of Algorithm 1 on the Abilene network topology. Fig. 4 shows the heatmap of the flexibility value with different time and cost constraint values. $X_1$ and $X_3$ represent the system implementation with 1 controller and 3 controllers, respectively. In general, the flexibility value of each system implementation increases when $T$ or $\mathbf{C}$ increases, meaning that relaxed constraints allow more adaptation successes. Besides, $X_3$ is more flexible for most of the $\mathbf{C}$ and $T$ constraint combinations. However, a counter-intuitive observation, is that for tight $T$ and relaxed $\mathbf{C}$, $X_1$ is more flexible than $X_3$ (upper-left corner of the heatmap). This is because in this region, the optimal placement of $X_1$ does not vary much but a fixed position is optimal for a majority of the demands, whereas that of $X_3$ changes more frequently and induces adaptation time that violates the time constraint. This effect corresponds to the reactivity and cost-efficiency perspective, which are given in the lines indicated with $\infty$. Comparing these, we can claim that $X_3$ is more cost-efficient with respect to the control plane latency than $X_1$. However, for low adaptation time constraints $X_1$ is more reactive, as it does not invest the time to re-optimize its state. Finally, for relaxed cost and time constraints, both systems are equally adaptive, as is seen in the upper right corner. In principle, $X_1$ and $X_3$ are able to serve any of the demand changes.

Beside these conclusions, we also would like to mention that this use case demonstrates another interesting aspect of our flexibility analysis: the potential drawbacks when blindly optimizing for specific costs. As our simulations demonstrate, minimizing the control plane latency renders $X_3$ to be less flexible than $X_1$ for tight adaptation times. However, this drawback could have been avoided by optimizing both $X_1$ and $X_3$ for the same latency target, i.e., satisfying control plane latency instead of minimizing for it. In such case, $X_3$ should have shown the same flexibility as $X_1$. We take this observation also as a motivation to even optimize for flexibility [36] – which should avoid such optimization pitfalls.

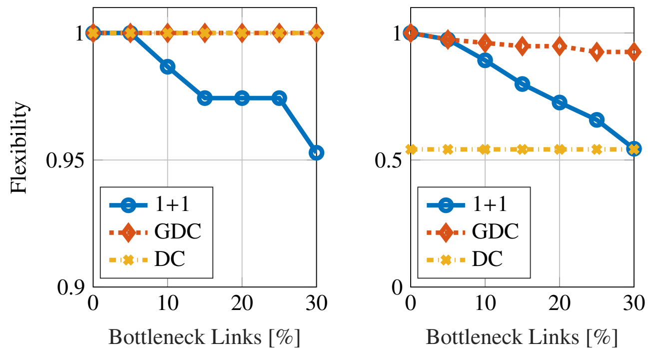
### 7.2. Protection Routing

In this use case we compare different protection approaches regarding their network flexibility. Each flow is routed from source to target node with minimum bandwidth cost, and in addition, the flows should be protected against single-link failures (e.g., with a disjoint path-pair). The protection methods are evaluated in a network where some flows were already embedded, which means that the free capacity on the links is changing after flow set-up or tear-down events. Hence, we consider the network topology with the currently available free capacities on its links as the input for our protection routing problem. Flexibility is evaluated as the ability to provide a minimum availability (i.e., single link failure resilient routing) under different underlying free capacities. In particular, we consider three different protection schemes [30], namely:

- $1+1$ dedicated path protection, which requires a disjoint path-pair between the source $s_i$ and target node $t_i$. The whole user data, split into two parts $AB$ and requiring two capacity units, is sent along both paths.

- *Diversity coding (DC)*, requires three disjoint paths between the source and target node. On each path, only a single capacity unit is required, as data halves $A$, $B$ and redundancy data $A\ XOR\ B$ is sent along the paths.

- *Generalized diversity coding (GDC)* [30], which combines the previous two approaches in order to circumvent their weaknesses, namely the *bottleneck links* with one unit of free capacity for $1+1$ (it can use only links with 2




(a) Average Node Degree 5.7 (b) Average Node Degree 2.9

Figure 5: 80 node networks with increasing number of bottleneck links.

capacity units), and the topological limit of DC (lack of three disjoint paths), respectively.

We can formally argue that GDC is the most adaptive approach from the three according to Observation 3 in Sec. 3.3, as it can realize each flow which either 1 + 1 or DC can (as they are special cases of GDC) and even more owing to its adaptable protection structure. In order to show this quantitatively, in our experimental evaluation we took a snapshot from the considered topology, and represent the state of the network as the free capacities on the links and the currently embedded protection paths (denoted as $d_0$). A demand is given as a source target pair, $d_i = (s_i, t_i)$. The demand change sequence $\mathcal{D} = \{d_{0,1}, d_{0,2}, \ldots, d_{0,N}\}$ contains single-shot demand changes. In our simulations $N = |\mathcal{N}| \cdot (|\mathcal{N}| - 1)$, i.e., a flow request is given between each source-target pair, which means $N = 6320$ demand changes in our 80 node topologies. Time and cost constraints are relaxed to infinity here, as we are interested in the adaptability perspective (i.e., acceptance ratio of new flow requests).

Fig. 5 contains our simulation results, which were obtained by decreasing the free link capacities (i.e., increasing the number of bottleneck links) on two topologies with different average node degrees, and challenging the topology with $\mathcal{D}$ in each snapshot. One can observe, that in the denser network with high node degree in Fig. 5(a), where three disjoint paths exist between the node-pairs more often, diversity coding is equally as flexible as GDC. Furthermore, as the number of bottleneck links increases (i.e., with higher traffic load), the number of protectable node-pairs for 1 + 1 decreases, making it the less flexible choice. On the other hand, in the sparser network in Fig. 5(b) only between the 54% of node-pairs exist three disjoint paths, which makes DC the less flexible choice up to 30% of bottleneck links. In the scenario with 30% bottleneck links the flexibility of 1 + 1 decreases to 54% as well. Hence, indeed in this limited situation GDC is 38% more flexible than any of its counterparts. In general, GDC is always at least as flexible as any of the other two schemes, which backs up our intuition.

Finally, we investigate the compound network flexibility of the two topologies. As we have the same $N$ number of demand changes in both 80-node networks in $\mathcal{D}$, the two settings are considered to be equally important; hence, the overall flexibility can be calculated simply as the average flexibility of the two system set-ups. Therefore, 1 + 1 outperforms DC up to 25% of bottleneck links, while DC is bit more flexible if the network is loaded (both around 0.75). However, the compound flexibility of GDC is always above 0.95. We also note that the additional flexibility of GDC does not induce more bandwidth cost (i.e., $\mathcal{K}^P$), as the capacity consumption of GDC always lower or equal to the better from 1 + 1 and DC. However, the XOR coding might have some additional equipment or software cost (i.e., $\mathcal{K}^R$), and also might increase the reaction time (i.e., $C^R$) owing to the deployment of multiple sub-flows between the source and the target nodes.

## 8. Conclusions

In this work we introduced a framework for evaluating network flexibility that can help researchers and network operators to quantitatively support the flexibility advantage of new networking approaches. We showed that relating network flexibility to the size of the achievable demand change set fulfills all of our expectations about the intuitive usage of the term, and allows a quantitative flexibility comparison and cost analysis of different networks systems and approaches. We further demonstrated that any mathematical measure function can, in general, serve as a flexibility metric. Thus, we introduced multiple normalizations which can give our flexibility value an intuitive meaning (e.g., flexibility degree). Building on these analytical considerations, a measurement flow for empirical assessment of flexibility was proposed. We extended the flexibility framework with a cost of flexibility analysis and different demand utilities as well, which helped to tailor the theoretical model to the characteristics of real network environments. We applied the measure to the use cases of flow embedding in software-defined networks and dedicated path protection, and showed that the novel measure reflects our intuition on flexibility, as well as enables concise argumentation.


## Acknowledgments

This work is part of a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 647158 - FlexNets "Quantifying Flexibility for Communication Networks"). The work of P. Babarczi was supported in part by Project no. 134604 that has been implemented with the support provided by the National Research, Development and Innovation Fund of Hungary, financed under the FK_20 funding scheme, and in part by the Alexander von Humboldt Foundation.


## References


## References



[1] M. He, A. M. Alba, A. Basta, A. Blenk, W. Kellerer, Flexibility in softwarized networks: Classifications and research challenges, IEEE Communications Surveys Tutorials (2019) 1–39doi:10.1109/COMST.2019.2892806.

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, Openflow: enabling innovation in campus networks, ACM SIGCOMM Computer Communication Review 38 (2) (2008) 69–74.

[3] N. Nikaein, E. Schiller, R. Favraud, K. Katsalis, D. Stavropoulos, I. Alyafawi, Z. Zhao, T. Braun, T. Korakis, Network store: Exploring slicing in future 5G networks, in: Proc. 10th Int. Wkshp. on Mobility in the Evolving Internet Archit., ACM, 2015, pp. 8–13.

[4] B. Han, V. Gopalakrishnan, L. Ji, S. Lee, Network function virtualization: Challenges and opportunities for innovations, IEEE Commun. Mag. 53 (2) (2015) 90–97.

[5] S. Kassing, A. Valadarsky, G. Shahaf, M. Schapira, A. Singla, Beyond fat-trees without antennae, mirrors, and disco-balls, in: Proc. of SIGCOMM '17, ACM Press, New York, NY, USA, 2017, pp. 281–294. doi:10.1145/3098822.3098836.

[6] A. Singla, Fat-FREE Topologies, in: Proceedings of the 15th ACM Workshop on Hot Topics in Networks - HotNets '16, ACM Press, New York, New York, USA, 2016, pp. 64–70. doi:10.1145/3005745.3005747.

[7] M. Klügel, M. He, W. Kellerer, P. Babarczi, A mathematical measure for flexibility in communication networks, in: Proc. IFIP Networking Conference (IFIP Networking), 2019, pp. 1–9.

[8] M. He, A. Basta, A. Blenk, W. Kellerer, How flexible is dynamic SDN control plane?, in: Proc. of IEEE Conference on Computer Communications Workshop SWFAN, 2017, pp. 689–694. doi:10.1109/INFCOMW.2017.8116460.

[9] W. Kellerer, A. Basta, P. Babarczi, A. Blenk, M. He, M. Klügel, A. M. Alba, How to measure network flexibility? A proposal for evaluating softwarized networks, IEEE Communications Magazine 56 (10) (2018) 186–192. doi:10.1109/MCOM.2018.1700601.

[10] P. H. Brill, M. Mandelbaum, On measures of flexibility in manufacturing systems, Int. J. Prod. Res. 27 (1989) 747–756. doi:10.1080/00207548908942584.

[11] A. K. Sethi, S. P. Sethi, Flexibility in manufacturing: A survey, Int. J. Flexible Manuf. Syst. 2 (4) (1990) 289–328. doi:10.1007/BF00186471.

[12] S. Benjaafar, Modeling and analysis of flexibility in manufacturing systems, Ph.D. thesis, Purdue University (1992).

[13] I. J. Chen, C.-H. Chung, An examination of flexibility measurements and performance of flexible manufacturing systems, Int. J. Prod. Res. 34 (2) (1996) 379–394. doi:10.1080/00207549608904909.

[14] S. K. Das, The measurement of flexibility in manufacturing systems, Int. J. Flexible Manuf. Syst. 8 (1) (1996) 67–93. doi:10.1007/BF00167801.

[15] R. E. Giachetti, L. D. Martinez, O. A. Sáenz, C.-S. Chen, Analysis of the structural measures of flexibility and agility using a measurement theoretical framework, Int. J. Prod. Econ. 86 (1) (2003) 47–62. doi:10.1016/S0925-5273(03)00004-5.

[16] T. Marschak, R. Nelson, Flexibility, Uncertainty and Economic Theory, Metroeconomica 14 (1-2-3) (1962) 42–58. doi:10.1111/j.1467-999X.1962.tb00293.x.

[17] M. Mandelbaum, Flexibility in decision making: an exploration and unification, Ph.D. thesis, University of Toronto (1978).

[18] J. Lasserre, F. Roubellat, Measuring decision flexibility in production planning, IEEE Trans. Autom. Control 30 (5) (1985) 447–452. doi:10.1109/TAC.1985.1103984.

[19] M. Mandelbaum, J. Buzacott, Flexibility and decision making, Eur. J. Oper. Res. 44 (1) (1990) 17–27. doi:10.1016/0377-2217(90)90310-8.

[20] S. Benjaafar, T. L. Morin, J. J. Talavage, The strategic value of flexibility in sequential decision making, Eur. J. Oper. Res. 82 (3) (1995) 438–457. doi:10.1016/0377-2217(93)E0210-O.

[21] W. Golden, P. Powell, Towards a definition of flexibility: in search of the Holy Grail?, Omega 28 (4) (2000) 373–384. doi:10.1016/S0305-0483(99)00057-2.

[22] R. D. Shachter, M. Mandelbaum, A measure of decision flexibility, in: Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., 1996, pp. 485–491.

[23] A. H. Eden, T. Mens, Measuring software flexibility, IEEE Proceedings of Software 153 (3) (2006) 113–125.

[24] A. Ulbig, G. Andersson, On operational flexibility in power systems, in: Proceedings of the 2012 Power and Energy Society General Meeting, IEEE, 2012, pp. 1–8.

[25] A. D. Wissner-Gross, C. E. Freer, Causal entropic forces, Phys. Rev. Lett. 110 (2013) 168702. doi:10.1103/PhysRevLett.110.168702.

[26] P. Kalmbach, J. Zerwas, P. Babarczi, A. Blenk, W. Kellerer, S. Schmid, Empowering self-driving networks, in: ACM SIGCOMM Workshop on Self-Driving Networks (SelfDN), 2018, pp. 8–14. doi:10.1145/3229584.3229587.

[27] K. Kar, M. Kodialam, T. V. Lakshman, Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications, IEEE Journal on Selected Areas in Communications 18 (12) (2000) 2566–2579. doi:10.1109/49.898737.

[28] D. P. Siewiorek, J. J. Hudak, B. . Suh, Z. Segal, Development of a benchmark to measure system robustness, in: FTCS-23 The Twenty-Third International Symposium on Fault-Tolerant Computing, 1993, pp. 88–97. doi:10.1109/FTCS.1993.627311.

[29] Q. Yin, P. Kong, H. Wang, Quantitative robustness metric for qos performances of communication networks, in: 2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications, 2006, pp. 1–5. doi:10.1109/PIMRC.2006.254090.

[30] P. Babarczi, J. Tapolcai, A. Pašić, L. Rónyai, E. R. Bérczi-Kovács, M. Médard, Diversity coding in two-connected networks, IEEE/ACM Trans. Netw. 25 (4) (2017) 2308–2319. doi:10.1109/TNET.2017.2684909.

[31] T. Tao, An introduction to measure theory, American Mathematical Society Providence, RI, 2011.

[32] D. Stutzbach, R. Rejaie, N. Duffield, S. Sen, W. Willinger, On unbiased sampling for unstructured peer-to-peer networks, IEEE/ACM Transactions on Networking 17 (2) (2009) 377–390. doi:10.1109/TNET.2008.2001730.

[33] S. Bradner, J. McQuaid, Benchmarking methodology for network interconnect devices.

[34] G. ETSI, Etsi gs nfv-tst 002 v1. 1.1-report on nfv interoperability testing methodology (2016).

[35] Etsi tr 138 912: Study on new radio (nr) access technology (3gpp tr 38.912 version 15.0.0 release 15), Tech. rep. (09 2018).

[36] M. He, A. Varasteh, W. Kellerer, Toward a flexible design of sdn dynamic control plane: An online optimization approach, IEEE Transactions on Network and Service Management 16 (4) (2019) 1694–1708. doi:10.1109/TNSM.2019.2935160.