CrossMark

Study of Network Migration to New Technologies Using Agent-Based Modeling Techniques

**Tamal Das · Marek Drogon · Admela Jukan ·
Marco Hoffmann**

Abstract Conventionally, network migration models study competition between emerging and incumbent technologies by considering the resulting increase in revenue and associated cost of migration. We propose to advance the science in the existing network migration models by considering additional critical factors, including (1) synergistic relationships across multiple technologies, (2) reduction in operational expenditures as a reason to migrate, and (3) implications of local network effects on migration decisions. To this end, we propose a novel agent-based migration model considering these factors. Based on the model, we analyze the case study of network migration to two emerging networking paradigms, i.e., IETF Path Computation Element (PCE) and Software-Defined Networking (SDN). We validate our model using extensive simulations. Our results demonstrate the synergistic effects of migration to multiple complementary technologies, and show that a technology migration may be eased by the joint migration to multiple technologies. In particular, we find that migration to SDN can be eased by joint migration to PCE, and that the benefits derived from SDN are best exploited in combination with PCE, than by itself.

T. Das (✉) · M. Drogon · A. Jukan
Institut für Datentechnik und Kommunikationsnetze, TU Braunschweig, Brunswick, Germany
e-mail: t.das@tu-bs.de

M. Drogon
e-mail: m.drogon@tu-bs.de

A. Jukan
e-mail: a.jukan@tu-bs.de

M. Hoffmann
Nokia Networks, Munich, Germany
e-mail: marco.hoffmann@nsn.com

1 Introduction

Technical novelties in conjunction with economic factors decide the fate of an emergent technology, protocol, standard or product in present-day communication networks. Networks are constantly migrating to new technologies and services, not only driven by the growth of subscribers base and application demand, but also new technological advances. The migration is typically a gradual transition over time, requiring the interoperability and integration of different network applications, technologies and protocols. For instance, though the first IPv6 specification was released in 1998 [1], the migration process is still ongoing with only 0.2 % of current Internet traffic being IPv6-compliant [2]. On the other hand, IP backbones today migrate to router interfaces of a higher capacity at a much faster pace. A typical carrier IP network is re-planned and increased capacity every 12–18 months, so that maximum utilization at peak traffic loads is never higher than approximately 30–40 % [3]. Thus, there is no doubt that understanding the strategy and the investments for network migrations, as well as the expected revenue, network operation expense and user growth are at the heart of every network migration decision.

Technology adoption has been significantly investigated in the literature using various migration models. However, a few increasingly important factors have not received enough attention. First, the majority of previous studies model technology migration in isolation, disregarding the effect of co-existing technologies in the market. Such studies, thus, do not account for the synergistic relationships that may exist across technologies, which as a result, may either facilitate or impede the adoption of a new technology. For instance, an offering of VPN services with guaranteed QoS may result in a higher revenue, when combined with automated network management systems. Second, the majority of migration models are based on the capital expenditures (CapEx) required to purchase the new technology. However, technology migration often results in tangible reduction of operational expenditures (OpEx) that is gained over time, which is typically neglected in the current models. Finally, human decisions are subject to influence of the social and behavioral factors involved in the process of migration. For example, although bandwagon effect [4] (or network effects) plays a significant role in the adoption of a technology, over and beyond its technological merits, it is rarely captured in migration models.

In this paper, we propose a generic agent-based model to explore network migration to multiple new *complementary* technologies—technologies whose simultaneous migration is expected to provide greater rewards than the sum of the rewards derived from their isolated migrations. In addition to CapEx, our model also incorporates the difference in the OpEx incurred pre- and post-migration, which significantly affects an agent's decision to migrate. In the proposed model, an agent also incorporates its estimates of its neighbor's decision to migrate, in its own migration decision. An agent estimates its *effective migration coefficient* (see Sect. 4.2) based on the migration decisions of all neighboring agents within its *circle of influence* (see Sect. 4.2). In this estimation process, it considers the mutual effect of a neighbor's migration state as inversely proportional to the distance between them.

We accomplish this by means of both deterministic and probabilistic heuristics. Our results confirm that a technology migration may be eased by the joint migration of a complementary technology that is more likely to be adopted [5].

To validate our proposed model, we analyze the case study of optimal path computation with joint migration to two emerging networking paradigms, i.e., IETF Path Computation Element (PCE [6]) and Software-Defined Networking (SDN [7]), respectively. The assumed network is a typical multi-vendor and multi-administration network, where separate *network islands* of routing systems need to cooperate to provision an end-to-end connection, and are subject to migration decision pertaining to PCE, SDN, or both. PCE enables optimal path computation across network islands, an improved price/performance ratio, while, at the same time simplifying path computation operations [8]. All these benefits added together attract considerably more users (and in turn traffic) to the network. Exchanges between PCE and network elements, though standardized, are limited to PCEP messages, and thus a PCE cannot setup the computed paths itself. To overcome this limitation, the network operator may decide to migrate to another technology, say, SDN, which facilitates configuration of all the network elements, and thereby helps in setting up the computed paths. Moreover, combining a stateful PCE with OpenFlow provides an efficient solution for operating transport networks [9]. Thus, there is an implicit correlation between the deployment of PCE and SDN in a network, which make these two technologies an interesting and practically relevant case study.[1]

We simulate various experiments studying the effect of a variety of factors affecting the network *migration profile*, such as equi-cost routing, strategy estimation approaches, early adopters, coupling coefficient, network topology, etc. Here, by *migration profile*, we mean the progress of the network-wide migration captured by monitoring the number of migrated nodes throughout an experiment. Our results indicate that the migration to SDN can be promoted by several factors, namely, (a) in combination with a widely-accepted complementary technology such as PCE, (b) early adopters, (c) an agent's ability to predict its neighbor's decisions to migrate to either of the technologies. Although our proposed generic model (in Sect. 4.1) is broadly applicable to diverse scenarios, our results (in Sect. 5) have been derived for the particular case study involving PCE and SDN.

Our paper is organized as follows. Section 2 discusses the related literatures and puts our contributions into perspective, while Sect. 3 provides an overview of the technologies that we study, namely PCE and SDN. Section 4 defines our generic multi-technology migration model and its application to the case study of PCE/SDN. Section 5 discusses the simulation framework to evaluate our network migration model, and highlights its various aspects using the empirical results, while, Sect. 6 presents some concluding remarks.

---

[1] In Sect. 3.3, we discuss at length the inter-relationships between PCE and SDN. However, here we would like the reader to note that SDN and PCE are not really fundamentally re-thinking the technologies in the networking landscape. For instance, SDN is more like a natural evolution, incorporating technologies that existed already for quite some time. Just like, before OpenFlow, there was SNMP.

2 Related Work and Our Contribution

In this section, we summarize the previous research in the domain of our work, and highlight our contributions in this paper.

2.1 Technology Adoption and Diffusion Modeling

Technology adoption and innovation diffusion have been extensively studied in the literature. The term *diffusion process* was first coined to track the purchase patterns of hybrid seed corn by farmers [10], and have since found widespread applications in various context, including technology adoption [11, 12]. This idea was further generalized to describe spread of new ideas and technologies in different cultures [13]. Bass proposed his first diffusion model [14] describing how new products get adopted in a population. This model has been revised over the years [15, 16] and still remains one of the most widely used models till date.

2.2 System Dynamics and Agent-Based Models

Network migrations are typically studied using *system-dynamics* [17, 18] and *agent-based* models [19, 20]. The former approach is based on aggregate system-wide properties, while, in the latter approach, simple rules of mutual interaction between agents govern the evolution of the system. In the system dynamics approach, the migration problem is treated as a dynamic system in [17, 18], where the rate of migration depends on the existing number of migrated agents in the system, according to the traditional diffusion theory of innovation [14]. On the other hand, in an agent-based approach [19, 20], the system consists of an ensemble of agents, each trying to increase its own utility. For example, in [21], the migration to secure BGP is studied as a series of decisions by each domain to adopt the technology, based on the inter-domain routing and the deployment of secure BGP in other domains. Both approaches demonstrate that the cumulative number of migrations increase over time assuming a '*S*'-shaped (or sigmoidal) curve, implying that a majority of migrations is triggered in a short time interval [22]. Despite comparable results, an agent-based approach is preferred over system dynamics, when the mutual interactions between agents in the system is non-uniform, for example, when an agent does not interact uniformly with *all* other agents, but only with those in its local neighborhood. Hence, we choose agent-based modeling over system dynamics approach for our study in this paper.

2.3 Single Versus Multiple Migrations

The network migration problem has typically been studied for a single technology or protocol (e.g., IPv6 [23, 24] or secure BGP [21, 25]), where it is assumed that an emerging protocol/technology *replaces* an incumbent protocol/technology. For example, in case of IPv6, the models assume that the domain operates either in IPv4 or migrates *fully* to IPv6, at which point it operates only with IPv6. Even when multiple protocols are considered, such as S-BGP and soBGP [25], there is only a

single prevalent protocol, and a decision is made by an agent to adapt to only one of the competing protocols. Sohn et al. propose an economic evaluation model for a particular aspect of migration, namely, joint development and standardization of correlated technologies [26]. Thus, although majority of the prior migration studies deal with migration of a single technology, the novelty of our model is in considering multi-technology migrations.

2.4 CapEx and OpEx Considerations

An agent's migration decision is often considered to be solely based on the CapEx involved. OpEx was recently introduced in cost analysis of migration research to precisely estimate the cost that the migration to a technology requires and compare the alternatives [27]. However, the game-theoretic modeling of migration have not yet considered it [17, 18, 21, 23–25]. In this paper, we consider both CapEx and OpEx in an agent's decision to migrate. In our work, the OpEx reflects an assumption that the proposed new system will include a level of automation into the network that alleviates human efforts, resulting in its overall cost reduction. Our model is thus novel in considering both revenue increase and OpEx reduction, resulting from migration, as the factors affecting an agent's decision to migrate.

2.5 Our Approach and Contribution

This paper extends our previous work on agent-based modeling of network migration to new technologies [28]. In this paper, we improve our CapEx, OpEx and revenue functions used in the network migration model. In particular, we take into consideration that revenue of a network island follows economies of scale, i.e., every subsequent unit of traffic incurs a lesser cost to the network operator than the previous. In contrast to [28], we differentiate between the OpEx functions in the unmigrated and migrated states. In migration to technologies such as PCE, the derived benefit depends not only on the particular network island deploying it, but also on the PCE deployments in each of the intermediate network islands between a given source and destination. As a result, it is important to study the technology migration of a network island with reference to that of its neighbors. Unlike [28], where only an agent's *immediate* neighbors were considered to affect the migration choices of the agent in question, we now extend this effect to include even *distant* neighbors within an agent's *circle of influence* (defined in Sect. 4.2). The mutual effect of an agent's migration choice on another is weighted by the reciprocal of the distance between them, restricted to a threshold distance (beyond which the effect is considered negligible). This inverse proportionality with distance results from the fact that for PCE-like technologies, the derived benefits are insignificant if an agent adopts it in isolation, and increase if neighboring agents adopt it too. Hence, the farther an unmigrated agent is from a migrated agent, the lesser is its incentive to migrate. We also introduce the notion of *coupling coefficient* to effectively capture the degree to which two complementary technologies couple with each other. We propose two novel heuristics for an agent to estimate the strategies of its neighboring agents in the immediate future, which, in turn, plays a significant role in

the migration decision of the agent in question. Another unique contribution is in definition of an agent's *payoff* from a transition, which is based on its CapEx, OpEx and revenue functions. An agent migrates only if such a transition results in a positive payoff for itself.

To validate our model, we consider a novel case study of multi-vendor enterprise network, considering the revenue of a network to vary with the volume of traffic it transits for its customers. To this end, we consider simultaneous and correlated deployment of an automated network management system for path computation (PCE) as well as a programmable network configuration with SDN controllers, such as based on OpenFlow [29]. We show that the proposed model is applicable for scenarios, where competing network solutions (such as multi-vendor environments) collaborate and compete at the same time for path setup, while aiming at maximum utilization in course of its operation. As is well-known, inter-operability of multi-vendor network islands remains a challenge, and a migration to standardized and programmable automated systems is an ongoing open problem in carrier networks [30].

3 Case Study of PCE and SDN: Background and Reference Architecture

In this section, we present an overview of the technologies, namely PCE and SDN, which we later study using our network migration model. We compare these two technologies on grounds of path computation and provisioning of a connection request across multiple network islands in a multi-vendor enterprise network based on emerging carrier-Ethernet (connection-oriented) networks.

3.1 Technology Overview

PCE is a network-wide centralized server that receives path computation requests from Path Computation Clients (PCC), and computes optimal constrained end-to-end paths within a network island. The PCE can reduce the computation overhead and optimize resource utilization by computing optimal paths. A major advantage of the PCE architecture is its ability to compute optimal paths across multiple network islands using the Backward Recursive Path Computation (BRPC) mechanism [31]. In the BRPC mechanism, PCEs in different islands along a pre-defined chain progressively compute a Virtual Shortest Path Tree (VSPT) from the destination to the source, in order to compute the optimal end-to-end path. In absence of PCE, network islands use Interior Gateway Protocols (like Open-Shortest-Path-First and Routing-Information-Protocol) and Exterior Gateway Protocols (like Border Gateway Protocol) to compute paths by means of predefined routing table entries.

Software-defined networking is an emerging networking architecture that facilitates programmability of the network control plane and its separation from the data plane [7]. It provides a centralized control interface to all the network elements that support SDN protocols, such as Open Flow [29], which helps in quick experimentation, reconfiguration, optimization, and monitoring of switching/routing algorithms. SDN reduces the network OpEx by simplifying operations, optimizing

resource usage through centralized data/algorithms, and simplifying network software upgrades. SDN also significantly cuts down a network operator's CapEx, since a commercial-off-the-shelf (COTS) server with a high-end CPU is much cheaper than a high-end router [8]. Further, SDN offers the possibilities of dynamic network topologies and network virtualization, which makes it currently a highly popular paradigm [32].

3.2 Reference Architecture

Figure 1 illustrates an automated connection setup in a typical multi-vendor, multi-technology network island setting. Two different network islands are shown. The network island A consists of six different IP routers (C1–C6) from vendor C (e.g. Cisco), whereas, the network island B consists of six IP routers (J1–J6) from Vendor J (e.g. Juniper).

The choice of technology for network island A is PCE-only. A Path Computation Element (PCE-A) is used within the network to compute constrained-based paths across intra- and inter-network island scenarios. The topology discovery and distribution is handled via separate protocols, such as OSPF, and the RSVP-TE protocol can be used for path setup. All protocols need to be installed and configured separately on every router, with only limited possibilities for functionality extensions and optimizations. Network island A has the possibility to migrate to SDN in future. The migration to SDN would benefit network island A by introducing a central intelligence that is capable of automating processes, thus saving OpEx.

The choice of technology for network island B is PCE + SDN. In this network, a central intelligence (SDN Controller B) is directly accessing every router in the network, via a SDN router interface for flexible configuration of router equipment. The SDN Controller B can choose from different network functionalities, such as topology discovery, topology distribution, path computation and path setup. All functionalities are software-defined modules, that are programmed on top of the SDN Controller for on-the-fly functionality extensions and optimizations. Network
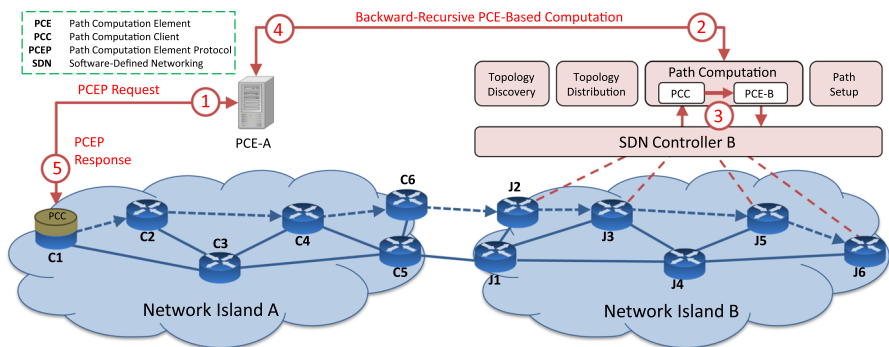
Fig. 1 Example of connection request setup in a multi-vendor network using both PCE and SDN

Island *B* already has the maximal technology set of our case-study. All operations can be fully automated, thus no manual intervention is necessary, resulting in significant OpEx savings.

Both network island are connected via two inter-network island connections. A path computation request from C1 to J6 is handled via the PCEP protocol supported by both network islands. Router C1 sends a PCEP Request message to PCE-A (1). PCE-A tries to compute an end-to-end path to J6, but does not have enough information to calculate this path. PCE-A knows the existence of PCE-B (either through pre-configuration or discovery), and issues a Backward-Recursive PCE-Based Computation. The PCE-B computes the shortest path from J2 to J6 by accessing the SDN Controller B, that is retrieving all necessary information from the Topology Discovery and Distribution for optimal path computation within network island *B*. The optimal path from the entry-router (J2) to the destination (J6) is returned to PCE-A (4). PCE-A now has the optimal path from J2 to J6 and computes the best path from C1 to J2 and returns the whole path to C1 (5). The resulting path (C1–C2–C4–C6–J2–J3–J5–J6) is used to reach the destination.

A couple of comments are worth noting. First, although each PCE sees only its own network topology, BRPC enables an optimized (i.e., best QoS) end-to-end path. Second, despite the fact that each SDN controller can implement its own path computation algorithm, the assumption here is that they often tend to be highly proprietary in nature. Thus, lack of standards makes it hard for SDNs to interoperate in a multi-vendor setting—that is where the IETF-standardized approach with PCE comes in as an effective solution for interoperability.

3.3 Interplay Involved in Joint Migration to PCE and SDN

As can be seen, the interplay involved in joint migration to PCE and SDN can lead to interesting, non-trivial network behavior, which we now discuss in further detail.

In our analysis, we assume a typical control plane with management network control environment. Although the PCE's design addresses mainly path computation, it was previously used in networking settings in conjunction with SNMP for path setup [33]. SDN brings more programmability in path setup, but this is not its only difference to PCE (later on we mention others, such as path provisioning, topology discovery and topology distribution). A network operator has an advantage in migrating to SDN over PCE, as a PCE can only compute paths, while a SDN controller can as well provision the computed paths in a highly programmable fashion. However, as previously mentioned, in a typical multi-vendor setting, a PCE has advantages over SDN. This is because PCE (being standardized) can communicate with neighboring PCEs, whereas, SDNs (being non-standardized) cannot. Thus, larger the diversity of network equipment in the same network, greater is the incentive for the network operator to migrate to PCE than SDN, on account of interoperability considerations.

Within a network island, a SDN controller is *likely* to be able to provision a path, even when a PCE may not. A typical SDN controller, based on OpenFlow, is in fact expected to access and configure network elements at the operator's liking, including the handling of lower layers of the network, such as optical circuits. Not only can a SDN controller find paths that a PCE is requesting, but it can potentially

even reconfigure the whole network such that a totally *new* path is configured to provision a connection request. Thus, SDN can potentially create paths with a better QoS unlike PCE, which only computes paths based on requests. Hence, the end-user benefits more if its network provider migrates to SDN, than PCE. On the other hand, as the PCE protocol is reactive in nature, unlike SDN (which is proactive), end-users stand to gain more from PCE than from SDN.

Whereas a SDN controller is triggered by the NMS/OSS in the network, PCE can be triggered by the end-user. Both SDN and PCE benefit the network operator through OpEx reduction; whereas, PCE, in addition, benefits the end-user by providing improved QoS for end-to-end connections involving multiple vendors. Although a network does not attract any additional traffic by migrating to PCE/ SDN, it benefits significantly by reducing its OpEx after migration.

As SDN offers more functionalities than PCE (such as path provisioning, topology discovery and topology distribution), both the CapEx required to migrate to SDN and the resulting OpEx is more than that required to migrate to PCE. In addition, unlike PCE, the non-standardized nature of SDN adds to its OpEx. Further, the CapEx involved in simultaneous migration of a network island to PCE and SDN is less than the sum of the CapEx involved in separate migrations to PCE and SDN. This is because, in case of simultaneous migrations, the PCE can be incorporated *within* the SDN controller, thus providing an integrated platform at a reduced cost.

In summary, network islands that migrate to PCE can compute optimal paths (i.e., with QoS), which can be provisioned using automated network management frameworks, such as SDN. Thus, it is clear that SDN controllers, with its reach limited to a network island, ideally complement the PCEs that can communicate across networks, thereby, enabling optimal end-to-end, multi-vendor, multi-domain path computation and provisioning under QoS constraints.

4 Multi-Technology Network Migration Model

In this Section, we present our generic agent-based model for studying network migration to complementary technologies. As a case study, we apply our model to study the dynamics of joint migration to multi-vendor path computation and provisioning, namely PCE and SDN, respectively.

4.1 Generic Model

Our model captures the collaborative and competitive business relationships between the agents and also the inter-dependencies involved in their decision-making process. The time is discretized, and thus the model progresses in time-steps. The agents are considered to be *myopic* (in time) in their decision-making and are assumed to act under *complete information*. The former assumption entails each agent optimizing their strategy choices *locally* (in time), while the latter means that each agent is aware of the complete network topology as well as the past strategy choices of all other agents.

Notations: The agents in our model are denoted by $N_1, N_2, \ldots, N_i, \ldots$. An agent's strategy set is represented by a compatible combination of the available strategies.

We denote this universal set of strategies available for the agents to choose from, by two sets of *substitutive* strategies, $S = \{S_u, S_v\}$, where u and v are the *complementary* technologies under consideration, which implies that the payoff that an agent derives by adopting both of them simultaneously is higher than the sum of its payoffs derived by adopting each of them separately (while, no such relationship is assumed to exist between $s_{u,0}$ and $s_{v,0}$). Here, $S_u = \{s_{u,0}, s_{u,1}\}$ represents the strategy of non-adoption and adoption of technology u, respectively. Similarly, $S_v = \{s_{v,0}, s_{v,1}\}$ represents the strategy of non-adoption and adoption of technology v, respectively. Further, $s_{u,0}$ (or $s_{v,0}$) and $s_{u,1}$ (or $s_{v,1}$) are *substitutive* strategies, as an agent can adopt only one of them at any given time. Thus, an agent's *strategy set* for any given time-step is denoted by $a = \{s_{u,k_1}, s_{v,k_2}\}$, where, $k_1, k_2 \in \{0, 1\}$. The volume of sales of agent N_i given its strategy set a is denoted by T_a^i.

An agent's revenue and OpEx depends on its amount of sales, while the cost of changing its strategy set depends on the required CapEx. Considering this, we define the following notations.

$$C_i(a \to a') \quad \triangleq \quad \text{CapEx of } N_i \text{ to from } a \text{ to } a'$$
$$R_i(a) \quad \triangleq \quad \text{Revenue of } N_i \text{ with strategy set } a$$
$$O_i(a) \quad \triangleq \quad \text{OpEx of } N_i \text{ with strategy set } a$$

where, a denotes the current strategy set of agent N_i and a' denotes the strategy set to which N_i migrates in the subsequent time-step. We define the payoff of an agent on migrating to a different strategy set by the *return on investment* it derives from such a decision. The payoff derived by an agent on migrating from a to a' is thus given by the CapEx involved in the migration and the corresponding change in revenue and OpEx as:

$$
\begin{aligned}
P_i(a \to a') &= \frac{\Delta(\text{Revenue}) - [\text{CapEx} + \Delta(\text{OpEx})]}{\text{CapEx}} \\
&= \frac{[R_i(a') - R_i(a)] - C_i(a \to a') - [O_i(a') - O_i(a)]}{C_i(a \to a')}
\end{aligned}
\tag{1}
$$

Each agent thus optimizes its strategy choices at every time-step based on its payoff maximization in the immediate future. Note that each of the CapEx, OpEx and revenue functions, in turn depend on the amount of sales of agent N_i, namely, T_a^i and $T_{a'}^i$. T_a^i, viz. the current amount of sales of agent N_i, can be deterministically computed by N_i from its system measurements, whereas, $T_{a'}^i$, viz. the expected amount of sales of N_i on transitioning from strategy set a to a', is unknown. We next present two different approaches to estimate this expected amount of sales, $T_{a'}^i$.

4.2 Estimation of $T_{a'}^i$

The amount of sales of an agent primarily depends on the agent's technology choices, which in turn is significantly affected by the strategy choices of the neighboring agents within its '*circle of influence*' [34, 35]. We define this novel

concept referred to as a *circle of influence* of an agent as its neighborhood comprising of all agents, whose technology choices *significantly* affects the migration decision of the agent under consideration. Thus, the circle of influence of, say, agent N_i comprises of all agents whose distance from agent N_i is bounded by a threshold distance (by the shortest path), say δ_i. We call δ_i as the '*relevant radius*'[2] of N_i's circle of influence. We also note that the mutual effect of the strategy choices of two agents (within each others circle of influence) is inversely proportional to the distance between them. To capture this aspect, we define the *effective migration coefficient* of agent N_i, as the weighted average of the strategy sets of all agents within N_i's circle of influence; the weights being the reciprocal of the distance of the corresponding agent from N_i. The influence of the strategy choices of an agent, which does not fall within N_i's circle of influence, on N_i's migration decision is, hence, considered negligible. Thus, for an agent to estimate its expected amount of sales in the immediate future, it needs to estimate of the strategy choices of all agents within its circle of influence, in the immediate future. This computation of effective migration coefficient for agent N_i is further illustrated in Algorithm 1.

Algorithm 1 Effective migration coefficient of agent N_i

$num \leftarrow 0$
$den \leftarrow 0$
for all agents N_j **do**
    if $i \neq j$ **then**
        $dist \leftarrow$ minumum number of hops between N_i and N_j
        if $dist < \delta_i$ **then**
            $num \leftarrow num + \dfrac{\text{Migration state of } N_j}{dist}$
            $den \leftarrow den + \dfrac{1}{dist}$
        end if
    end if
end for
effective migration coefficient of $N_i \leftarrow \dfrac{num}{den}$

Figure 2 shows a 12-node network to illustrate the above mentioned concepts. In this topology, the relevant radius of agent N_1, i.e. δ_1, is considered to be 2 hops, and N_1's circle of influence is marked by a dotted line. The adjoining tables in Fig. 2 list the current migration state of all agents in the network. Given this, the effective migration coefficient of N_1 is thus given by,

$$\frac{\overbrace{\frac{1}{1}}^{N_2} + \overbrace{\frac{0}{2}}^{N_3} + \overbrace{\frac{0}{1}}^{N_4} + \overbrace{\frac{1}{2}}^{N_5} + \overbrace{\frac{0}{1}}^{N_6} + \overbrace{\frac{0}{2}}^{N_7} + \overbrace{\frac{1}{2}}^{N_{10}}}{\frac{1}{1} + \frac{1}{2} + \frac{1}{1} + \frac{1}{2} + \frac{1}{1} + \frac{1}{2} + \frac{1}{2}} = \frac{2}{5} = 0.4$$

---

[2] Although in our simulations (in Sect. 5), we consider the relevant radius to be five hops, it would be interesting to investigate the effect of this parameter on the migration profile for real-world scenarios. This effect can probably be game-theoretically modeled, however, this is in particular is not our focus here, and hence beyond the scope of this paper.

We next present two heuristics for an agent to estimate its neighbor's strategy in the subsequent time-slot, based on *probabilistic* and *deterministic* methods. The underlying rationale behind both these heuristics is that an agent's strategy choice is very likely to vary with that of the majority of the agents in its circle of influence [36, 37].

1. *Deterministic Strategy Estimation*: In the *deterministic approach*, an agent considers the strategy choices of its neighboring agents to be the same as that of the majority of the agents in their circle of influence. Thus, while agent N_i is estimating its future amount of sales, if N_j is within N_i's circle of influence, and if more than 50 % of the agents in N_j's circle of influence employ strategy set a in the current time-step, then N_i expects N_j to switch to strategy set a in the next time-step, under this approach.

2. *Probabilistic Strategy Estimation*: In this estimation approach, an agent considers the *probability* of its neighbor's strategy choice in the subsequent time-slot to be, a, as x, if x denotes the fraction of agents with strategy set a, in this neighbor's circle of influence, in the current time-slot. Thus, in the process of agent N_i estimating its future amount of sales, if N_j is within N_i's circle of influence, and if, say, 30 % of the agents in N_j's circle of influence employ strategy set a in the current time-step, then N_i assumes the probability of N_j switching its strategy set to a in the subsequent time-step as 0.3, under this approach.

Note that it is due to our assumption of complete information that these heuristics can be realized. Figure 3 plots the probability of migration of an agent using deterministic and probabilistic estimation approaches, as a function of its effective migration coefficient.

An agent thus estimates the strategy sets of all agents within its circle of influence in the immediate future, using one of the two strategy estimation approaches, mentioned above. It thus disregards the future strategy choices of agents outside its circle of influence, and assumes them to maintain the same strategy set, in the subsequent time-step. Thereafter, the agent takes note of its own set of possible transitions from its current state, i.e., $a \rightarrow \{a_1, a_2, \ldots\}$ (see Fig. 4). It then computes the payoffs resulting from each of its possible transitions, in sync with the strategy set estimations of the agents within its circle of influence, i.e., $\{P_i(a \rightarrow a_j)\}, \forall j$, and accordingly chooses its future strategy set as the one that maximizes its resulting payoff, i.e., $a' = \arg\max_{a_j}\{P_i(a \rightarrow a_j)\}$, given its current strategy set a. In this way, an agent optimizes its strategy set at each time-step.

### 4.3 Agent-Based Model Applied

In this subsection, we customize our generic network migration to the particular scenario of migration to PCE and SDN.

Table 1 summarizes the mappings between the generic network migration model and PCE/SDN scenario. In the context of PCE/SDN, agents translate to network islands, strategies correspond to technology choices, amount of sales relate to the

N_1	1	N_7	0
N_2	1	N_8	1
N_3	0	N_9	0
N_4	0	N_{10}	1
N_5	1	N_{11}	0
N_6	0	N_{12}	1
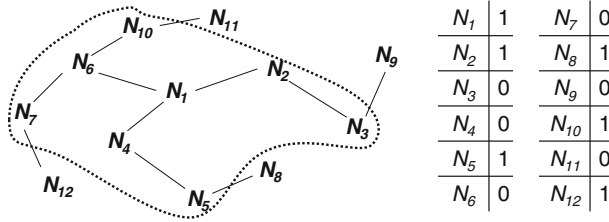
Fig. 2 Circle of influence

Fig. 3 Probability of Migration of an agent in deterministic and probabilistic estimation approaches, as a function of its effective migration coefficient
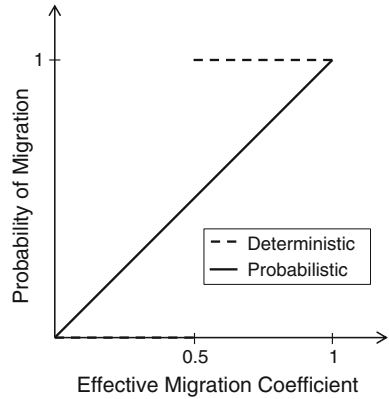

Table 1 Mapping generic migration model to PCE/SDN

Agent	$\longleftrightarrow$	Network Island
Strategy	$\longleftrightarrow$	Technology Choice
Amount of Sales	$\longleftrightarrow$	Amount of Traffic
Technology u	$\longleftrightarrow$	PCE
Technology v	$\longleftrightarrow$	SDN

amount of traffic that a network transits for its customers, technology u maps to PCE, while, technology v maps to SDN.

Figure 4 shows all possible strategy set transitions for a network island, under the assumption that an island that has once migrated to $s_{PCE,1}$ or $s_{SDN,1}$ does not revert back to $s_{PCE,0}$ or $s_{SDN,0}$, respectively, in the future. This assumption is justified because the functionalities provided by PCE and SDN are beneficial to a network, irrespective of external factors, such as the technology choices of other network islands, etc. For instance, a migrated node definitely saves its OpEx, even if the resulting traffic does not increase post-migration (see Fig. 8).

A network island incurs CapEx if it migrates to PCE or SDN. Secondly, the CapEx of a network island is expected to follow economies of scale, i.e., cost of provisioning per unit of traffic decreases with increasing scale. Thus, the second derivative of the CapEx function with respect to traffic should be less than zero. Although, there are many functions that satisfy this condition, both intuitively and
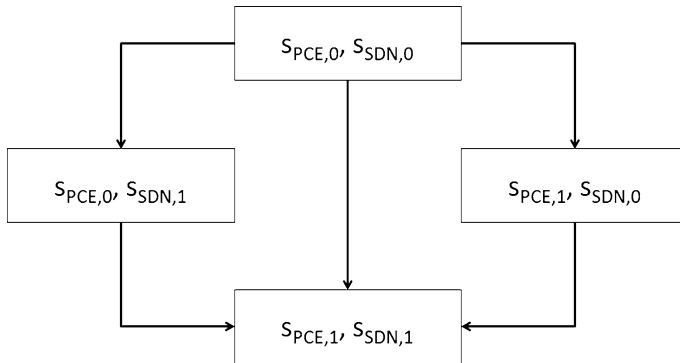
Fig. 4 Strategy set transitions in a network

arbitrarily, we assume the CapEx to vary with the square root of the network traffic. In addition, on account of the complementary relationship between PCE and SDN, the CapEx incurred by a network island in migrating to both the technologies simultaneously is less than the sum of the CapEx incurred by migrating to each of them separately. This is because, although PCE and SDN are separate components, if a network island migrates to both of them simultaneously, it can integrate both the technologies into a single, integrated component, leading to a reduced CapEx, as compared to a PCE component, and a separate SDN component. Considering both these aspects, the CapEx of network island N_i from the generic model in Eq. (1) can be expressed, in this case, as

$$C_i(a \rightarrow a') = c_i(a, a')\sqrt{T_{a'}^i} \tag{2}$$

where, $c_i(a, a') \in [0, 1]$ is a coefficient given by,

$$c_i(a, a') = \begin{cases} c_{\text{PCE}} & \{s_{\text{PCE},0}, s_{\text{SDN,k}}\} \rightarrow \{s_{\text{PCE},1}, s_{\text{SDN,k}}\} \\ c_{\text{SDN}} & \{s_{\text{PCE,k}}, s_{\text{SDN},0}\} \rightarrow \{s_{\text{PCE,k}}, s_{\text{SDN},1}\} \\ \left(\dfrac{c_{\text{PCE}} + c_{\text{SDN}}}{\eta}\right) & \{s_{\text{PCE},0}, s_{\text{SDN},0}\} \rightarrow \{s_{\text{PCE},1}, s_{\text{SDN},1}\} \end{cases} \tag{3}$$

where, $k \in \{0, 1\}, c_{\text{PCE}}, c_{\text{SDN}} \in [0, 1]$ and $\eta \in [1, 2]$ denotes the *coupling coefficient*—$\eta = 1$ implies fully independent technologies, such that, migrating to both these technologies simultaneously is equivalent to migrating to each of them separately, whereas, $\eta = 2$ implies fully substitutive technologies, such that, migrating to both of them simultaneously is equivalent to migrating any one of them. In the context of PCE and SDN, we consider $\eta = 1.5$ in this paper.

The revenue of a network island primarily depends on the amount of traffic flowing through it, and does not vary with the set of technologies deployed by the network operator. This is because the revenue comes from the customer, who is oblivious to the technology adopted by its network operator. The customer, generally, pays to the network operator, solely based on the amount of traffic that the operator transits for it. In addition, revenue of a network island is expected to

follow economies of scale. We, thus, consider the revenue of a network island to vary as the square of the network traffic. And, given the *qualitative* nature of our model, without loss of generality, we set,

$$R_i(a) = (T_a^i)^2 \tag{4}$$

Similar to CapEx, the OpEx of PCE and SDN in a network island is expected to follow economies of scale, i.e., every subsequent unit of traffic incurs a lesser CapEx than the previous. Hence, we consider the OpEx of a network island to vary with the square root of the network traffic. Thus,

$$O_i(a) = \alpha_i(a)\sqrt{T_a^i} \tag{5}$$

where, $\alpha_i(a)$ is a coefficient given by,

$$\alpha_i(a) = \begin{cases} \alpha_{\overline{\text{PCE}}} + \alpha_{\overline{\text{SDN}}} & a = \{s_{\text{PCE},0}, s_{\text{SDN},0}\} \\ \alpha_{\overline{\text{PCE}}} + \alpha_{\text{SDN}} & a = \{s_{\text{PCE},0}, s_{\text{SDN},1}\} \\ \alpha_{\text{PCE}} + \alpha_{\overline{\text{SDN}}} & a = \{s_{\text{PCE},1}, s_{\text{SDN},0}\} \\ \left(\dfrac{\alpha_{\text{PCE}} + \alpha_{\text{SDN}}}{\eta}\right) & a = \{s_{\text{PCE},1}, s_{\text{SDN},1}\} \end{cases} \tag{6}$$

where, the overline operator ($\overline{\text{PCE}}$ and $\overline{\text{SDN}}$) denotes the alternatives available (say, manual operations) to the corresponding technology, (i.e., PCE and SDN, respectively) and $\alpha_{\overline{\text{PCE}}}, \alpha_{\overline{\text{SDN}}}, \alpha_{\text{PCE}}, \alpha_{\text{SDN}} \in [0, 1]$. Thus, α_{PCE} is the coefficient of the PCE component of OpEx in the presence of PCE, whereas, $\alpha_{\overline{\text{PCE}}}$ denotes the corresponding coefficient in the absence of PCE. Similarly, for α_{SDN} and $\alpha_{\overline{\text{SDN}}}$. The presence of η in Eq. (6) captures the complementary relationship between PCE and SDN, i.e., the OpEx incurred by a network island on migrating to both the technologies simultaneously is less than the sum of the OpEx incurred by migrating to each of them separately.

We also note that both PCE and SDN are significantly more efficient than their alternative technologies (say, manual operations). Thus, a domain migrating to either PCE or SDN is expected to result in a non-negative change in OpEx, or in other words, in OpEx savings. To put it mathematically, the corresponding OpEx coefficients of PCE and SDN, pre- and post-migration must satisfy the following inequalities.

$$\begin{aligned} \alpha_{\text{PCE}} &< \alpha_{\overline{\text{PCE}}} \\ \alpha_{\text{SDN}} &< \alpha_{\overline{\text{SDN}}} \end{aligned} \tag{7}$$

In all migration scenarios in general, and in migration to PCE or SDN in particular, the major investment is often in the CapEx involved, whereas, the post-migration OpEx decreases, compared to pre-migration OpEx costs. Moreover, the CapEx of migration generally supersedes the post-migration OpEx costs by a significant margin. This, in conjunction with Eqs. (3) and (6), leads us to state,

$$c_{\text{PCE}} > \max\left\{\alpha_{\text{PCE}} + \alpha_{\overline{\text{SDN}}}, \frac{\alpha_{\text{PCE}} + \alpha_{\text{SDN}}}{\eta}\right\} \tag{8}$$

$$c_{\text{SDN}} > \max\left\{\alpha_{\overline{\text{PCE}}} + \alpha_{\text{SDN}}, \frac{\alpha_{\text{PCE}} + \alpha_{\text{SDN}}}{\eta}\right\} \tag{9}$$

$$\frac{c_{\text{PCE}} + c_{\text{SDN}}}{\eta} > \frac{\alpha_{\text{PCE}} + \alpha_{\text{SDN}}}{\eta} \tag{10}$$

Equation (8) results from the fact that the CapEx of migrating from $\{s_{\text{PCE},0}, s_{\text{SDN},k}\}$ to $\{s_{\text{PCE},1}, s_{\text{SDN},k}\}$ is greater than the post-migratin OpEx costs in both cases ($k = 0, 1$). However, since the CapEx and OpEx functions are similar in nature, this relationship also holds for the corresponding coefficients. Thus, the corresponding CapEx coefficient (c_{PCE}) must be greater than both the OpEx coefficients in the two scenarios (viz., $\alpha_{\text{PCE}} + \alpha_{\overline{\text{SDN}}}$ and $\frac{\alpha_{\text{PCE}} + \alpha_{\text{SDN}}}{\eta}$). Equations (9) and (10) result from similar arguments for migrations from $\{s_{\text{PCE},k}, s_{\text{SDN},0}\}$ to $\{s_{\text{PCE},k}, s_{\text{SDN},1}\}$, and from $\{s_{\text{PCE},0}, s_{\text{SDN},0}\}$ to $\{s_{\text{PCE},1}, s_{\text{SDN},1}\}$, respectively.

Eliminating $\alpha_{\overline{\text{SDN}}}$ between Eqs. (7) and (8), we have,

$$c_{\text{PCE}} > \alpha_{\text{PCE}} + \alpha_{\text{SDN}} \tag{11}$$

Similarly, eliminating $\alpha_{\overline{\text{PCE}}}$ between Eqs. (7) and (9), we have,

$$c_{\text{SDN}} > \alpha_{\text{PCE}} + \alpha_{\text{SDN}} \tag{12}$$

With the above definitions of CapEx (Eq. (2)), OpEx (Eq. (5)) and revenue (Eq. (4)), as applicable for the joint migration to PCE and SDN, and subject to the associated contraints amongst the various coefficients (Eqs. (7)–(12)), the payoff function in Eq. (1), reduces to,

$$P_i(a \to a') = \frac{[(T_{a'}^i)^2 - (T_a^i)^2] - [c_i(a, a') + \alpha_i(a')]\sqrt{T_{a'}^i} + \alpha_i(a)\sqrt{T_a^i}}{c_i(a, a')\sqrt{T_{a'}^i}} \tag{13}$$

We now summarize the overall migration decision-making process of a network island. As our model advances in discrete time-steps, at every time-step, each network island chooses its strategy set for the subsequent time-step by computing its payoff (Eq. (13)) resulting from each of its feasible strategy set transitions (see Fig. 4). As evident from Eq. (13), this payoff computation for a network island depends on the traffic it transits in the subsequent time-step i.e. $T_{a'}^i$, which is unknown as it depends on the new network-wide routing configuration resulting from the migration decisions of each network island. Each network island, hence, estimates this traffic as detailed in Sect. 4.2, and thus determines their strategy set for the subsequent time-step. It is to be noted that although the migration decisions of a network island effectively depend on the migration coefficients, it is the payoff

function of a network island that provides a uniform metric to quantify the effect of the PCE/SDN migration coefficients of all neighboring nodes.

5 Numerical Results

In this section, we present our simulation framework and the empirical results to evaluate various aspects of our proposed network migration model.

5.1 Simulation Model

The inter-AS (autonomous system) network topology of the Internet is not a random mesh, but has a well-defined hierarchical tree-like structure, with Tier-1 ISPs at the top, and smaller and smaller ISPs as we go down. Also, the links are not randomly distributed between the nodes, but are more concentrated at the Tier-1 ISPs and decrease with lower and lower ISPs. These realistic characteristics are effectively (and very simply) modeled by the scale-free network and, as a result, we model our inter-AS topology using a scale-free network.

For our simulation, we consider a scale-free network of 100 interconnected network islands, comprising of 39 "*transit*" islands and 61 "*stub*" islands. Akin to the terminology used in global Internetworks, a network island that is not a provider for any other island is called a *stub island*, while all other islands are called as *transit islands* [38]. Stub islands represent the end-users, and hence, the choice of migration rests only with the transit islands. Our topology was generated using Barabási and Albert's topology generation algorithm [39], where the seed network comprised of 16 fully inter-connected network islands, referred to as *seed islands* due to their higher resulting connectivity. In our topology, a node represents a network island and a link represents an inter-island connection. To comply with policy-aware routing, each edge is marked as either Customer-to-Provider (C2P) or Peer-to-Peer (P2P). We employ No-Valley-Prefer-Customer (NVPC) routing to provision connection requests between two network islands, which comprises of the following two rules [40]:

- Paths learned from providers or peers are never advertised to other providers or peers.
- Paths learned from customers are preferred to the paths learned from peers and providers, and paths learned from peers are preferred to the paths learned from providers, regardless of path length.

Our simulation concerns with migration to technologies such as PCE, which are beneficial to a connection request, only when *all* domains on its path from source to destination, have migrated to the technology in question. This reflects in our routing algorithm, such as, while provisioning a connection request, amongst various equi-cost paths, the source domain prefers a path in which all domains have migrated to PCE. And if multiple equi-cost, shortest paths exist, the traffic is uniformly distributed across all such paths, or randomly over one of these paths, depending on

the user preference. We model the incoming connection requests for each source-destination stub domain pairs as Poisson arrivals. The connection requests once provisioned are assumed to stay the same till the end of simulation. Link capacity is assumed to be unlimited, since for a given increment in incoming traffic (which translates to revenue for the host network island), the host network operator can easily increment the link bandwidth, with minimal effort. This is especially true since our study is not based on infinitesimal timescales, but of the order of weeks or months, wherein a domain has the flexibility to increase its link capacity, subject to incoming requests. It is to be noted here that our choice of unlimited link capacities does not contradict our need for multi-path routing. Had the routing depended only on the link capacities, multi-path routing would have been redundant. However, in our model, the routing depends on the link capacities as well as the migration states of the intermediate nodes. Furthermore, to *exclusively* study the inter-relationships between the migration decisions and the network-wide routing, we set the link capacities to be unlimited. All stub-to-stub paths had traffic since the beginning of the simulation.

As a new connection request arrives in a network island, the network provisions the request and reconsiders its migration choices based on its payoff function, as defined in Sect. 4. This, in turn, leads to its neighbors reconsidering their respective migration choices, which thus cascades throughout the network. Finally, on registering a change in the migration decision of any domain in the network, each domain revises the routes of its provisioned connections. All the presented results plot average values across 50 traffic profiles (each Poisson distributed), with each traffic profile replicated 5 times to eliminate any statistical variations. Paths were precomputed and stored, instead of on-the-fly path computations, as it significantly improved the simulation run time. The two primary input preferences to our simulation are (1) *equi-cost routing*—when multiple equi-cost paths exist to provision a given connection request, we consider both possibilities of assigning it to a single random node amongst them (single-path routing), as well as, that of uniformly distributing the traffic over all such paths (multipath routing), and, (2) *strategy estimation approach*—the approach used by domain to estimate the future technology deployment in neighboring domains in the process of optimizing its own migration decision; we consider two approaches for the same, namely, deterministic and probabilistic approaches, as defined in Sect. 4.2.

We next present our simulation results from various experiments studying a variety of factors affecting the network *migration profile*. By '*migration profile*', we mean the progress of the network-wide migration captured by monitoring the number of migrated nodes throughout the simulation. Unless otherwise stated, the parameter values assumed in our simulation are $\eta = 1.5$, $c_{\text{PCE}} = 0.3$, $c_{\text{SDN}} = 0.4$, $\alpha_{\text{PCE}} = 0.1$, $\alpha_{\text{SDN}} = 0.2$, $\alpha_{\overline{\text{PCE}}} = 0.5$ and $\alpha_{\overline{\text{SDN}}} = 0.8$ (though other parameter combinations were also found to result in similar plots). The relevant radius for each domain (as defined in Sect. 4.2) was set to 5 hops. Throughout our case studies, we consistently observe that the number of migrants increase during the simulation, perhaps rapidly in the beginning, and saturate gradually. Further, it is important to note that although the nature of the plots looks similar *across* the case studies, what

is important to note is the difference in the migration profiles subject to variation in parameters *within* a case study. The choices made by network islands in our proposed model play a vital role on the resultant migration profile, be it the choice of strategy estimation approach (deterministic/probabilistic), routing preferences (single/multi-path) or the values of the various coefficients. Equations (7)–(12) capture real-life constraints on these coefficients to reflect the real-world conditions. Similarly, incorporation of more and more such relationships would bring the model closer to real-world conditions. The effects of these choices (deterministic/ probabilistic estimation strategy, single/multi-path routing preferences) on the migration profile are further demonstrated in the following sub-sections.

5.2 Single Versus Double Migration

In this experiment, we study the migration profiles of PCE and SDN, under varied circumstances.

Figure 5 plots the migration profiles of nodes in the network to PCE, SDN and PCE+SDN, under probabilistic strategy estimation approach and multi-path routing preference. Given that we assume migration to SDN is more expensive than that to PCE (i.e., $\alpha_{PCE} < \alpha_{SDN}$), Fig. 5 shows that a greater number of nodes migrate to PCE, than SDN, and also that almost every node that migrates to SDN also migrates to PCE. We observe from Fig. 5 that almost every domain that migrates to SDN, also migrates to PCE. This demonstrates the fact the benefits derived from SDN are best exploited in combination with PCE, than by itself.

Figure 6 plots the migration profiles to PCE and SDN, in three different scenarios, under deterministic strategy estimation approach and multi-path routing
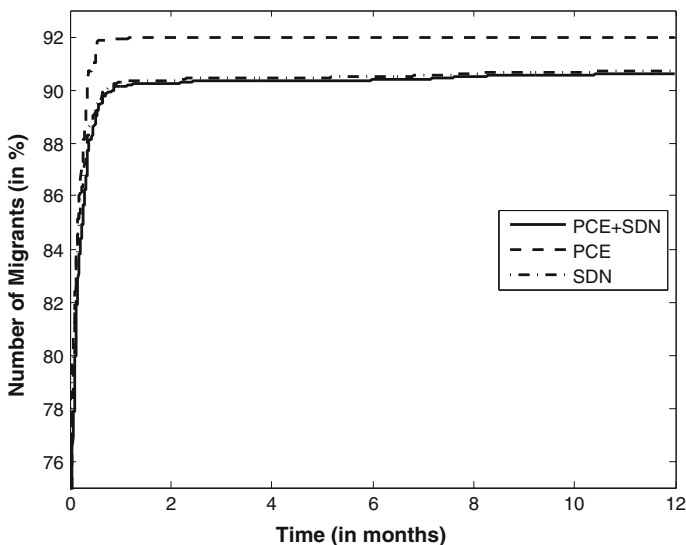
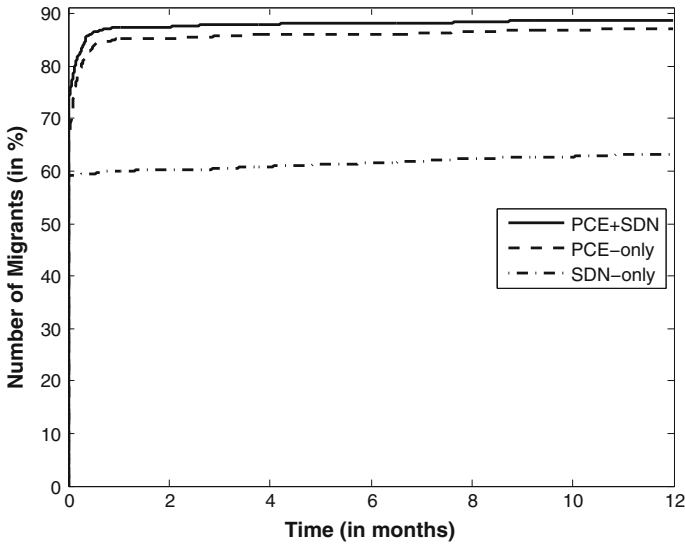**Fig. 5** Migration profiles of PCE, SDN, and PCE+SDN

Fig. 6 Single versus double migrations

preference. *PCE-only* plots the PCE migration profile in the network, when only migration to PCE is studied in isolation, i.e., SDN is not considered at all. Similarly, *SDN-only* plots the SDN migration profile in the network, when only migration to SDN is considered in isolation, i.e., PCE is not studied at all. Finally, *PCE+SDN* plots the profile of nodes migrating to *both* PCE and SDN, when PCE and SDN migrations are considered simultaneously. This plot shows that migration to SDN which is generally small by itself, can be further promoted by joint migration to PCE, which is more widely accepted, given the complementary relationship between PCE and SDN. Also, a small increase can be observed in the PCE migration from *PCE-only* to *PCE+SDN*, thus SDN also has a small impact in improving the PCE deployment.

5.3 Early Adopters

We next study the effect of early adopters on the PCE and SDN migration profiles in the network, based on the type and number of early adopters. In this experiment, an early adopter is a network domain that has migrated to PCE since the beginning of simulation. Early adopters act as the seed for migration in the network, thereby catalyzing the migration process.

Figure 7 (top) plots the effect of type of PCE early adopters on the PCE migration profile in the network, under deterministic strategy estimation approach and multi-path routing preference. We choose the early adopters based on their degree of connectivity in the network. Figure 7 (top) contrasts the PCE migration profile in the network given no early adopters, 3 early adopters (amongst the minimum degree nodes in the network), and 3 early adopters (amongst the maximum degree nodes in the network). As can be intuitively expected, these plots suggest that nodes with
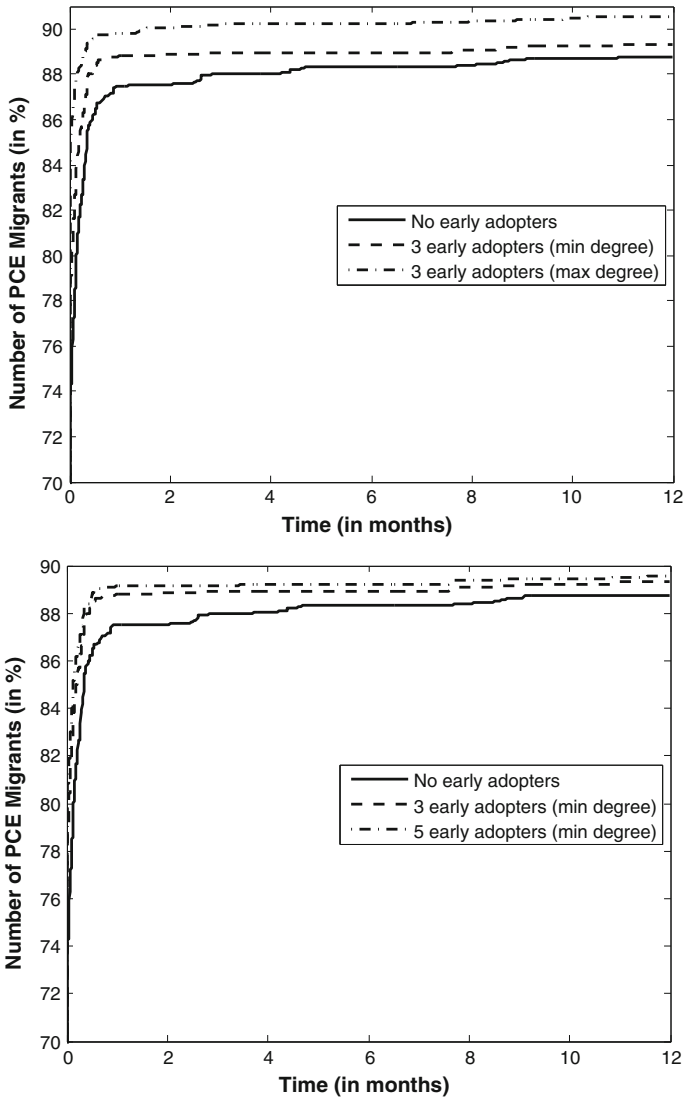
Fig. 7 Effect of early adopters on PCE migration profile by type (*top*) and number (*bottom*)

high degrees, on migrating, have a greater effect in promoting the network-wide migration profile, than nodes with smaller degrees. This can be attributed to the fact that a large number of paths pass through the high-degree nodes in the network. Thus, the migration of a single high-degree node would affect the migration choices of a large number of transit nodes, due to its high degree of connectivity.

Figure 7 (bottom) plots the effect of number of PCE early adopters on the PCE migration profile in the network, under deterministic strategy estimation approach and multi-path routing preference. It contrasts the PCE migration profile in the network given 0, 3 and 5 early adopters (amongst the minimum degree nodes in the

network). As can be intuitively expected, the plot shows that a higher the number of early adopters result in a better migration profile.

5.4 Cause of Migration

In this experiment, we study the motivations for transit domains to migrate to either PCE or SDN or both. As discussed earlier, a transit node migrates either to reduce its OpEx, or to increase the traffic flowing through it (and, in turn its revenue), or both. For every domain that choose to migrate during the simulation, we monitored them, and categorized their cause of its migration, amongst (1) exclusive reduction in OpEx, (2) exclusive increase in traffic (in turn, resulting in an increase in its revenue), and (3) both (1) and (2). Figure 8 plots this data (in percentages) for various combinations of routing choice (single- or multi-path) and strategy estimation choice (deterministic or probabilistic). This plot contradicts the common misnomer that a domain migrates primarily because of a resulting increase in traffic (or revenue). The plot illustrates an important aspect of migration, which is, a transit node may migrate even if its migration decision does not result in an increase in traffic (or revenue), but only based on its OpEx reduction. We observe that a significant fraction of migrations result exclusively due to decrease in OpEx. Moreover, OpEx reduction proves to be more important in case of single path routing, than multi-path routing. Figure 8 also demonstrates that revenue increase almost always results in combination with OpEx reduction as a cause of migration, and rarely in isolation.

5.5 Effect of Coupling Coefficient

In this experiment, we study the effect of coupling coefficient on the migration profile. Figure 9 plots the effect of coupling coefficient on PCE (top) and SDN (bottom) migration profiles in a 150-node topology with 92 stubs and 58 transits, under deterministic strategy estimation approach and multi-path routing preference. We observe that the resulting migration profile is enhanced, when we account for
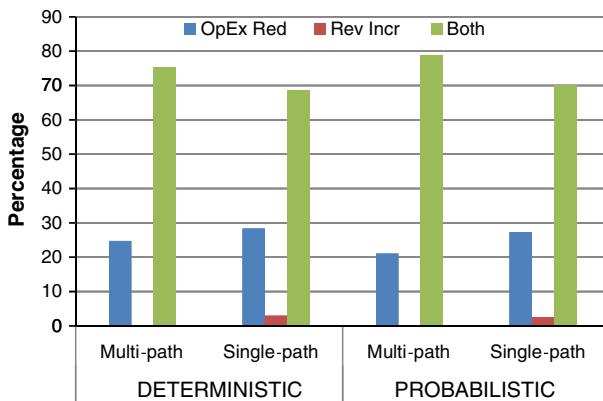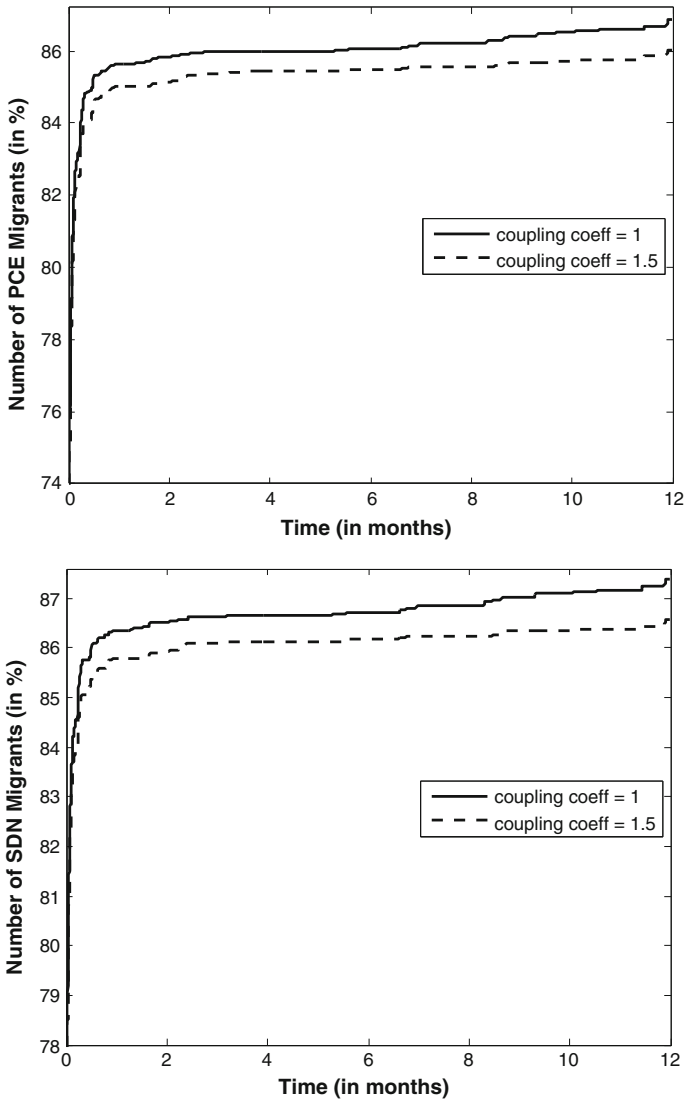

Fig. 8 Cause of migration

Fig. 9 Effect of coupling coefficient on PCE (*top*) and SDN (*bottom*) migration profiles

the complementary relationship between PCE and SDN (coupling coefficient = 1.5) than otherwise (coupling coefficient = 1). This is because, when PCE and SDN operate simultaneously in a domain, the resulting benefits are larger than the sum of benefits derived from PCE and SDN individually. Thus, domains deploying either PCE or SDN benefit from this aspect, and also choose to adopt the complementary technology i.e., SDN or PCE, respectively, consequently resulting in a higher number of migrants.
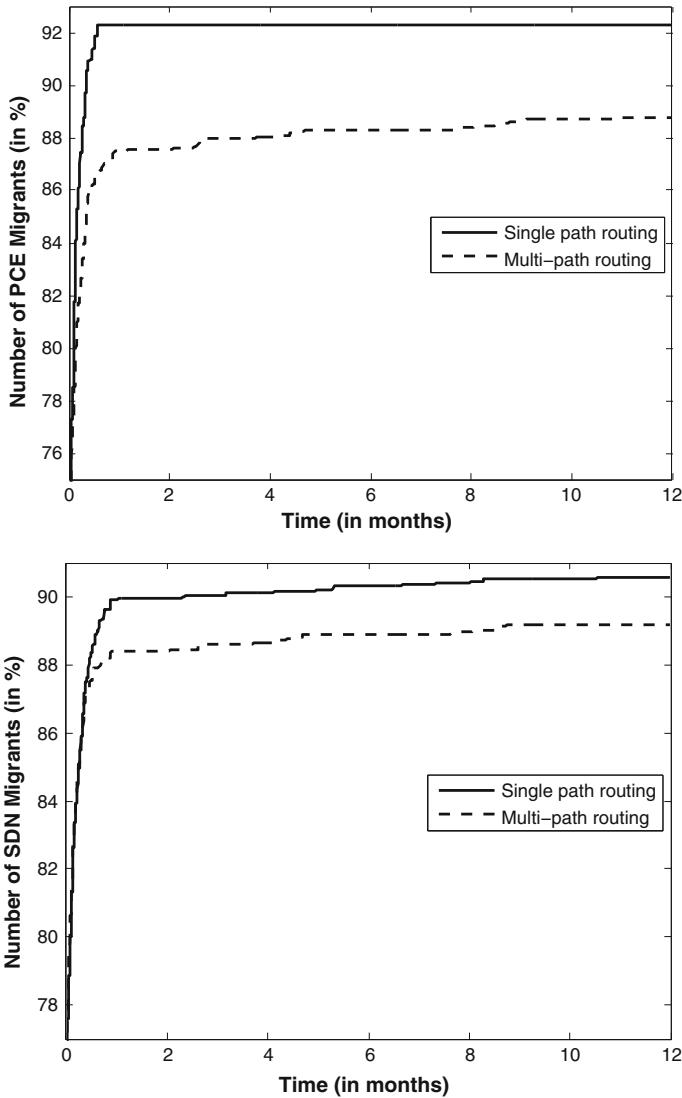
Fig. 10 Effect of routing choices on PCE (*top*) and SDN (*bottom*) migration profiles

5.6 Effect of Equi-Cost Routing Preferences

In this experiment, we study the effect of routing choices, when multiple equi-cost shortest paths exist in the network to provision a user request. Figure 10 plots the effect of equi-cost routing preferences on the PCE (top) and SDN (bottom) migration profile, under deterministic strategy estimation approach and multi-path routing preference. In presence of multiple equi-cost shortest paths, we consider the routing choices of randomly choosing any one of them (single-path routing), or

distributing traffic uniformly across all of them (multi-path routing). As can be observed from Fig. 10, the former choice results in faster migration than the latter. This may be attributed to the fact that distributing traffic over multiple paths reduces the amount of traffic flowing through each such path, thereby lessening the incentive derived by the intermediate transit nodes from migration.
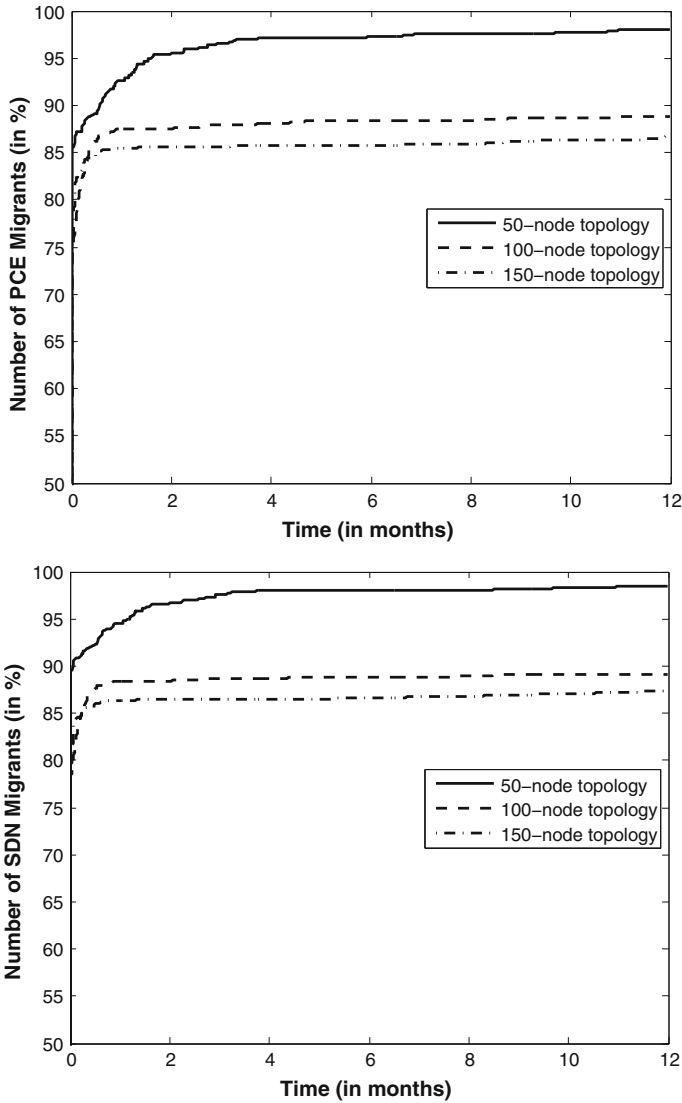
5.7 Effect of Network Topology

In this section, we discuss the effect of size of the network topology on the migration profile of a network. In addition to the 100-node topology, we consider 50- and 150-node topologies, with similar characteristics, in terms of the fraction of stub/transit nodes in the network, seed network size, degree of stub nodes, etc. Figure 11 plots the percentage of nodes migrating to PCE (top) and SDN (bottom) migration profiles, under deterministic strategy estimation approach and multi-path routing preference. We observe that a larger fraction of nodes migrate in the 50-node topology, than in the 100-node topology, which in turn has a larger number of migrants than the 150-node topology. This leads us to conclude that for the same set of parameters, the migration profile is increasingly pronounced in smaller topologies than larger topologies, as the path length increases with the network size.

5.8 Strategy Estimation Approach

In this experiment, we compare the effect of different strategy estimation heuristics employed by a domain on the migration profile of the network. Figure 12 plots the number of domains migrating to PCE or SDN over time, when multipath routing is enabled. We observe that the deterministic approach results in a lesser number of migrants than the probabilistic approach for both PCE and SDN migrations.

This behavior can be explained as follows. As a thumb rule, greater the number of neighboring migrated domains, greater is the likelihood of a domain to migrate. In the deterministic and probabilistic approaches, the estimated number of neighboring migrated domains considered by a domain is *greater than or equal* to the actual number of migrated domains in the neighborhood. Amongst the deterministic and probabilistic approaches, the likelihood of migration of a node varies with the effective migration coefficient of neighboring nodes as shown in Fig. 3. The reader may note that the area under curves in Fig. 3 are proportional to the total number of migrations resulting from each estimation approach. Had the effective migration coefficient of the nodes be uniformly varying between 0 and 1, both approaches would have resulted in similar migration profile. However, we observe in our simulation (and can also be intuitively derived) that the effective migration coefficient varies roughly between 0 and 0.8, thereby providing the probabilistic approach an upper hand. As a result, the probabilistic approach results in a greater number of migrants than that from the deterministic approach.

Although more and more transit domains migrate with increasing traffic in the network, it is important to note that the saturation point of migration is reached not when *all* transit domains migrate, but at a *lesser* number of migrants. For example,

Fig. 11 Effect of topology size on PCE (*top*) and SDN (*bottom*) migration profiles

out of 39 transit nodes in the network, only about 36 migrate at saturation, as seen in Fig. 12. This is because of the shortest-path routing between the stub nodes. Thus, only those transit domains which *lie* on the shortest path(s) between a pair of stub nodes, eventually migrate, whereas, transit nodes with no stub-to-stub traffic have no incentive in migrating, even when every other node in its neighborhood may have migrated.
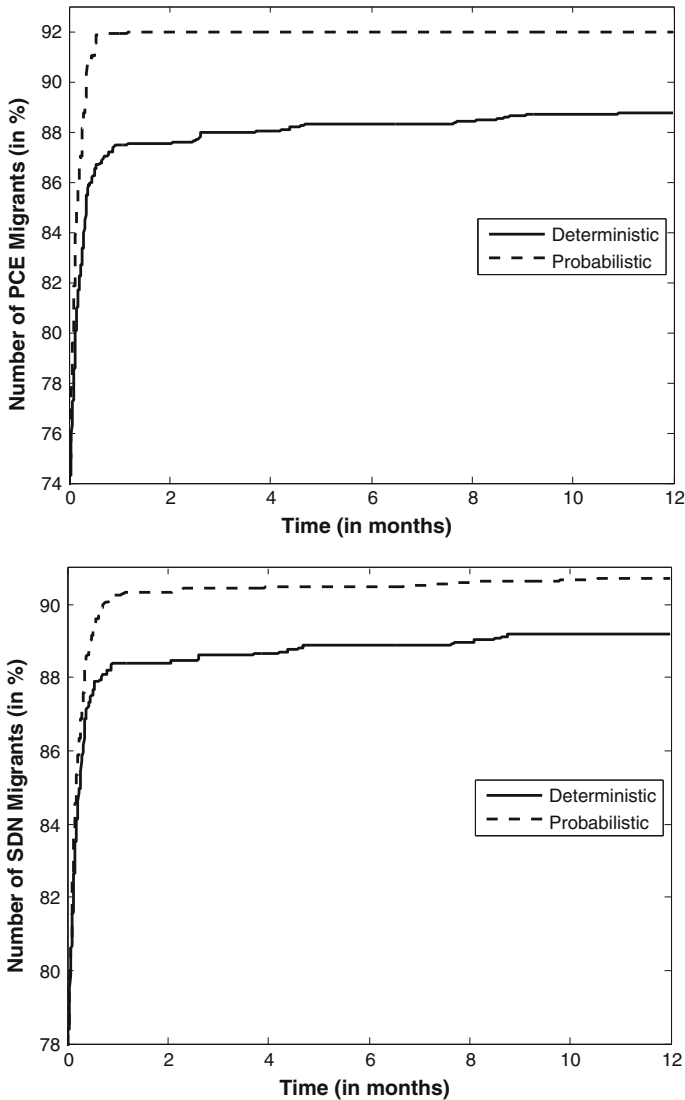
Fig. 12 Effect of the strategy estimation approaches on PCE (*top*) and SDN (*bottom*) migration profiles

6 Conclusion

In this paper, we proposed an agent-based model to study network migration to multiple technologies that may be correlated, and applied it to study two emerging technology frameworks, i.e., PCE and SDN. We believe to have advanced the science in the existing agent-based models by considering a few novel critical factors, including (1) synergistic relationships across multiple technologies, (2) reduction in OpEx as a reason to migrate, and, (3) implications of local network

effects on migration decisions. As is characteristic of agent-based models, defining the mutual, microscopic interactions between agents lead to insights about the macroscopic, system-wide behavior, which was analyzed and demonstrated by our model.

The results obtained from our case study suggest that migration to SDN can be eased by joint migration to PCE, and that the benefits derived from SDN are best exploited in combination with PCE, than by itself. The case study also showed that studying migration to related technologies in combination is important than studying migration to each technology in isolation. The results indicate that the migration to SDN can be promoted by several factors, namely, (a) in combination with a widely-accepted complementary technology such as PCE, (b) early adopters, (c) an agent's ability to predict its neighbor's decisions to migrate to either of the technologies.

Our future work includes applying our model to study larger topologies (of the scale of thousands of domains). Also, multi-vendor, multi-layer network migration scenarios with IP/Optical network integration is a relevant scenario to investigate. Our model can also be extended to study inter-relationships between three or more migrating technologies, which can be explored should a relevant case study emerge. Another important aspect would be to study the order of migration in a network, i.e., "migration scheduling", showing which type of nodes should migrate first.

References

1. Deering, S., Hinden, R.: RFC 2460: Internet Protocol, Version 6 (IPv6) specification, (1998)
2. Iekel-Johnson, S.: World IPv6 launch : A longer view. http://ddos.arbornetworks.com/2012/06/ipv6-launch-day-a-longer-view/
3. Internet2 Headroom Practice. https://wiki.internet2.edu/confluence/download/attachments/17383/Internet2+Headroom+Practice+8-14-08.pdf
4. Leibenstein, H.: Bandwagon, snob, and veblen effects in the theory of consumers' demand. Quart. J. Econ. **64**(2), 183–207 (1950)
5. Economides, N., Viard, B.: Pricing of complementary goods and network effects. In: Regulation and the Economic Performance of Communication and Information Networks. Edward Elgar, (2011)
6. Farrel, A., Vasseur, J.-P., Ash, J.: A Path Computation Element (PCE)-Based Architecture, RFC 4655 (Informational), Internet Engineering Task Force (2006)
7. ONF White paper. (2012, April 13) Software-defined networking: The new norm for networks. https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf
8. Metaswitch. (2012, February) PCE - An Evolutionary Approach To SDN. http://pcetosdn.com/downloads/PCE-Evolutionary-SDN-feb2012.pdf
9. L. Online. (2013, March 11) Researchers integrate openflow with path computation to improve network navigation. http://www.lightwaveonline.com/articles/2013/03/researchers-integrate-openflow-with-path-computation-to-improve-.html
10. Beal, G.M., Bohlen, J.M.: The diffusion process. Agricultural Experiment Station. Iowa State College, Ames (1957)
11. Peres, R., Muller, E., Mahajan, V.: Innovation diffusion and new product growth models: A critical review and research directions. Int. J. Res. Market. **27**(2), 91–106 (2010)
12. Eveland, J.: Diffusion, technology transfer, and implementation thinking and talking about change. Sci. Commun. **8**(2), 303–322 (1986)

13. Rogers, E.M.: Diffusion of Innovations. Simon and Schuster, New York (1962)
14. Bass, F. M.: A new product growth for model consumer durables. Manag. Sci. **15** (5), 215–227 (1969). http://pubsonline.informs.org/doi/abs/10.1287/mnsc.15.5.215
15. Norton, J.A., Bass, F.M.: A diffusion theory model of adoption and substitution for successive generations of high-technology products. Manag. Sci. **33**(9), 1069–1086 (1987)
16. Bass, F.M., Krishnan, T.V., Jain, D.C.: Why the bass model fits without decision variables. Market. Sci. **13**(3), 203–223 (1994)
17. Jin, Y., Sen, S., Guérin, R., Hosanagar, K., Zhang, Z.-L.: Dynamics of competition between incumbent and emerging network technologies. In Proceedings of the 3rd International Workshop on Economics of Networked Systems (NetEcon). ACM, pp. 49–54. (2008)
18. Sen, S., Jin, Y., Guerin, R., Hosanagar, K.: Modeling the dynamics of network technology adoption and the role of converters. IEEE/ACM Transactions on Networking **18**(6), 1793–1805 (2010)
19. Macy, M.W., Willer, R.: From factors to actors: Computational sociology and agent-based modeling. Ann. Rev. Sociol. **28**(1), 143–166 (2002)
20. Bonabeau, E.: Agent-based modeling: Methods and techniques for simulating human systems. Proc. Natl. Acad. Sci. USA **99**(Suppl 3), 7280–7287 (2002)
21. Gill, P., Schapira, M., Goldberg, S.: Let the market drive deployment: A strategy for transitioning to bgp security. In: Proceedings of the ACM SIGCOMM 2011 Conference. ACM, pp. 14–25. (2011)
22. Borshchev, A., Filippov, A.: From system dynamics and discrete event to practical agent-based modeling: Reasons, techniques, tools. In Proceedings of the 22nd international conference of the system dynamics society, no. 22, (2004)
23. Joseph, D., Shetty, N., Chuang, J., Stoica, I.: Modeling the adoption of new network architectures. In Proceedings of the 2007 ACM CoNEXT conference. New York, NY, USA: ACM, pp. 1–12, (2007)
24. Trinh, T., Gyarmati, L., Sallai, G.: Migrating to IPv6: A game-theoretic perspective. In: 2010 IEEE 35th Conference on Local Computer Networks (LCN), pp. 344–347
25. Chan, H., Dash, D., Perrig, A., Zhang, H.: Modeling adoptability of secure bgp protocol. In: Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. ACM, pp. 279–290
26. Sohn, S.Y., Kim, Y.: Economic evaluation model for international standardization of correlated technologies. IEEE Trans. Eng. Manag. **58**(2), 189–198 (2011)
27. Verbrugge, S., Pasqualini, S., Westphal, F.-J., Jäger, M., Iselt, A., Kirstä dter, A., Chahine, R., Colle, D., Pickavet, M., Demeester, P.: Modeling operational expenditures for telecom operators. In: 2005 Conference on optical network design and modelling. IEEE, pp. 455–466
28. Das, T., Drogon, M., Jukan, A., Hoffmann, M.: An agent-based modeling approach of network migration to new technologies. In: 2013 IEEE International Conference on Communications (ICC)
29. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: Openflow: Enabling innovation in campus networks. SIGCOMM Comput. Commun. Rev. **38**(2), 69–74 (2008)
30. Yannuzzi, M., Jukan, A., Masip-Bruin, X., Chamania, M., Serral-Gracia, R., Lopez-Alvarez, V., Gonzalez de Dios, O., Azanon, A., Maciejewski, M., Brunn, C., Roth, M., Altmann, J.: The internet and transport network management ecosystems: A roadmap toward convergence. In: 2012 16th International Conference on Optical Network Design and Modeling (ONDM), pp. 1–6
31. Vasseur, J., Zhang, R., Bitar, N., Roux, J. L.: A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths, RFC 5441 (Proposed Standard), Internet Engineering Task Force, Apr. (2009)
32. Merritt, R.: (2012, April 17) Google describes its openflow network. EE Times. http://www.eetimes.com/electronics-news/4371179/Google-describes-its-OpenFlow-network
33. Stephan, E., Zhao, Q., King, D.: Definitions of managed objects for path computation element discovery. http://tools.ietf.org/html/draft-ietf-pce-disc-mib-04
34. Marsden, P.V., Friedkin, N.E.: Network studies of social influence. Sociol. Methods Res. **22**(1), 127–151 (1993)
35. Katona, Z., Zubcsek, P.P., Sarvary, M.: Network effects and personal influences: The diffusion of an online social network. J. Market. Res. **48**(3), 425–443 (2011)
36. DeGroot, M.H.: Reaching a consensus. J. Am. Stat. Assoc. **69**(345), 118–121 (1974)
37. DeMarzo, P.M., Vayanos, D., Zwiebel, J.: Persuasion bias, social influence, and unidimensional opinions. Quart. J. Econ. **118**(3), 909–968 (2003)
38. Gao, L.: On inferring autonomous system relationships in the internet. IEEE/ACM Trans. Netw. **9**(6), 733–745 (2001)

39. Albert, R., Barabási, A.-L.: Topology of evolving networks: Local events and universality. Phys. Rev. Lett. **85**, 5234–5237 (2000)
40. He, Y., Faloutsos, M., Krishnamurthy, S., Chrobak, M.: Obtaining provably legitimate internet topologies. IEEE/ACM Trans. Netw. **20**(1), 271–284 (2012)

Tamal Das received his Ph.D. from IIT Bombay, India in 2012, and is a postdoctoral researcher in the Institut für Datentechnik und Kommunikationsnetze at TU Braunschweig, Germany since. His research interests are in stochastic analysis, telecommunication networks and network algorithms. Tamal was awarded the IEEE ANTS 2010 Best Paper Award and has authored numerous high quality scientific publications.

Marek Drogon obtained his B.Sc. and M.Sc. in Computer Science at TU Braunschweig, where he was one of the developers of the open-source PCE project (http://www.opensourcepce.org). His previous research work resulted in papers published in IEEE Journal of Lightwave Technology and a Post deadline paper at OFC conference.

Admela Jukan is Chair Professor of Communication Networks at the Technische Universitt Carolo-Wilhelmina zu Braunschweig (Brunswick) in Germany. Dr. Jukan has chaired and co-chaired several international conferences, including IFIP ONDM, IEEE ANTS, IEEE ICC and IEEE GLOBECOM. She serves as Associate Technical Editor for IEEE Communications Magazine. She is a co-Editor in Chief of the Elsevier Journal on Optical Switching and Networking (OSN), and an elected Chair of the IEEE Optical Network Technical Committee, ONTC (2014–2015). She is recipient of an Award of Excellence for the BMBF/CELTIC project "100Gb Ethernet" and was also awarded the IBM Innovation Award (2009). She is the Coordinating Principal Investigator of the FP7 EU-Project PACE (ict-pace.net).

Marco Hoffmann is Technology Manager for software defined networking and network virtualization and Project Manager for international projects in the Research division of NOKIA. He was consortium leader and board member of several national and international projects and member of company-internal and nation-wide Future Internet strategy teams. Additionally, he is core member of several EU-funded projects, like 5G-PPP.