An Agent-based Modeling Approach of Network Migration to New Technologies

Tamal Das, Marek Drogon and Admela Jukan
Technische Universität Carolo-Wilhelmina zu Braunschweig
Email: {das,drogon, jukan} @ida.ing.tu-bs.de

Marco Hoffmann
Nokia Siemens Networks
Email: marco.hoffmann@nsn.com

***Abstract*—Conventionally, network migration models study competition between emerging and incumbent technologies by considering the revenue increase and migration cost. We propose to extend the existing network migration models with new critical factors, including (i) synergistic relationships across multiple technologies, (ii) reduction in operation expenditures (OpEx), and, (iii) effect of social factors on human 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 optimal path computation with joint migration to two emerging networking paradigms, i.e., IETF Path Computation Element (PCE) and Software-Defined Networking (SDN). Our results demonstrate the synergistic effects of migration to multiple complementary technologies, and shows that a technology migration may be eased by the joint migration to multiple technologies.**

I. Introduction

Technical novelties in conjunction with economic factors decide the fate of an emergent technology, protocol, standard or product in the 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 gradual transition over time, requiring the interoperability and integration of different network applications, technologies and protocols. For instance, though the first IPv6 specification was 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 a higher capacity at a much faster pace. A typical carrier IP network is re-planned every 12-18 months and the IP links are designed so that maximum utilization at peak traffic loads is approximately 30%-40% [3]. Thus, there is no doubt that understanding the strategy and the investments for network migration, as well as the expected revenue and user growth are at the heart of every network migration process.

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, most models are based on the capital expenditures (CapEx) required to purchase the new technology. However, technology migration often results in reduction of operational expenditures (OpEx) that is gained over time, which is typically neglected in the current models. Finally, human decisions are majorly influenced by the social and behavioral factors involved in the process of migration. For example, herd mentality may be the actual cause of a technology adoption, over and beyond its technological merits.

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 individual migrations. In addition to CapEx, our model also incorporates the OpEx incurred in pre- and post-migration, which significantly affects an agent's decision to migrate. In the proposed model, an agent also incorporates its estimation about its neighbor's decision to migrate, in its own migration decision. We accomplish this by means of both deterministic and probabilistic heuristics. Finally, we study and present the equilibrium conditions of our model. 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.

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 [4]) and Software-Defined Networking (SDN [5]), respectively. The assumed network is a typical multi-vendor and multi-administration network, where separate *network islands* of routing systems need to cooperate for an end-to-end connection provision and are subject to migration decision pertaining to PCE, SDN, or both. PCE enable optimal path computation across network islands, an improved price/performance ratio, while, at the same time simplifying path computation operations [6]. All these benefits added together attracts considerably more users (and in turn traffic) to the network. Exchanges between PCE and network elements is, while standardized, 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, i.e, SDN, which facilitates configuration of all the network elements and thereby helps in setting up the computed paths. Thus, there is an implicit correlation between the deployment of PCE and SDN in a network.

The paper is organized as follows. Section II presents the related work, while Section III defines our multi-technology migration model and the case study. Section IV discusses our numerical results and Section V concludes the paper.

II. Related Work and Our Contribution

Network migrations are generally modeled using game-theory and implemented either by *system-dynamic* or *agent-based* models. Using either approach, i.e., system-dynamic or agent-based, the results demonstrate that the cumulative number of migrations increase over time showing '*S*'-like curves, which implies that a majority of migrations happen in a short time interval [7]. Despite comparable results, an agent-based approach is preferred over system dynamic approach in cases where the interactions between agents is non-uniform, i.e., where an agent does not interact with all the other agents.

The network migration problem has been typically studied for a single technology or protocol (e.g., IPv6 [8]–[11] or secure BGP [12], [13]), where it is assumed that an emerging protocol or technology substitutes an incumbent one. Even when multiple protocols are considered, such as S-BGP and soBGP [13], there is only a single prevalent protocol, and a decision is made by an agent to adapt to only one of the competing protocols. However, [14] shows that some correlated technologies may affect the migration to each other.

Prior migration studies focus on the migration of a single technology, and base an agent's migration decision on the CapEx involved [8]–[13]. The novelty of our approach, however, is in considering multi-technology migrations as opposed to a single technology migration. Further, we also incorporate both CapEx and OpEx in an agent's migration decision, instead of CapEx only. 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 [15]. However, the game-theoretic modeling of migration [8]–[13] have not yet considered it. 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 also novel in considering resulting revenue increase after migration as one of the factors affecting an agent's decision to migrate. We also consider the scenarios, where an agent migrates not only because of revenue increase, but also due to OpEx reduction, and, propose a novel method for an agent to estimate its neighbor's future strategies. Further, we consider *change* in revenue and OpEx as factors influencing an agent's decision, rather than the absolute values of the same, post-migration.

To validate our model, we propose a novel case study of multi-vendor enterprise network, considering the revenue of a network to vary with the the volume of traffic it transits through each network equipment 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 [16]. 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 it is well known today, inter-operablity of multi-vendor network islands remains a challenge, and a migration to standardized and programmable automated systems is an ongoing open challenge in carrier networks [17].

III. Multi-Technology Network Migration Model

In this Section, we first propose a generic agent-based model, which we later customize for the particular case study of multi-vendor path computation and provision.

A. 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* (both in *time* and *space*) 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 and space), 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 $N_1, N_2, \cdots, N_i, \cdots$. 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 individually (while, no such relationship is assumed to exist between $s_{u,0}$ and $s_{v,0}$). Here, $S_u = \{s_{u,0}, s_{u,1}\}$ represent the strategy of non-adoption and adoption of technology u, respectively. Similarly, $S_v = \{s_{v,0}, s_{v,1}\}$ represent 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 a given time-step. Thus, an agent's *strategy set* for any given time-step is denoted by $a_l = \{s_{u,k_1}, s_{v,k_2}\}$, where, $k_1, k_2 \in \{0, 1\}$. The volume of sales of agent N_i given its strategy set a_l is denoted by $T^i_{a_l}$. In this paper, we interchangeably use the notation of $s_{j,k}$ and s_{jk}.

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

$$\begin{aligned} C(a_l, T^i_{a'_l}) &\triangleq \text{CapEx of } N_i \text{ migrating from } a_l \text{ to } a'_l \\ R(T^i_{a'_l}) &\triangleq \text{Revenue of } N_i \text{ on adopting } a'_l \\ E(T^i_{a'_l}) &\triangleq \text{OpEx of } N_i \text{ on adopting } a'_l \end{aligned}$$

where, a_l is the current strategy set of N_i, a'_l is the strategy set to which N_i desires to migrate in the subsequent time-step and $T^i_{a'_l}$ denotes the *projected* traffic of N_i on migration to a'_l. The payoff derived by a network island on migrating from a_l to a'_l is thus given by the CapEx involved and the corresponding change in revenue and OpEx :

$$\begin{aligned} P_i(a_l \to a'_l) &= \Delta(\text{Revenue}) - \text{CapEx} - \Delta(\text{OpEx}) \\ &= \Big(R(T^i_{a'_l}) - R(T^i_{a_l})\Big) - C(a_l, T^i_{a'_l}) \\ &\quad - \Big(E(T^i_{a'_l}) - E(T^i_{a_l})\Big) \qquad (1) \end{aligned}$$

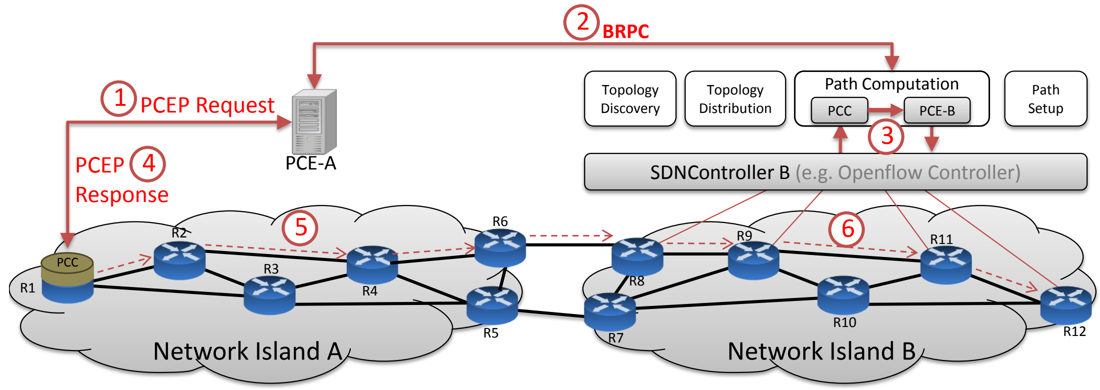


Fig. 1: Example of connection request setup in a multi-vendor network using both PCE and SDN.

Each agent optimizes its strategy choices at every time-step based on its payoff maximization.

Estimation of $T^i_{a'_l}$: The amount of sales of an agent primarily depends on the agent's strategy choices, which in turn is significantly affected by the strategies deployed by its neighboring agents. The general approach for an agent to estimate its neighbor's strategy is to assume that its neighbor maintains the same strategy set from the previous time-step (which is known because of our assumption of complete information) and choose its strategy accordingly. In this paper, we refer to this as the *standard approach*. However, such an estimate of strategy is too simplistic in nature and does not lead to effective decisions. Hence, we propose two heuristic approaches for an agent to estimate its neighbor's strategy in subsequent slots, based on *probabilistic* and *deterministic* methods. In both approaches, an agent approximates its neighbor's strategy to vary with the strategy of majority of its neighbors.

Strategy Estimation Approaches: In the *deterministic approach*, an agent assumes its neighbor to migrate if at least 50% of the neighbor's neighbors have migrated to a new strategy, while, in the *probabilistic approach*, an agent considers the probability of migration of its neighbor to be $x/100$, if $x\%$ of its neighbor's neighbors have migrated to a new strategy. Note that it is due to our assumption of complete information about network topology, that both these heuristics can be realized.

Equilibrium Characterization: For our model to reach a state of equilibrium, an agent's payoff by retaining its current strategy set should be at least as much as the payoff derived by migrating to any other strategy set. Moreover, it follows from our payoff model, that the payoff derived by an agent by retaining its current strategy set in the subsequent time-step is 0. Thus, at equilibrium, the payoff derived by an agent migrating to any strategy set, other its current strategy set, must be non-positive. Hence, the equilibrium conditions for our model are $P_i(a_l \rightarrow a'_l) \leq 0, \forall a'_l, \forall i$.

B. Case Study: PCE and SDN

In this sub-section, we apply our migration model to study the dynamics of migration to PCE and SDN. 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 an emerging carrier-Ethernet (connection-oriented) networks.

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 [18]. 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 OSPF and RIP) and Exterior Gateway Protocols (like Border Gateway Protocol) to compute paths by means of predefined routing tables entries.

SDN is an emerging networking architecture that facilitates programmability of the network control plane and its separation from the data plane [5]. It provides a centralized control interface to all the network elements that support SDN protocols, such as Open Flow [16] 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 COTS server with a high-end CPU is much cheaper than a high-end router [6]. Further, SDN offers the possibilities of dynamic network topologies and network virtualization, which makes it currently a highly popular and promising paradigm [19].

Fig. 1 illustrates an automated connection setup in a typical multi-vendor setting using PCE and SDN. Here, user R1 (vendor A) desires a connection to R12 (in vendor network B). Thus, since vendor A supports PCE, R1 sends a PCEP request to PCE-A for an optimal path to R12. PCE-A, in turn, requests PCE-B for information about paths to R12 and the corresponding costs. PCE-B queries its SDN controller and responds back to PCE-A with path options and the associated costs. PCE-A then computes a full path through islands A and B, taking into account the cost of paths in island A and the additional cost of the path options in island B. Finally, PCE-A informs R1 of the route to R12. 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 would tend to be highly proprietary in nature, and thus, due to lack of standards, hard to interoperate in a multi-vendor setting; that is where the IETF standardized approach with PCE comes in as an effective solution for interoperability.

2) The interplay involved in joint migration to PCE and SDN: As it 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 more detail.

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. 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 due to interoperability.

Within a network island, a SDN controller is *likely* to be able to provision a path, when a PCE cannot. 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 an SDN controller find paths that a PCE is requesting, but it can potentially even reconfigure the whole network such that a completely *new* path is configured to honor 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 migrates to SDN. On the other hand, as the PCEP 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, topology distribution; see Fig. 1), 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 hardware platform at a reduced cost.

In summary, network islands that migrate to PCE can compute optimal paths (i.e., with QoS), which can be setup 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.

3) Agent-based Model Applied: In this case study, agents translate to network islands; strategies correspond to technology choices; amount of sales relate to the amount of traffic that a network transits for its customers; technology u maps to PCE, while, technology v maps to SDN. Fig. 2 shows all possible strategy set transitions for a vendor, under the assumption that an island that has once migrated to $s_{\text{PCE},1}$ or $s_{\text{SDN},1}$ does not revert back to $s_{\text{PCE},0}$ or $s_{\text{SDN},0}$, respectively, in the future[1].

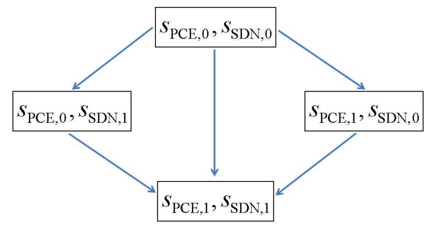


Fig. 2: Strategy set transitions in a network.

A network island incurs CapEx only if it changes its strategy. Thus, the CapEx from the generic model in Eq. (1) can be simplified, in this case, as

$$C(a_l, T^i_{a'_l}) = \sum_{s_{jk} \in a'_l} \delta_{s_{jk},a_l} c(s_{jk}, T^i_{a'_l})$$

where $c(s_{jk}, T^i_{a'_l})$ denotes the CapEx of adopting technology s_{jk} for a projected traffic value of $T^i_{a'_l}$, while,

$$\delta_{s_{jk},a_l} = \begin{cases} 0, & \text{if } s_{jk} \in a_l \\ 1, & \text{otherwise} \end{cases}$$

In consideration of economies of scale, the CapEx required for adoption of either technologies in a network is assumed to vary as the square root of traffic. However, in case of PCE deployment, the relationship between CapEx and traffic is weaker, which we capture using the cube root function, i.e.,

$$c(s_{jk}, T^i_{a'_l}) = \begin{cases} c_{\text{PCE}} \sqrt{T^i_{a'_l}} & \text{if } s_{jk} = s_{\text{PCE},0} \\ c_{\text{PCE}} \sqrt[3]{T^i_{a'_l}} & \text{if } s_{jk} = s_{\text{PCE},1} \\ c_{\text{SDN}} \sqrt{T^i_{a'_l}} & \text{if } s_{jk} \in S_{\text{SDN}} \end{cases}$$

where, $c_{\text{PCE}}, c_{\text{SDN}} \in [0, 1]$ are arbitrary coefficients.

We next consider the revenue of a network island in Eq. (1) to vary linearly with the traffic passing through the island, in case of PCE and SDN. And, given the *qualitative* nature of our model, without loss of generality, we set, $R(T^i_{a'_l}) = T^i_{a'_l}$.

The OpEx of PCE and SDN in a network island is independent of traffic and is thus constant, which given the qualitative nature of our study, without loss of generality, we set to zero. The OpEx for non-PCE and non-SDN technology choices in a network island is, however, approximated to vary linearly with the network traffic. Thus,

[1]This assumption can be justified because the functionalities provided by PCE and SDN are beneficial in a network irrespectively of external factors, such as the technology choices of other agents, etc.

$$E(T^i_{a'_l}) = \begin{cases} (\alpha_{\text{PCE}} + \alpha_{\text{SDN}})T^i_{a'_l} & \text{if } a'_l = (s_{\text{PCE},\,0}, s_{\text{SDN},\,0}) \\ \alpha_{\text{PCE}}T^i_{a'_l} & \text{if } a'_l = (s_{\text{PCE},\,0}, s_{\text{SDN},\,1}) \\ \alpha_{\text{SDN}}T^i_{a'_l} & \text{if } a'_l = (s_{\text{PCE},\,1}, s_{\text{SDN},\,0}) \\ 0 & \text{if } a'_l = (s_{\text{PCE},\,1}, s_{\text{SDN},\,1}) \end{cases}$$

where $\alpha_{\text{PCE},\,0}, \alpha_{\text{SDN}} \in [0, 1]$ are arbitrary coefficients.

Thus, with the above definitions of CapEx, Revenue and OpEx, as applicable for the case study of joint migration to PCE and SDN, the payoff function in Eq. (1), reduces to,

$$\begin{aligned} P_i(a_l \to a'_l) &= \Delta(\text{Revenue}) - \text{CapEx} - \Delta(\text{OpEx}) \\ &= \left(T^i_{a'_l} - T^i_{a_l}\right) - \sum_{s_{jk} \in a'_l} \delta_{s_{jk}, a_l} c(s_{jk}, T^i_{a'_l}) \\ &\quad - \left(E(T^i_{a'_l}) - E(T^i_{a_l})\right) \end{aligned}$$

IV. Numerical Results

Simulation Model: For the network topology studied, we generate a scale-free network comprising of 100 interconnected network islands, out of which 52 are "transit" and 48 are "stub" islands, akin to the terminology used in global Internetworks[2] [20]. The topology is created using Barabási and Albert's topology generation algorithm [21], where the seed network comprised of two 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. Finally, stub islands are considered as end-users, while technology migration is studied in the transit islands only. At each time-step, we consider 10 units of traffic from every stub to every other stub in the network. In this topology, we employ No-Valley-Prefer-Customer (NVPC) routing algorithm [22]. If multiple shortest paths exist, traffic is uniformly distributed across all of them. The parameter values assumed in the simulation are $c_{\text{PCE}} = c_{\text{SDN}} = 0.7$ and $\alpha_{\text{PCE}} = \alpha_{\text{SDN}} = 0.008$, unless otherwise mentioned.

Effect of Early Adopters: In this experiment, we study the effect of early adopters on migration profile. An early adopter is a network island that has migrated to both PCE and SDN since the start of simulation. We choose candidates for early adoptions based on the degree of connectivity.

Fig. 3 plots the migration profile for two scenarios, namely, (1) no early adopters, (2) two transit islands with highest connectivity as early adopters. Early adopters act as the seed for migration in the network, thereby catalyzing the migration process. This results in higher migration rate as in Fig. 3.

Effect of Complementary Technologies: Fig. 4 contrasts the migration profile for single migrations (where, migration of only one technology is considered in isolation) with joint migrations (where, two technologies simultaneously migrate). We observe that, when considered in isolation, PCE is easily adopted, while, SDN is hard to adopt. However, when both migrations are simultaneously considered, the complementary relationship between PCE and SDN leads to a better migration profile for SDN. In essence, migration to a hard-to-adopt technology can be eased by an accompanying migration to a complementary, easy-to-adopt technology.

[2] A network island, which is not a provider for any other island, is called a *stub island*, while all other islands are called as *transit* islands

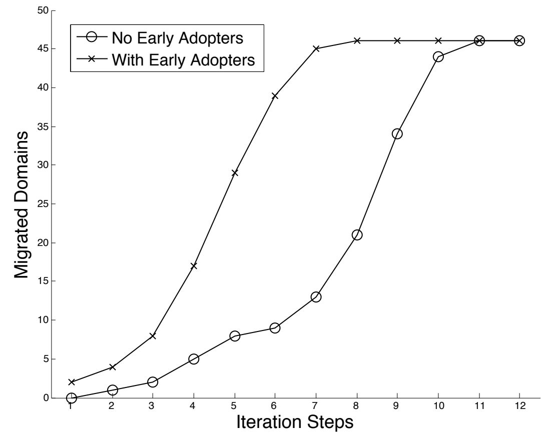


Fig. 3: Effect of type of early adopters on the migration profile

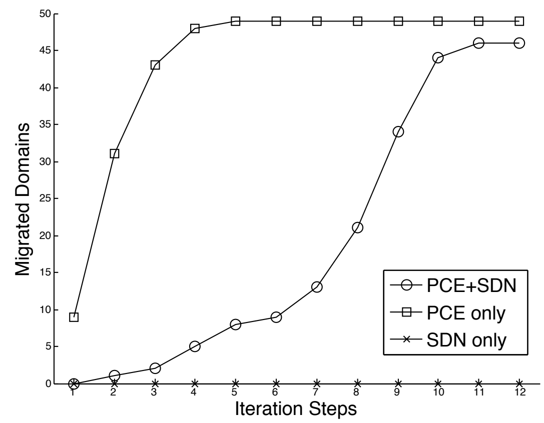
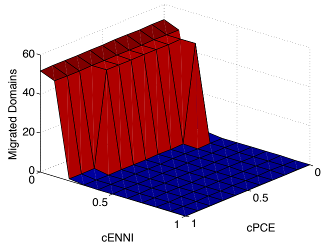


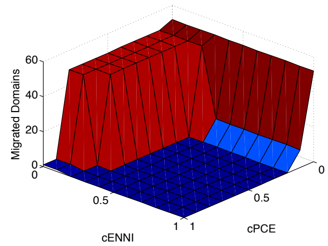
Fig. 4: Complementary effect of migration to PCE and SDN

Effect of OpEx: In this experiment, we study the effect of introduction of OpEx reduction in our migration model. We, thus, compare the migration profile with varying degrees of reduction in OpEx, which is controlled by α_{SDN} and α_{PCE} in our model. We consider low ($\alpha_{\text{PCE}} = \alpha_{\text{SDN}} = 0.0008$), medium ($\alpha_{\text{PCE}} = \alpha_{\text{SDN}} = 0.008$) and high ($\alpha_{\text{PCE}} = \alpha_{\text{SDN}} = 0.08$) values of OpEx reduction. Fig. 5 plots the separate migration profiles to PCE and SDN for varying degrees of OpEx reduction and varying values of CapEx for both PCE and SDN migration. As we see from Fig. 5, the migration profile improves with the extent of OpEx reduction. For high values of α_{PCE} and α_{SDN}, we observe complete migration. This demonstrates the novelty of our model in incorporating the effect of OpEx reduction on migration.

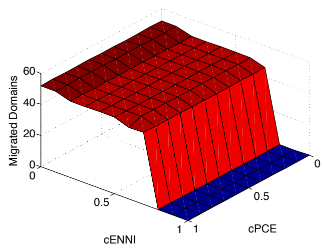
Effect of Migration Strategy: In this experiment, we study the effect of strategy estimation approaches on migration decisions. Fig. 6 plots the extent of migration to both PCE and SDN for varying degrees of risk aversion of a network island. Here, an island which migrates only given high degree of migration amongst its neighbors is considered to be more risk-averse than one who does so even given a lower degree of migration amongst its neighbors. As it can be seen, standard, probabilistic and deterministic approaches all lead to similar migration profile in the initial few time-steps. In the intermediate stage, however, we observe a striking ordering pattern between the deterministic (DET=30%), probabilistic (PROB=30%) and standard approaches, which eventually



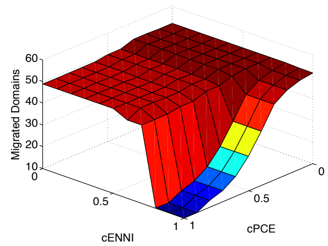
(a) SDN: Low OpEx reduction



(b) PCE: Low OpEx reduction



(c) SDN: Medium OpEx reduction



(d) PCE: Medium OpEx reduction

Fig. 5: Migration pattern with varying OpEx and CapEx costs.

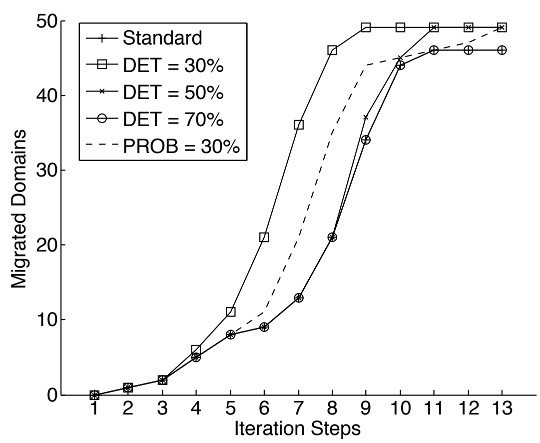


Fig. 6: Migration with varying degrees of risk aversion.

overlap in the final stages. The migration rate and the final number of migrants is observed to be the highest in case of deterministic estimation, followed by probabilistic estimation, followed by standard approach. This is because, the standard approach is the most conservative about its migration decision, followed by the probabilistic and the deterministic approach.

We also observe from Fig. 6, that amongst deterministic approaches, the degree of migration is the same for DET=70% and DET=0% (which is the standard approach), while it improves for DET=50% and peaks at DET=30%. This plot implies that a global predominance of risk-loving islands leads to a faster and better migration profile.

V. 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. The proposed model yields the usual *S*-like curve of the cumulative number of migrated agents. In addition, our model suggests that the migration to a technology can be promoted by several factors, namely, (a) complementary technologies, (b) early adopters, (c) an agent's estimation of its neighbor's decision to migrate. The results indicate that presence of few well-connected early adopters, simultaneous migration of easy-to-adopt complementary technologies and strategy estimation approaches facilitate migration. Our future work will include the total budget constraint in a multi-vendor, multi-layer network migration scenario with IP optical integration as well as study with the relaxed assumptions of uniform traffic, invariant traffic demand and the inter-island topology in the simulations.

Acknowledgements

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under support code 01BP12300A; EUREKA-Project SASER. We also thank Mohit Chamania, Marcel Caria and Kibae Kim for their contributions in the earlier phases of this research.

References

[1] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," IETF RFC 2640, Dec 1998, http://tools.ietf.org/html/rfc2460
[2] S. Johnson, "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] A. Farrel, J.-P. Vasseur, and J. Ash, "A Path Computation Element (PCE)-based architecture," IETF RFC 4655, 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4655.txt.
[5] Software-Defined Networking:The New Norm for Networks; ONF white paper, April 13, 2012.
[6] Metaswitch, PCE - An Evolutionary Approach to SDN, Available: www.pcetosdn.com/downloads/PCE-Evolutionary-SDN-feb2012.pdf.
[7] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," Proc. of Nat. Acad. Sci., vol. 99, pp. 7280-7287, 2002.
[8] Y. Jin, et al. "Dynamics of competition between incumbent and emerging network technologies," in NetEcon '08, New York, NY, 2008.
[9] S. Sen, et al. "Modeling the dynamics of network technology adoption and the role of converters," IEEE/ACM Trans. Netw., vol. 18, no. 6, pp. 1793-1805, 2010.
[10] D. Joseph, N. Shetty, J. Chuang, and I. Stoica, "Modeling the adoption of new network architectures," in ACM CoNEXT 2007.
[11] T. A. Trinh, L. Gyarmati and G. Sallai, "Migrating to IPv6: A game-theoretic perspective," IEEE LCN, Denver, CO, 2010.
[12] P. Gill, et al. "Let the market drive deployment: A strategy for transitioning to BGP security," in ACM SIGCOMM, Toronto, Ontario, 2011.
[13] H. Chan, et al. "Modeling adoptability of secure BGP protocols," in ACM SIGCOMM, Pisa, Italy, 2006.
[14] S.Y. Sohn and Y. Kim, "Economic Evaluation Model for International Standardization of Correlated Technologies," Engineering Management, IEEE Transactions on , vol.58, no.2, pp.189-198, May 2011
[15] S. Verbrugge, et. al., "Modeling operational expenditures for telecom operators," 9th Conference on Optical Network Design and Modelling, Milan, Italy, Feb. 2005.
[16] Nick McKeown, et. al. , "OpenFlow: Enabling innovation in campus networks," SIGCOMM Comput. Commun. Rev. 2008, 38, 2, 69-74.
[17] M. Yannuzzi, et. al. , "The Internet and Transport Network Management Ecosystems: A Roadmap Toward Convergence," in Optical Networking Design and Modeling (ONDM)", Apr 2012
[18] J. P. Vasseur, et al. "A Backward-Recursive PCE-based Computation (BRPC) procedure to compute shortest constrained inter-domain traffic engineering label switched paths," IETF RFC 5441, 2009.
[19] "Google g-scale network." Available: http://www.eetimes.com/electronics-news/4371179/Google-describesits-OpenFlow-network
[20] L. Gao, "On inferring autonomous system relationships in the internet," IEEE/ACM Trans. Netw., vol. 9, no. 6, pp. 733-745, 2001.
[21] R. Albert, and A.-L. .Barabási, "Topology of evolving networks: Local events and universality," Phys. Rev. Lett., vol. 85, no. 24, pp. 5234-5237.
[22] Y. He, M. Faloutsos, S. V. Krishnamurthy and M. Chrobak "Obtaining provably legitimate Internet topologies," IEEE/ACM Trans. Netw., vol. 20, no. 1, pp. 271-284.