A Hybrid Management Substrate Structure for Adaptive Network Resource Management

Daphne Tuncer, Marinos Charalambides, Hisham El-Ezhabi, and George Pavlou
Department of Electronic and Electrical Engineering
University College London
Email: {d.tuncer, m.charalambides, h.elezaby, g.pavlou}@ee.ucl.ac.uk

***Abstract*—Centralized and offline network management functionality, traditionally deployed by operators, cannot easily deal with the traffic patterns of emerging services, which are becoming more dynamic and unpredictable. As such, decentralized solutions that are flexible and adaptive to traffic and network dynamics are of paramount importance. To this end, we have been developing an in-network management approach in which an intelligent substrate allows the dynamic reconfiguration of resources according to network conditions. The set of nodes forming this logical structure are able to communicate with each other to coordinate their decisions. While in previous work we investigated the use of full-mesh and ring structures to connect the substrate nodes, we consider here a hybrid approach that combines the benefits of the other two. We describe algorithms that can be practically used to compute this hybrid structure and that take into account important criteria such as minimizing the latency and the communication overhead among the substrate nodes. We evaluate the impact of key parameters associated with the construction process.**

I. Introduction

Network resource management approaches traditionally deployed by operators rely on offline functionality that cannot easily deal with the traffic patterns of emerging services, which are becoming more dynamic and unpredictable. As such, solutions that are flexible and adaptive to traffic and network dynamics are of paramount importance. Furthermore, network resource management normally relies on centralized managers that periodically compute new configurations according to dynamic traffic behaviors. These centralized approaches have limitations especially in terms of scalability (i.e. communication overhead between the central manager and devices at runtime) and lag in the central manager reactions that may result in sub-optimal performance. To meet the requirements of emerging services, network resource management functionality that is decentralized, flexible, reactive and adaptive to traffic and network dynamics is necessary.

To overcome the limitations of current approaches, recent research efforts have proposed a new in-network management framework for dynamic resource reconfiguration in fixed backbone networks [1]. According to the proposed framework, the decision-making process is distributed across network edge nodes, so that each node is responsible for deciding on reconfiguration actions based on local feedback regarding the state of the network. Nodes are equipped with the necessary logic that enables them to perform reconfigurations, so that the network resources can be better utilized. In order to avoid inconsistencies between several independent decisions, the network nodes cooperatively decide on the most suitable changes to apply depending on network characteristics and conditions. The network nodes participating in the resource management process form a *management substrate*, which is a logical structure used to facilitate the exchange of information between distributed decision-making points. Such a framework was used in our previous work for the purpose of adaptive traffic engineering [2], energy efficiency [3] and in-network cache management [4]. However, due to the distributed nature of the decision-making process, the performance of the proposed management scheme, in terms of communication overhead, can be affected by the structure of the management substrate. In [2], we have considered the use of simple topology structures (i.e. full-mesh and ring), which may have some limitations in terms of scalability in practice. In this paper, we propose a more sophisticated structure to connect the network edge nodes. The proposed hybrid model is a combination of the ring and the full-mesh structures. We investigate a set of methods to compute the corresponding hybrid structure given a set of network nodes and the underlying physical topology, with the objective of minimizing the latency and the communication overhead among the substrate nodes.

II. Related Work

Interaction and communication between autonomic elements have been described as fundamental architectural features of autonomic computing systems in [5]. In particular, the authors highlight that autonomic elements can establish relationships between each other in order to request or offer a service. A generic model based on negotiation is proposed to drive the interaction between autonomic elements. Other generic interaction models have been considered in [6], where four types of behavior that can be exhibited by an autonomic element towards other autonomic elements are described, i.e. the cooperative, selfish, punishment and mixed behaviors. Communication models between network entities to support management tasks have also been considered in [7] in the context of autonomic networks. In this work, the interaction between the decision elements relies on a hierarchical structure in which the decisions taken by each decision element are orchestrated by one or more "arbiter" elements that are in charge of detecting potential overlapping or contracting actions and configurations.

Some research efforts have also investigated the use of generic hierarchical architectures inspired by multi-agent systems to support interaction and cooperation between nodes, e.g. [8] [9]. The use of gossip-based protocols to propagate information across distributed decision-making points was considered in [10] [11] [12]. According to gossip-based approaches, the interaction between nodes relies on a random process, so that at regular time intervals, one node in the network initiates a communication with a randomly selected neighbor in order to exchange information. The work in [10] focused on the development of scalable and adaptive mechanisms for calculating aggregates in a pro-active manner. A gossip-based approach was used in [11] for dynamic resource allocation in cloud environments and in [12] for the development of decentralized self-adaptive aggregation mechanisms.

The design of logical infrastructures to connect a set of nodes has received significant attention from the research community over the last decade, especially in the context of overlay networks [13] [14]. While research efforts in this area have focused on developing scalable systems through optimized logical topologies and overlay routing protocols, the purpose of the work presented in this paper is not to investigate features and techniques to support overlay systems. It focuses, instead, on the design of topology structures that can offer trade-off performance in terms of communication cost and management overhead for supporting the interaction between network reconfiguration entities.

III. In-Network Management Substrate

A. Decentralised Resource Management Framework

In the proposed in-network resource management framework, network edge nodes are embedded with a level of intelligence that allows them to react to network conditions in a decentralized and adaptive fashion based on periodical feedback information received from the network. Compared to centralized offline solutions, where reconfigurations are decided by a centralized management system that has a global knowledge about the network, reconfiguration decisions are directly taken by the network edge nodes that coordinate among themselves in order to decide upon the best sequence of actions to perform to satisfy a common resource optimisation objective. In order to support this decentralized decision-making process, the network edge nodes are organized into a *management substrate* (MS), which is a logical structure used to facilitate the exchange of information between decision-making entities. The management substrate is used by the edge nodes for coordination purposes, in particular, since it provides a means through which nodes can communicate. It is worth mentioning that the substrate is only used for signalling, and not for direct traffic routing/forwarding.

A management substrate structure example is depicted in Fig. 1, where each network edge node N is logically connected to a set of other network edge nodes (neighbor nodes in the substrate). Any MS node can directly communicate only with its neighbors, which are defined by the topological structure used. The choice of the substrate topology can be driven by different parameters related to the physical network, such as its topology, the number of edge nodes, but also by the constraints of the coordination mechanism between the nodes and the associated communication protocol. The overhead incurred by the communication protocol in terms of delay and the number of messages exchanged, for example, is a key factor that can influence the choice of the structure.

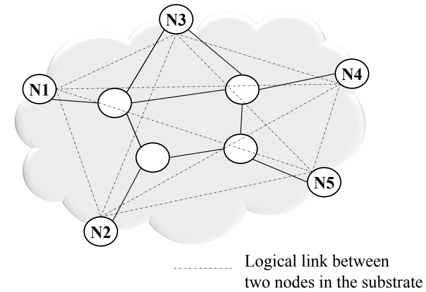

Fig. 1. In-network management substrate overview.

In the rest of this paper, we note $\mathcal{L}$ the set of network links and $\mathcal{N}$ the set of network edge nodes. In addition, it is assumed that that the network nodes do not fail.

B. Substrate Characteristics

The main objective considered for the design of each structure is to minimize the communication overhead incurred by the coordination process. This is defined by the volume of signalling messages and the delay, which is driven by the communication cost between MS nodes.

The communication cost between two MS nodes N_i and N_j is defined as the cost of the logical link between the two nodes. This cost, denoted $C_{\mathrm{LL}}\left(ij\right)$, is defined by the cost of the path between node N_i and node N_j in the underlying physical network topology. This is equal to the sum of the cost of the links involved in the path:

$$C_{\mathrm{LL}}\left(ij\right) = \sum_{l \in \mathcal{L}} \delta_{ij}^{l} \cdot c\left(l\right) \tag{1}$$

where δ_{ij}^{l} is a $\{0-1\}$ binary variable equal to 1 if link l is included in the path between nodes N_i and N_j, and $c\left(l\right)$ is the cost of link l.

The cost $c\left(l\right)$ of link l can be defined, for instance, according to the administrative cost (i.e. link weight) which is the metric used to compute the shortest paths. Administrative costs are usually assigned based on the characteristics of the underlying physical network topology and on traffic engineering requirements. A common practice is to set link weights equal to the inverse of the link capacities [15]. These costs may not, however, be sufficient to account for the communication cost in terms of delay between two nodes since the delay is also influenced by the geographical distance between the nodes (i.e. propagation delay). In order to take the geographical distance into account, an additional metric, called link distance factor (c_{φ}), is defined for each link. This represents the relative

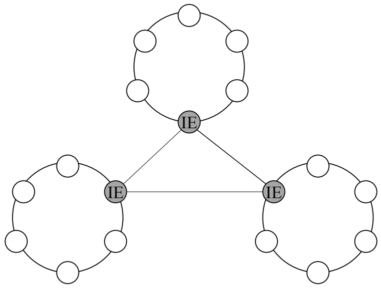

Fig. 2. Hybrid topology structure.

distance between two nodes in the network and is defined as the ratio between the geographical distance d_l (e.g. in kilometres) obtained for each link l divided by the smallest geographical distance observed in the network:

$$c_{\varphi} = \frac{d_l}{\min_{l \in \mathcal{L}} (d_l)} \tag{2}$$

The cost of a link l is then defined as the product of the link administrative cost c_α and the link distance factor c_φ:

$$c(l) = c_\alpha \cdot c_\varphi \tag{3}$$

It is assumed that the path used between two nodes is the shortest path and that all network links are bidirectional, so that for any pair of nodes N_i and N_j, $C_{LL}(ij) = C_{LL}(ji)$. Finally, it is worth noting that reliable transfer of messages between nodes is assumed and that the proposed framework relies on the Transmission Control Protocol (TCP) [16] as the underlying transport protocol.

C. Hybrid Management Substrate Structure

In previous work we have used simple logical structures to connect MS nodes, i.e. full-mesh and ring structures. While a direct logical link exists between all network edge nodes in the full-mesh topology (as depicted in Fig. 1), each node in the MS is connected to two other nodes only, in the ring model. Due to their characteristics, both models present some limitations in terms of scalability when the number of MS nodes increases. In the case of the full-mesh, this incurs a significant increase in the volume of substrate information to be maintained locally at each MS node and in the case of the ring model, it significantly affects the total communication delay [2]. In order to overcome the limitations of these two simple structures, the design of a more sophisticated model is investigated to organize MS nodes. This model, referred to as a hybrid topology, is a combination of the ring and full-mesh structures, as depicted in Fig. 2.

The hybrid topology consists of a set of rings inter-connected in a fully-meshed fashion through *Intermediate Entity* (IE) nodes, so that there exists exactly one IE node in each ring. More specifically, MS nodes are partitioned into at least two clusters, so that nodes in each of the clusters are connected according to a ring topology. One node is then selected in each cluster to be the IE, i.e. to act as an interface to the other clusters. It is worth noting that each MS node belongs to one cluster only. One of the incentives for using a hybrid structure is to provide a trade-off in terms of performance between the message overhead and the delay incurred when two MS nodes need to exchange information. Such a trade-off raises some requirements when deciding how to connect nodes according to the hybrid model. The next section presents a set of methods that define how to partition the MS nodes into clusters and how to select the IE node in each cluster.

IV. Hybrid Model Construction

A. Ring Model Construction

One of the objectives of the hybrid topology structure is to obtain better performance than the ring model in terms of communication delay between MS nodes. Given the characteristics of the ring topology, the communication between MS nodes relies on a hop-by-hop mechanism [2]. In order to communicate with any other node, a message needs to be sent over the ring until it reaches its destination. Given that the total communication cost can be defined as the sum of the cost between all successive nodes, it is affected by the order according to which the nodes in the ring are connected.

This problem is similar to the Travelling Salesman Problem (TSP) [17]. The TSP is a well-know NP-Hard combinatorial optimization problem that consists in determining, given a list of locations and their pairwise distances, the shortest possible route that visits each location exactly once and that returns to the starting location. Although a number of approaches with near-optimal performance exists in the literature to solve the TSP, they are computationally expensive. In order to keep the complexity of the construction algorithm low, an approach based on the simple *Nearest Neighbors* tour construction heuristic [18] has been developed. This has a time complexity $O(N^2)$, with N being the number of nodes to consider.

The principle of the proposed approach is as follows. Given a node N_i, node N_j is selected as the successor of N_i such that the cost $C_{\mathrm{LL}}(ij)$ is the lowest. The *Nearest Neighbor* algorithm considers each node N_i iteratively and selects, among other MS nodes that have not already been considered, the successor of N_i, i.e. the node with the lowest logical link cost to N_i. The algorithm terminates when all nodes have been considered and the successor of the last node is set to be the initial node.

B. Constructing Multiple Rings

A key challenge when forming the hybrid structure is to determine which metric to use in order to partition the MS nodes into clusters. A natural choice is to use the logical link cost metric defined in section III-B, which is a function of the link administrative cost and the geographical distance. As such, nodes are clustered based on their proximity with respect to the logical link cost.

In order to reduce the communication delay compared to the ring structure, the total communication cost permitted in each sub-ring of the hybrid structure needs to be less than an upper bound threshold θ. The value of the threshold is a key factor since it can influence whether a node should be considered as a member of a specific ring or not, and, as such, directly affects the size of each sub-ring. To set the appropriate threshold value, it is also essential to take into account the fact that nodes located in different sub-rings can communicate. Given that nodes can directly communicate in the full-mesh model, it can be inferred that the communication cost in this model is less than the cost in the ring model. As such, we use the maximum logical link cost obtained if MS nodes were connected in a full-mesh fashion as a reference metric to derive the value of the threshold to apply when constructing sub-rings. Two cases are investigated:

1) θ is equal to θ_{HalfMax}, i.e. to half of the maximum logical link cost obtained in the full-mesh case.
2) θ is equal to θ_{Avg}, i.e. to the average logical link cost obtained between all possible pairs of nodes in the full-mesh case.

An approach has been designed to partition the MS nodes into the different clusters according to θ, and compute the resulting sub-rings. The proposed algorithm follows an iterative process where all MS nodes are considered one-by-one. The number of clusters is not determined a priori. One cluster is initially formed by the algorithm and nodes are successively added to this cluster until the threshold condition is violated. In this case, the initial cluster is said to be complete and a new cluster is formed to accommodate the remaining nodes. The different clusters are thus formed successively according to the threshold value θ and the order in which nodes are considered. To ensure that each node belongs to one cluster only, the algorithm maintains the list of MS nodes that have not been considered yet. The list contains initially all MS nodes and is updated at each iteration by removing the node selected by the algorithm. The output of the algorithm is a set of rings.

The steps of the algorithm are as follows. N_{curr} is the node considered by the algorithm at each iteration and $\mathcal{N}_{\text{wait}}$ is the list of the MS nodes that have not been considered yet. N_{ini} is the initial node and $\mathcal{S}_{\text{RINGS}}$ is the set of constructed rings.

1) Select an initial node N_{ini}, set N_{curr} to N_{ini} and remove N_{ini} from $\mathcal{N}_{\text{wait}}$.
2) Create a new cluster $\mathcal{C}$ with N_{curr}.
3) Compare the cost of the logical links from N_{curr} to all nodes in $\mathcal{N}_{\text{wait}}$. Select the pair (i.e. logical link) with the lowest cost and mark the relevant peer node as N_{test}.
4) Apply the ring construction algorithm described in section IV-A to the set of nodes formed by the union of the set of nodes in cluster $\mathcal{C}$ and N_{test}. Determine the total ring cost C_{ring}.
5) Compare C_{ring} to the threshold value θ. If $C_{\text{ring}} \leq \theta$, add N_{test} to cluster $\mathcal{C}$, remove N_{test} from $\mathcal{N}_{\text{wait}}$ and set N_{curr} to N_{test}. Go back to step 3. If $C_{\text{ring}} > \theta$, apply the ring construction algorithm to nodes in cluster $\mathcal{C}$ and add the resulting ring to $\mathcal{S}_{\text{RINGS}}$. Set N_{curr} to N_{test}, remove N_{test} from $\mathcal{N}_{\text{wait}}$ and go back to step 2.
6) Continue until $\mathcal{N}_{\text{wait}}$ is empty.

Two criteria to select the initial node are investigated. In the first case, the node connected to the logical link with the lowest cost in the MS is selected as the initial node, while, in the second case, the node connected to the logical link with the highest cost in the MS is selected. Given that logical links are bidirectional, the node with the lowest identifier is selected by default.

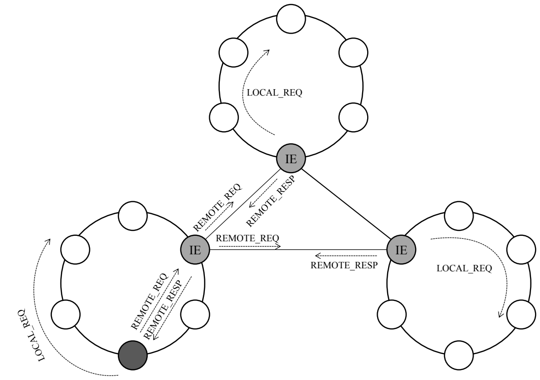

Fig. 3. Overview of the hybrid topology communication model.

There may be cases where some of the sub-rings obtained contain one element only, which is not acceptable by definition. The structure of each sub-ring is therefore analyzed at the end of the algorithm. If single node sub-rings are found, the algorithm disregards them and assigns the involved nodes to other sub-rings, so that the selected rings are those for which the addition of an extra node leads to the lowest increase in terms of cost.

C. Intermediate Entity Selection

Another key issue raised by the design of the hybrid topology is the selection of the most appropriate IE in each sub-ring, so that these can be efficiently inter-connected in a full-mesh. In a similar fashion to the method used to select successor nodes in the ring construction algorithm, IE nodes are chosen according to their proximity, in terms of logical link cost, to other rings. This can be formally described as the problem to determine which node to select in each sub-ring so that the maximum logical link cost between all pairs of the IE nodes is minimized. In order to simplify the IE selection procedure, we have investigated a heuristic to select the node in each sub-ring that is the closest, on average, to every other remote node in the substrate (i.e. to the nodes in other sub-rings). The proposed approach relies on an iterative process, where sub-rings are considered one-by-one, so that at each iteration, one node in the considered sub-ring is selected as the IE. In order to select the appropriate IE in each ring, the algorithm computes, for each node in the ring, the average logical link cost to every other remote node. The selected IE in each ring is the one with the lowest average cost.

V. Communication Protocol For Management Operations

A. Communication Model

The protocol for the communication between the MS nodes organized into a hybrid structure supports two modes of communication as depicted in Fig. 3 and described below.

Local Sub-ring Communication: The first mode concerns the communication between nodes located in the same ring.

This mode corresponds to the case where the node that initiates the communication (represented by a gray disc in the figure) needs to exchange some information with (an)other node(s) in the local sub-ring. In that case, the initiator node sends a local request in the form of a $LOCAL\_REQ$ message to one of its neighboring nodes according to the communication direction followed. The message then travels hop-by-hop through the ring until it reaches the initiator node again.

Remote Sub-ring Communication: The second mode concerns the communication between nodes located in different rings, when for example the initiator node needs to retrieve information from a node located in a remote sub-ring. To do this, the initiator needs to first communicate with its local IE since this node acts as the interface to the other rings. It is assumed that the address of the IE in a given sub-ring is known by all the nodes of that ring. The initiator starts by sending a remote request ($REMOTE\_REQ$) message directly to its IE node, which then forwards it to all the other IE nodes of the MS. Each IE is subsequently responsible for circulating a $LOCAL\_REQ$ message in its local ring. Upon receiving this message back, each IE analyzes its content and creates a remote response ($REMOTE\_RESP$) message that contains information about potential satisfactory replies from its ring. This is sent back to the original requesting IE, which forwards it to the initiator.

B. Communication Overhead

In the full-mesh MS topology model, the communication overhead incurred when a node requests information is proportional to the number of nodes in the MS, since a message is exchanged with every other node [2]. According to the communication protocol used in the hybrid model, the total number of messages exchanged depends on the communication mode considered. For the local sub-ring communication case, only one message needs to be sent by each node: a $LOCAL\_REQ$ message to its direct neighbor. For the remote sub-ring communication case, however, with r being the number of sub-rings, one $REMOTE\_REQ$ message is sent by the initiator node to the local IE and $(r-1)$ $REMOTE\_REQ$ messages are sent from the local IE to other IE nodes in the substrate. As such the communication overhead in terms of number of messages in the hybrid model is, in the worst case, proportional to the number of sub-rings. Compared to the full-mesh topology, the performance of the hybrid model improves as the size of each ring increases, and consequently, as the number of rings decreases. Given the hybrid nature of the model, it can be deduced that the communication cost in terms of delay will be driven by the characteristics of the full-mesh and ring structures. It can therefore be inferred that the total delay will be influenced by the size of the largest sub-ring and the maximum distance between IE nodes.

VI. Evaluation

A. Experiment Settings

The impact of key parameters associated with the construction process has been evaluated and analyzed using two real PoP (Point of Presence)-level network topologies, Abilene [19] and GEANT [20]. The PoPs, in each topology, are mapped to cities, which enables us to determine the geographical distance between every pair of nodes. While the full 11-node topology is used in the case of the Abilene network, a reduced GEANT topology, which excludes the two non-European PoPs, i.e. 21 instead of 23 nodes, is considered.

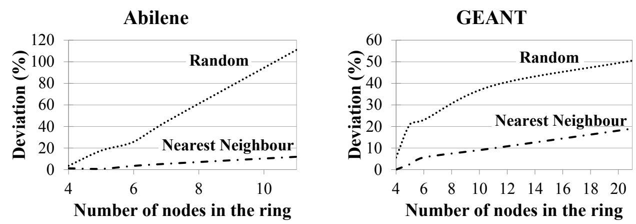

Fig. 4. Evolution of the deviation from the optimum in case of the Abilene and GEANT networks.

To evaluate the proposed topology models, a Java program that computes the MS topology structure corresponding to any physical network topology has been developed. The program takes as input the network topology, the identifiers of the network edge nodes and a set of configuration parameters. The latter enable the user to control the type of MS structure to compute (ring, full-mesh, hybrid), the logical link cost model, the threshold value and the initial node selection criterion.

B. Ring Construction

To evaluate the performance of the ring construction algorithm described in section IV-A, the total ring cost obtained for a set of nodes is compared to the cost of the optimal ring structure [17], which is computed using the GLPK linear/mixed integer programming solver [21]. We also consider the performance obtained by a method that randomly connects the nodes in a ring. To analyze the influence of the number of nodes on the performance of the algorithm, we have performed experiments using different number of nodes in both the Abilene and GEANT networks. A subset of nodes is randomly selected and connected into a ring according to the three methods mentioned above. The deviation of the total ring cost obtained with the proposed and the random algorithms from the optimum, for different number of nodes, is depicted in Fig. 4. The deviation increases linearly with the number of nodes for both algorithms. Given that the proposed approach follows an iterative process where nodes are iteratively added to the ring structure, the error introduced at each iteration by the choice of a successor node incurs a cost penalty to the total ring cost. The penalty increases as the number of nodes to consider increases, and, as a result, the deviation from the optimum increases. It can be noticed, however, that the proposed algorithm outperforms the random one since the deviation is significantly lower in all cases.

C. Multiple Rings Construction

This subsection provides an analysis of how the logical link cost C_{LL}, the threshold θ and the initial node selection criterion can influence the structure of the sub-rings (i.e. number and

TABLE I
MULTIPLE RINGS EVALUATION.

Cost C_{LL}	Threshold θ	Initial Node	Rings Size
GEANT Topology			
$M_{1\times D}$	θ_{Avg}	Lowest	4,6,3,2,2,2,2
$M_{1\times D}$	θ_{Avg}	Highest	5,2,3,3,3,3,2
$M_{1\times D}$	$\theta_{HalfMax}$	Lowest	4,6,3,2,2,2,2
$M_{1\times D}$	$\theta_{HalfMax}$	Highest	5,2,2,3,2,5,2
$M_{W\times D}$	θ_{Avg}	Lowest	9,6,3,3
$M_{W\times D}$	θ_{Avg}	Highest	2,5,7,5,2
$M_{W\times D}$	$\theta_{HalfMax}$	Lowest	15,6
$M_{W\times D}$	$\theta_{HalfMax}$	Highest	15,6
Abilene Topology			
$M_{1\times D}$	θ_{Avg}	Lowest	3,2,3,3
$M_{1\times D}$	θ_{Avg}	Highest	3,2,3,3
$M_{1\times D}$	$\theta_{HalfMax}$	Lowest	3,2,3,3
$M_{1\times D}$	$\theta_{HalfMax}$	Highest	3,2,2,2,2
$M_{W\times D}$	θ_{Avg}	Lowest	5,3,3
$M_{W\times D}$	θ_{Avg}	Highest	3,2,3,3
$M_{W\times D}$	$\theta_{HalfMax}$	Lowest	5,3,3
$M_{W\times D}$	$\theta_{HalfMax}$	Highest	3,2,3,3

size) computed according to the algorithm described in section IV-B. A comparison of the structures obtained when using the threshold values θ_{HalfMax} and θ_{Avg}, and the initial node selection criterion lowest C_{LL} and highest C_{LL}, as explained in section IV-B, is performed. In addition, two different cases for the logical link cost are considered: a) the administrative link weight of each network link is set to 1 (i.e. in this case, the C_{LL} is mainly driven by the geographical distance between the nodes), and, b) the administrative link weights are the original ones. These are denoted as $M_{1\times\text{D}}$ and $M_{\text{W}\times\text{D}}$, respectively. The sub-ring structures obtained in 8 different scenarios in both the Abilene and GEANT networks are analyzed. The results obtained for each scenario are reported in Table I.

As observed, using $M_{1\times\text{D}}$ to set the logical link cost leads, on average, to the formation of more sub-rings in both Abilene and GEANT, although the difference is smaller in the Abilene case given the small size of the network. In the case of GEANT, it can be noticed that using the threshold value θ_{HalfMax} results in sub-rings that are more balanced in terms of size, especially when $Model_{\text{W}\times\text{D}}$ is used. More precisely, when comparing the ratio between θ_{HalfMax} and θ_{Avg} in the case of $M_{\text{W}\times\text{D}}$, we obtained a value of 3 for GEANT, whereas it is equal to 1.15 for Abilene. As a result, the structure of the sub-rings obtained are strongly affected by the value of θ in the case of GEANT. In the case of $M_{1\times\text{D}}$, the ratios are 1.18 and 1.04 for GEANT and Abilene, respectively, and as shown in the table, the structure of sub-rings is less affected by the choice of the threshold value. In addition, it can be observed that the structure of the sub-rings is not significantly affected by the initial node selection criterion in all the cases.

D. Intermediate Entity Selection

An analysis of how the logical link cost C_{LL} and the threshold θ can influence the IE selection in each sub-ring is finally presented in Table II. Here, the highest C_{LL} is used as the initial node selection criterion. To compare the different scenarios, a metric, denoted Δ, is defined to represent the ratio of the maximum logical link cost between the different IE nodes to the value of the average ring cost obtained in the corresponding scenario in Table I.

TABLE II
INTERMEDIATE ENTITY SELECTION.

Cost C_{LL}	Threshold θ	Rings Size	Selected IE Nodes	Δ
GEANT Topology				
$M_{1\times D}$	θ_{Avg}	5,2,3,3,3,3,2	16,21,1,20,17,7,2	1.31
$M_{1\times D}$	$\theta_{HalfMax}$	5,2,2,3,2,5,2	10,16,21,1,20,17,2	1.28
$M_{W\times D}$	θ_{Avg}	2,5,7,5,2	21,17,1,7,20	0.12
$M_{W\times D}$	$\theta_{HalfMax}$	15,6	9,8	0.01
Abilene Topology				
$M_{1\times D}$	θ_{Avg}	3,2,2,2,2	2,6,11,7,10	2.75
$M_{1\times D}$	$\theta_{HalfMax}$	3,2,2,2,2	2,6,11,7,10	2.75
$M_{W\times D}$	θ_{Avg}	3,2,3,3	2,6,7,11	1.68
$M_{W\times D}$	$\theta_{HalfMax}$	3,2,3,3	2,6,7,11	1.68

The value of Δ increases with the number of sub-rings. A larger number of sub-rings means that more clusters were formed during the multiple rings construction process. As such, more "longer" logical links exist between the clusters. In addition, it can be observed that the value of Δ is higher in the case of the Abilene network, which shows that the total cost in the sub-rings is on average smaller than the cost between the different sub-rings.

E. Time Complexity Analysis

The complexity of the proposed construction algorithms is influenced by the number of nodes in the substrate. The complexity of the ring construction algorithm is $O(N^2)$ [18], where N is the number of substrate nodes. In the case of the multiple rings construction algorithm, the complexity depends both on the number of nodes N and the number of sub-rings r that are formed. Given the steps of the algorithm presented in section IV-B, the complexity is $O(\frac{N^3}{r^2} + N^2)$. Finally, the IE selection algorithm consists in determining and comparing, for all nodes in each sub-ring, the distance to every other node in the substrate, and as such, has a complexity of $O(r^2N^2)$. Given that the construction of the substrate is an offline process, the above computational complexities are acceptable for the size of traditional Internet Service Provider domains.

VII. Summary

In this paper, we propose a hybrid topology structure to organize the network edge nodes into a management substrate. We describe different methods to compute the hybrid model. We define several construction parameters to control the structure characteristics and we evaluate the influence of these parameters by applying the proposed methods to two real network topologies.

Acknowledgment

This research has received funding from the Seventh Framework Programme of the European Commission, through the Flamingo Network of Excellence (318488) project.

REFERENCES

[1] M. Charalambides, G. Pavlou, P. Flegkas, N. Wang, and D. Tuncer, "Managing the future Internet through intelligent in-network substrates," *Network, IEEE*, vol. 25, no. 6, pp. 34–40, 2011.

[2] D. Tuncer, M. Charalambides, G. Pavlou, and N. Wang, "DACoRM: A coordinated, decentralized and adaptive network resource management scheme," in *the proceedings of the IEEE Network Operations and Management Symposium (NOMS'12)*, 2012, pp. 417–425.

[3] M. Charalambides, D. Tuncer, L. Mamatas, and G. Pavlou, "Energy-aware adaptive network resource management," in *the proceedings of the IFIP/IEEE International Symposium on Integrated Network Management (IM'13)*, 2013, pp. 369–377.

[4] D. Tuncer, M. Charalambides, R. Landa, and G. Pavlou, "More Control Over Network Resources: An ISP Caching Perspective," in *the proceedings of the 9th IEEE/IFIP International Conference on Network and Service Management (CNSM'13)*, 2013.

[5] S. White, J. Hanson, I. Whalley, D. Chess, and J. Kephart, "An architectural approach to autonomic computing," in *the proceedings of the International Conference on Autonomic Computing (ICAC'04)*, 2004, pp. 2–9.

[6] N. Samaan, "Achieving self-management in a distributed system of autonomic but social entities," in *Modelling Autonomic Communications Environments*, ser. Lecture Notes in Computer Science, S. Meer, M. Burgess, and S. Denazis, Eds., 2008, vol. 5276, pp. 90–101.

[7] N. Tcholtchev, M. Grajzer, and B. Vidalenc, "Towards a unified architecture for resilience, survivability and autonomic fault-management for self-managing networks," in *the proceedings of the 2009 International Conference on Service-Oriented Computing (ICSOC/ServiceWave'09)*, 2009, pp. 335–344.

[8] E. Lavinal, T. Desprats, and Y. Raynaud, "A generic multi-agent conceptual framework towards self-management," in *the proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium, (NOMS'06)*, April 2006, pp. 394 –403.

[9] D. Gracanin, S. Bohner, and M. Hinchey, "Towards a model-driven architecture for autonomic systems," in *the proceedings of the 11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems*, 2004.

[10] M. Jelasity, A. Montresor, and O. Babaoglu, "Gossip-based aggregation in large dynamic networks," *ACM Transactions on Computer Systems*, vol. 23, no. 3, pp. 219–252, Aug. 2005.

[11] F. Wuhib, R. Stadler, and M. Spreitzer, "Gossip-based resource management for cloud environments," in *the proceedings of the 2010 International Conference on Network and Service Management (CNSM'10)*, 2010, pp. 1–8.

[12] R. Makhloufi, G. Doyen, G. Bonnet, and D. Gaiti, "Situated vs. global aggregation schemes for autonomous management systems," in *the proceedings of the 4th IFIP/IEEE Workshop on Distributed Autonomous Network Management Systems*, 2011, pp. 1135–1139.

[13] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *Communications Surveys Tutorials, IEEE*, vol. 7, no. 2, pp. 72 – 93, quarter 2005.

[14] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: search methods," *Computer Networks*, vol. 50, pp. 3485–3521, December 2006.

[15] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, "Providing public intradomain traffic matrices to the research community," in *the proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM'06)*, vol. 36, January 2006, pp. 83–86.

[16] J. Postel, "Transmission Control Protocol," RFC 793 (Standard), Internet Engineering Task Force, Sep. 1981, updated by RFCs 1122, 3168, 6093. [Online]. Available: http://www.ietf.org/rfc/rfc793.txt

[17] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1982.

[18] Y. Crama, A. Kolen, and E. Pesch, "Local search in combinatorial optimization," in *Artificial Neural Networks*, ser. Lecture Notes in Computer Science, P. Braspenning, F. Thuijsman, and A. Weijters, Eds. Springer Berlin Heidelberg, 1995, vol. 931, pp. 157–174.

[19] "The Abilene topology and traffic matrices dataset," 2004, http://www.cs.utexas.edu/~yzhang/research/AbileneTM/.

[20] "The GEANT topology," 2004, http://www.dante.net/server/show/nav.007009007.

[21] "GNU Linear Programming Kit (GLPK)," http://www.gnu.org/software/glpk/.