

Contributions to Efficient Resource Management in Virtual Networks

Rashid Mijumbi, Juan-Luis Gorricho, Joan Serrat

▶ **To cite this version:**

Rashid Mijumbi, Juan-Luis Gorricho, Joan Serrat. Contributions to Efficient Resource Management in Virtual Networks. 8th IFIP International Conference on Autonomous Infrastructure, Management and Security (AIMS), Jun 2014, Brno, Czech Republic. pp.47-51, 10.1007/978-3-662-43862-6_5 . hal-01401288

HAL Id: hal-01401288

https://inria.hal.science/hal-01401288

Submitted on 23 Nov 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Contributions to Efficient Resource Management in Virtual Networks

Rashid Mijumbi, Juan-Luis Gorricho, and Joan Serrat

Universitat Politècnica de Catalunya,
08034 Barcelona, Spain
rashid@tsc.upc.edu, juanluis@entel.upc.edu, serrat@tsc.upc.edu

Abstract. Network virtualisation is a promising technique for a better future Internet by allowing for network resource sharing. However, resource sharing requires that virtual nodes and links be embedded onto substrate nodes and links (virtual network embedding), and thereafter the allocated resources dynamically managed throughout the lifetime of the virtual network (dynamic resource allocation). Since the constrained virtual network embedding problem is NP–Hard, many existing approaches are not only static, but also make simplifying assumptions, most of which would not apply in practical environments. This PhD research proposes improvements to both virtual network embedding and dynamic resource allocation. The objective is to achieve an efficient utilisation of physical network resources. To this end, we propose a path generation-based approach for a one-shot, unsplittable flow virtual network embedding, and a reinforcement learning-based dynamic allocation of substrate network resources.

Keywords: Autonomic and self-management, Future Internet, Virtual Networks, Virtual Network Embedding, Dynamic Resource Allocation.

1 Introduction

Network Virtualisation [1] promises to allow infrastructure providers (InP) who own substrate networks (SN) to lease out chunks of their physical resources to service providers (SP) who use these resources to create virtual networks (VN), which can be used to provide end-to-end services to final users. A VN is composed of virtual links and nodes, which have to be embedded onto SN links and nodes respectively. One problem encountered in setting up a virtual network is the need for efficient management of SN resources. This problem can normally be divided into two steps; the first of which performs the embedding of virtual nodes and links onto substrate nodes and links respectively−also known as virtual network embedding (VNE)−and the second one involves dynamic resource allocation (DRA) throughout the lifetime of VNs, aimed at efficient resource utilisation.

VNE with constraints on virtual nodes and links (e.g. topology, node capacity and link bandwidth) reduces to the multi-way separator problem, which is known to be NP-Hard [2]. Therefore, several existing solutions that propose one-shot

embedding such as [3] assume that all VN resource requirements are known a priori (offline case) and that the SN has unbounded resources, while others reduce the complexity further by assuming that the SN supports splitting of flows [4], and perform node and link embedding in two separate steps [5]. Even proposals that coordinate the two steps such as [2] still perform two separate embedding steps. Carrying out node and link embedding in two separate steps can lead to blocking or rejecting of resource requests at the link mapping stage and hence a sub-optimal substrate resource utilisation, which would negatively impact the revenue of InPs. The first part of this research (Section 2.1) proposes a *near optimal* solution to the unsplittable flow, one-shot VNE problem. This is achieved by formulating a mathematical program [6] of the VNE problem, and applying path generation [7] to allow for a faster, efficient solution. The objective is not only to reduce the time complexity of one-shot VNE, but also achieve comparatively good performance compared to two-step embedding proposals.

In addition, many current approaches [2]–[4] are static approaches (stop at the VNE step) without any considerations for possibilities of adjustments to the initial embedding, while those that propose dynamic solutions [8]–[11] do allocate a fixed amount of node and link resources to the VNs through out their life time. Overall, the number of dynamic solutions to VNE is limited [1]. Since network loading due to user traffic varies with time, allocating a fixed amount of resources based on peak loading could lead to an inefficient utilisation of overall SN resources, whereby, during periods when some virtual nodes and/or links are lightly loaded, substrate resources are still reserved for them, while possibly rejecting new requests for such resources. Once again, this would lead to inefficient resource utilisation and hence have a negative impact on the revenue of InPs, and could hinder the practical advancement of network virtualisation. Our second proposal (Section 2.2) follows a successful VNE, and involves the use of a decentralised multi-agent system (MAS) [12] which uses reinforcement learning (RL) [13] for life-cycle dynamic allocation of resources to all embedded VNs. The goal is to achieve efficient utilisation of overall SN resources, while ensuring that that the quality of service parameters such as packet drop rate (for virtual nodes) and delay (for virtual links) for the VNs are not affected.

2 Proposed Approaches: PaGe–VNE and RL–DRA

2.1 Path Generation-based VNE (PaGe–VNE)

PaGe–VNE uses two mathematical programs. One is a path-based formulation of the one-shot VNE problem, also known as the primal problem, while the other is its corresponding dual problem. For given instances of the problem, both the primal and dual problems have approximately the same solution value. As shown in Fig. 1, for a given VN request, PaGe–VNE begins by obtaining an *initial solution* (composed of paths P_1 in an augmented substrate network [2]) to the primal problem using a VNE approach that performs node and link mapping in two *coordinated* stages. The coordination involves solving a mathematical program of the node mapping problem, which is biased by the virtual link demands

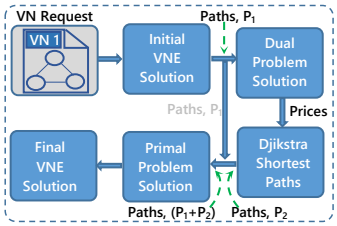


Fig. 1. PaGe–VNE

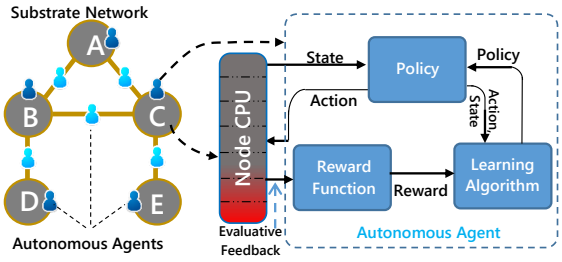


Fig. 2. RL–DRA

and substrate link capacities, and then the use *Dijkstra's* shortest path algorithm [14] for the link mapping stage. Since the initial solution involves node and link mapping in two separate stages, its computation takes much less time. However, the quality of the initial solution is not as good as would be obtained from performing a one-shot mapping. Therefore, PaGe–VNE's next step refines the initial solution, by using it as an input into the *dual problem*. This results into *prices* for the substrate network links and nodes. For every virtual link, these prices are then used to determine a *shortest path* from each of the possible substrate nodes at each of its ends using *Dijkstra's algorithm*. This set of paths, P_2–together with those obtained in the initial solution–are finally used to solve the primal problem and obtain a *final VNE solution*.

2.2 Reinforcement Learning-based DRA (RL–DRA)

After VNE, we dynamically and opportunistically allocate resources to virtual nodes and links depending on the perceived need for them by virtual nodes and links. The opportunistic use of resources involves carefully taking advantage of unused virtual node and link resources to ensure that VN requests are not rejected when resources reserved to already embedded requests are idle. To this end, we use a distributed and dynamic approach (RL-DRA) shown in Fig. 2 to allocate resources to virtual nodes and links using RL. In RL-DRA, each substrate node or link is represented by an autonomous agent (AA). The node AAs manage node capacities while the link AAs manage link bandwidths. These AAs use a RL-based evaluative feedback mechanism to dynamically adjust the resources allocated to virtual nodes and links, ensuring that resources are not left under-utilised, and that enough resources are available to meet VN QoS needs. Specifically, at periodic intervals, each AA perceives the current level of utilisation of resources (*state*), and uses its *policy* to take an *action* (hence changing resource allocations). The AA then receives feedback which is an evaluation of the desirability of its action. This is achieved by defining a *reward function* which takes into account how efficient the AA's action lead to better resource utilisation and its effect on VN QoS, to calculate a scalar *reward* that is sent back to the agent. The AA then uses a *learning algorithm* (such as Q-Learning) to adjust its policy so as to maximise the reward it achieves in the long run.

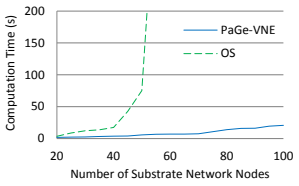


Fig. 3. Computation Time

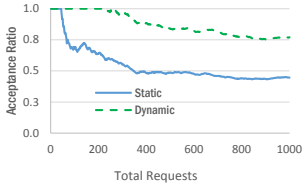


Fig. 4. Acceptance Ratio

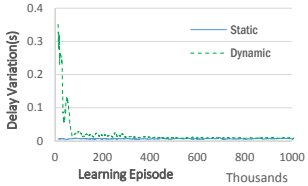


Fig. 5. Delay Variation

2.3 Obtained Results

Figs. 3–5 show *some* of the results from our proposals. In Fig. 3, PaGe–VNE has a significant computation time saving compared with a solution that determines an optimal solution (OS) of VNE while achieving almost the same VNE effciency (result not shown due to space restrictions). Fig. 4 confirms that RL-DRA leads to better substrate resource utilisation by accepting more VN requests than a static solution. Morever, as shown in 5, the better performance in RL-DRA is not at the expense of VN QoS, as the packet delay variation of the dynamic approach converges to that of the static one after the agent has learnt. We have written two papers [15], [16] as a result of these proposals. [15] is still under review while [16] has been accepted.

3 Conclusion and Future Work

This research is aimed at contributing to better resource management in network virtualisation. We have proposed a one-shot unsplittable flow VNE that improves solution computation time, and a distributed and dynamic approach that leads to better resource utilisation. However, some of the mathematical formulations used in PaGe–VNE can still become intractable for bigger problem instances. While a significant improvement in computation time is achieved compared to the optimal solution, and although in practice there are *high performance* tools [17] for efficiently solving such problems, more work can be done, for instance, by seeking a relaxation to the program which permits to solve it in polynomial time. In addition, implementing RL–DRA in real networks and in a multi-domain VN environment may raise more challenges since it may require a clear communication protocol (and a real management of this extra network load), negotiations and agreements between competing AAs that support inter-domain substrate paths. In future, we will study these issues and develop a prototype LAN where the AAs are based on a real AA development framework such as JADE [18].

Acknowledgments. This work is partly funded by Flamingo, a Network of Excellence project (318488) supported by the European Commission under its 7th FP, the EVANS project (PIRSES-GA-2010-269323), and TEC2012-38574-C02-02 from Ministerio de Economa y Competitividad.

References

1. Fischer, A., Botero, J., Beck, M., De Meer, H., Hesselbach X.: Virtual Network Embedding: A survey. IEEE Communications Surveys Tutorials, pp. 1–19 (2013)
2. Chowdhury, M., Rahman, M., Boutaba, R.: ViNEYard: Virtual Network Embedding Algorithms With Coordinated Node and Link Mapping. IEEE/ACM Transactions on Networking, vol. 20, no. 1, pp. 206–219 (2012)
3. Houidi, I., Louati, W., Zeghlache, D.: A Distributed Virtual Network Mapping Algorithm. IEEE International Conference on Communications, pp 5634–5640, (2008)
4. Yu, M., Yi, Y., Rexford, J., Chiang, M.: Rethinking virtual network embedding: Substrate support for path splitting and migration, ACM/SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 17–29, (2008)
5. Zhu, Y., Ammar, M.: Algorithms for Assigning Substrate Network Resources to Virtual Network Components. Proceedings of IEEE INFOCOM, pp. 1–12, (2006)
6. Bradley, H., Magnanti,: Applied Mathematical Programming. Addison-Wesley, (1977)
7. Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., Vance, P. H.: Branch-and-price: Column generation for solving huge integer programs. Operations Research, 46, pp 316–329, (1998)
8. Fan, J., Ammar, M.: Dynamic topology configuration in service overlay networks, A study of reconfiguration policies. Proc. IEEE INFOCOM, (2006)
9. Rahman, M. R., Boutaba, R.: SVNE: Survivable Virtual Network Embedding Algorithms for Network Virtualisation. IEEE Transactions on Network and Service Management, vol 10, no. 2, (2013)
10. Fajjari, I., Aitsaadi, N., Pujolle, G., Zimmermann, H.: VNR algorithm: A greedy approach for virtual networks reconfigurations. IEEE Global Telecommunications Conference, pp. 1–5, (2011)
11. Gang, S., Yu, H., Vishal, A., Lemin, L.: A cost efficient framework and algorithm for embedding dynamic virtual network requests. Future Generation Computer Systems 29, pp 1265–1277, (2013)
12. Russell, S. J., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, Englewood Cliffs, New Jersey, Third Edition, (2010)
13. Sutton, S., Barto, G.: Reinforcement Learning: An Introduction. Cambridge MA USA MIT Press, (1998)
14. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C.: Introduction to Algorithms. Third edition, Cambridge, MIT Press. (2009)
15. Mijumbi, R., Serrat, J., Gorricho, J. L., Boutaba, R.: Path Generation-based Virtual Network Embedding. IEEE/ACM Transactions on Networking. Submitted, Under Peer Review.
16. Mijumbi, R., Gorricho, J.L., Serrat, J., Claeys, M., De Turck, F., Latré, S.: Design and Evaluation of Learning Algorithms for Dynamic Resource Management in Virtual Networks. IEEE/IFIP Network Operations and Management Symposium (NOMS), Krakow, Poland, 2014. Accepted November 2013.
17. IBM ILOG CPLEX Optimizer. `http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/about/`. Visited January 08, 2014. (2014)
18. Bellifemine, F., Bergenti, F., Caire, G., Poggi, A.: Jade – A Java Agent Development Framework. Multi-Agent Programming. Springer US, Multiagent Systems, Artificial Societies, and Simulated Organizations, vol 15, pg. 125–147. (2005)