Network in the Air

Amir Varasteh
amir.varasteh@tum.de
Technical University of Munich

Wolfgang Kellerer
wolfgang.kellerer@tum.de
Technical University of Munich

Carmen Mas Machuca
cmas@tum.de
Technical University of Munich

ABSTRACT

Today, on-board passengers request Internet-based services, such as video streaming, Voice over IP, etc., at low cost. Many of those services have stringent QoS requirements, e.g., low end-to-end delay. To offer these services efficiently, some problems need to be solved jointly: placement of services (i.e., Virtual Machines (VMs)), at Datacenters (DCs) on the ground, assigning airplanes to services on DCs, and the routing from the flight to the associated VM considering the dynamic position of the flights over time. Further, dynamic VM migrations can be employed for guaranteeing the QoS requirements and/or improving resource utilization. In this work, we introduce and evaluate two heuristic solutions to jointly determine VM placement, routing, and migration decisions for flying airplanes with the objective of minimizing total operational costs. The first results indicate that while reducing the runtime from hours to seconds, the heuristics are able to achieve near-optimal solutions.

CCS CONCEPTS

• **Networks** → **Network management**; **Network dynamics**.

1 PROBLEM(S)

Internet-based services, demanded by passengers are usually deployed on VM instances located in distributed DCs owned/leased by the airlines. We face three challenges in this scenario, namely VM placement, routing, and VM migration. Firstly, the VM placement (i.e.,VM-to-DC mapping), providing the flight services, can affect the total operational costs of the service provider (e.g., airline) and the QoS levels offered to passengers [8, 10].

Secondly, it is important to decide which flight is assigned to which DC and how they should reach their respective DCs on the ground for the flight duration (i.e., routing). Two Air-To-Ground A2G alternatives with differing characteristics (bandwidth capacity, latency, and cost) [2] exist for flights to communicate to the ground (See Fig. 1): (i) Direct Air-To-Ground (DA2G), (ii) Satellite (LEO/GEO) links. DA2G establishes a direct connection between the flight and the ground if conditions allow (distance and the link capacity), while the satellite relays the received traffic from the flights towards the satellite gateway(s) in the ground core network.

Fig. 1 illustrates two snapshots of a Space-Air-Ground Integrated Network (SAGIN) at 5 PM and 7 PM, with two flights flying over Europe. It can be seen that at 5 PM, the VMs offering the services are located in the central DC. However, at 7 PM, a VM migration is

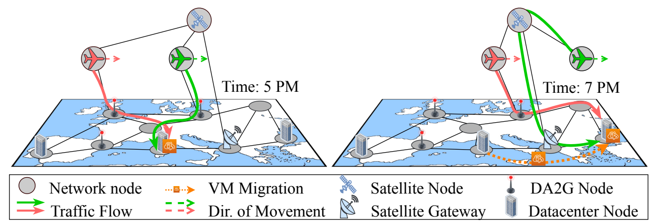


Figure 1: Two snapshots of a Space-Air-Ground Integrated Network, showing two flights, VM placement, routing, and a VM migration.

performed to keep the required QoS level and/or to reduce the total routing cost. Thus, determining dynamic VM migration decisions is yet another challenge that needs to be addressed in this scenario.

Considering the mobility of flights (different locations at each timeslot), and the heterogeneity of the SAGIN, five challenges need to be addressed: (i) What is the best VM-to-DC mapping for each service at each timeslot, considering VM capacity? (ii) At each timeslot, each airplane should be assigned to which DC to receive which service? (iii) Considering the QoS requirements of services, how to route the airplane traffic to the DC? (iv) When VM migrations are crucial, and between which DCs? (v) How to jointly answer these questions to minimize the total operational costs? The total operational cost is defined as the sum of VM deployment, routing, and VM migration costs.

2 MODELLING AND SOLUTIONS

We divide the time horizon $\mathcal{T}$ in to T timeslots denoted by t, forming a dynamic graph $G_t = (N_t, V_t)$. N_t is the set of graph nodes including flights, ground network nodes (core, DA2G, DC, and satellite gateway) and V_t is the set of links between them at each $t \in \mathcal{T}$. In fact, G_t can be updated at each $t \in \mathcal{T}$ (See Fig. 1), according to the flights' positions and their A2G connections to the ground network. We define the service request $r_t = (src, k, B, D)$ for each flight $\forall t \in \mathcal{T}$, where src, k, B, D are the flight ID (position), service type, bandwidth, and maximum end-to-end delay, respectively.

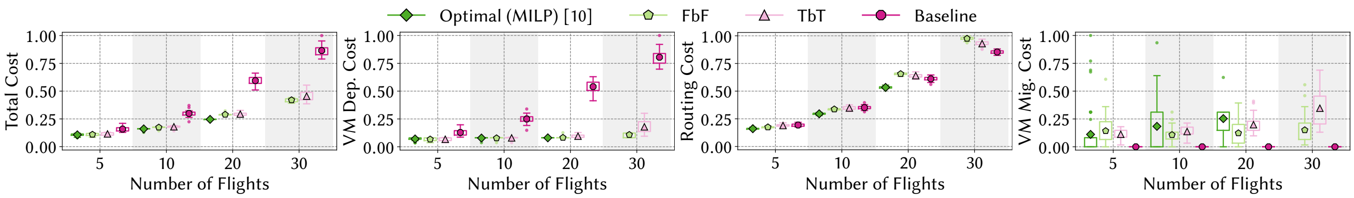
In our previous work [9, 10], we combined the Multi-period Capacitated Facility-Location (MPCFL) and Multi-Commodity Flow (MCF) problems and formulated it as a Mixed Integer Linear Program (MILP). It jointly determines the optimal VM placement, routing, and VM migrations for the whole $\mathcal{T}$.

Hardness: The Capacitated Facility Location (CFL) is known to be NP-Hard [1, 3, 6]. Therefore, the MPCFL, with an additional dimension (i.e., time) is also NP-Hard, since CFP is a special case of it. Additionally, the MCF with deadlines is proved to be NP-Hard [4]. Consequently, the combination of these two problems is even harder to solve.

Heuristics: Considering the problem complexity, to be able to solve it for practical-size instances, we introduce two heuristic solutions. In both of these heuristics, we decompose the problem into smaller sub-problems, but in different dimensions. Considering

CoNEXT '19 Companion, December 9–12, 2019, Orlando, FL, USA

ACM ISBN 978-1-4503-7006-6/19/12.
https://doi.org/10.1145/3360468.3366775



(a) Total Operational Costs **(b) VM Deployment Cost** **(c) Routing Cost** **(d) VM Migration Cost**

Figure 2: Comparison of different approaches in terms of average total, VM deployment, routing, and VM migration costs (normalized).

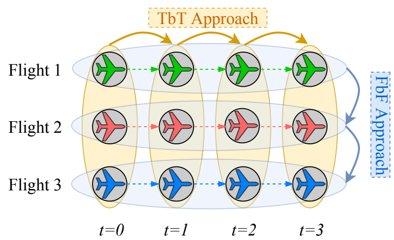


Figure 3: An example of FbF and TbT approaches for three flights and $T = 4$ timeslots (flight positions at the beginning of each $t \in T$.)

their dependencies, we then combine these sub-problems to achieve a solution for the main problem.

1. Flight-by-Flight (FbF): It solves the problem for each flight sequentially over $\mathcal{T}$, taking into account the VM placement and the routing of previous flights at $\mathcal{T}$ (See Fig. 3). Moving from t to $t+1$, VM migrations are performed towards meeting QoS requirements and/or total cost minimization.

2. Timeslot-by-Timeslot (TbT): This heuristic solves the problem per timeslot, that is, based on the position of all flights at a given timeslot $t \in \mathcal{T}$. Solving the problem for timeslot t, TbT considers the network status (VM-to-DC mapping, and routing) from timeslot $t-1$ (See Fig. 3).

Baseline: It works based on the FbF, but in a static and greedy fashion, without considering the VM placement and routing from previous flights. Moreover, it refrains itself from VM migration and creates a new VM instance when using the existing VM(s) leads to delay violation.

3 PRELIMINARY RESULTS

Setup: We simulate a European-based SAGIN using the Cost266 topology [7] as the ground core network, real-world DA2G locations [5], and flight data from *FlightRadar24* live air traffic from 9.11.2017. We consider flights with a duration of $\mathcal{T} = 3.5$ hours, and timeslot duration of 0.5 hour (i.e., $T = 7$). *Madrid*, *Hamburg*, and *Budapest* are chosen as the distributed DC locations. Also, three service types with different QoS requirements and migration costs have been defined, and randomly assigned to each flight for every $t \in \mathcal{T}$. Each scenario is run 30x for random set of flights and the mean value is reported. The detailed evaluation setup is available in our previous work [10]. We note that due to high runtime, we do not run MILP for more than 20 flights.

Results: While both FbF and TbT find a solution around 2000X faster than the MILP [10], as it is illustrated in Fig. 2a, they are able to achieve near-optimal results. The total operational cost of TbT increases for a higher number of flights, more than the FbF approach. This is mainly due to the VM cost, since TbT does not consider the whole $\mathcal{T}$; hence, compared to FbF, the possibility of VM sharing among flights is lower, leading to deploying more VMs (Fig.2b). However, since TbT considers all the flights on a single timeslot, the average routing cost is lower than FbF approach, since it does not stretch the routing paths over $\mathcal{T}$. The difference in routing cost increases by increasing the number of flights.

As it is depicted in Fig. 2c, the routing cost of the baseline approach is lower than the FbF and TbT, because it always tries to use a DC with the minimum routing delay (which can be translated to minimum #hops, hence, routing cost). Moreover, refusing from reusing the VMs assigned to other flights leads to having higher VM cost for the baseline, increasing the probability of finding a closer VM (i.e., lower routing cost).

Finally, Fig. 2d indicates that the optimal case triggers more VM migrations, since it considers the flights during the whole $\mathcal{T}$ and minimizes the total operational cost. It can be seen that TbT has lower VM migrations compared to FbF for lower number of flights and it increases with number of flights. In fact, FbF improves the VM sharing for the flights, leading to lower VM migration possibility.

4 DISCUSSION

The limitation of TbT approach is the consideration of the deployed VM and routing from only the previous timeslot. On the other hand, FbF is limited to one flight at each round of the algorithm. Therefore, the solution depends on the location of the previous flights. Also, a *Rolling Horizon* method might be helpful to solve the problem over time horizon $\mathcal{T}$. Thus, an improvement to TbT (FbF) can be the consideration of two or even more previous timeslots (flights) to reduce the *blindness* of the algorithm (using time window to control the size of the sub-problem).

Another improvement could be to combine TbT and FbF. For example, a problem can be solved sequentially firstly by TbT and then FbF considering the solution of the TbT. In this way, we can consider both timeslots and the flight positions at the same time. Additionally, clustering algorithms (based on the path and geographical location of flights), local search, and randomization algorithms can help to improve the solution quality.

Moreover, applying machine learning techniques such as deep reinforcement learning or message passing neural networks looks promising as a future work. We note that the proposed resource management algorithms can be applied to other scenarios with mobile users e.g., Mobile Edge Computing, Internet of Vehicles (Mobility prediction methods might be needed in these cases).

5 ACKNOWLEDGMENT

This work is funded by Germany's Federal Ministry of Education and Research under the ambit of project OptiCON (ID 16KIS0991).

REFERENCES

[1] Gérard Cornuéjols, Ranjani Sridharan, and Jean-Michel Thizy. 1991. A comparison of heuristics and relaxations for the capacitated plant location problem. *European journal of operational research* 50, 3 (1991), 280–297.

[2] Ergin Dinc, Michal Vondra, and Cicek Cavdar. 2017. Multi-user Beamforming and Ground Station Deployment for 5G Direct Air-to-Ground Communication. In *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 1–7.

[3] Dorit S Hochba. 1997. Approximation algorithms for np-hard problems. *ACM Sigact News* 28, 2 (1997), 40–52.

[4] Christian Vad Karsten, David Pisinger, Stefan Ropke, and Berit Dangaard Brouer. 2015. The time constrained multi-commodity network flow problem and its application to liner shipping network design. *Transportation Research Part E: Logistics and Transportation Review* 76 (2015), 122–138.

[5] Urs Mansmann. 2018. Fast Internet for aircraft in Europe, *Available:* https://goo.gl/KUFGGL.

[6] Gábor Nagy and Saïd Salhi. 2007. Location-routing: Issues, models and methods. *European journal of operational research* 177, 2 (2007), 649–672.

[7] SNDLib Topologies. 2019. *Available:* https://sndlib.zib.de.

[8] Amir Varasteh and Maziar Goudarzi. 2017. Server consolidation techniques in virtualized data centers: A survey. *IEEE Systems Journal* 11, 2 (2017), 772–783.

[9] A. Varasteh, S. Hofmann, N. Deric, A. Blenk, D. Schupke, W. Kellerer, and C. M. Machuca. 2019. Toward Optimal Mobility-Aware VM Placement and Routing in Space-Air-Ground Integrated Networks. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 1–2. https://doi.org/10.1109/INFCOMW.2019.8845123

[10] Amir Varasteh, Sandra Hofmann, Nemanja Deric, Mu He, Dominic Schupke, Wolfgang Kellerer, and Carmen Mas Machuca. 2019. Mobility-Aware Joint Service Placement and Routing in Space-Air-Ground Integrated Networks. In *International Conference on Communications (ICC)*. IEEE.