Supplemental Material of "Virtual Network Embedding with Opportunistic Resource Sharing"

Sheng Zhang, *Student Member, IEEE,* Zhuzhong Qian, *Member, IEEE,* Jie Wu, *Fellow, IEEE,* Sanglu Lu, *Member, IEEE,* and Leah Epstein

1 The Win-Win Situation of Opportunistic Resource Sharing

In this section, we aim to provide some intuitive insights on how opportunistic resource sharing can lead to a win-win situation—service providers' costs are lowered, while infrastructure providers' revenues increase, as well.

In a network virtualization environment [1], service providers lease substrate resources from infrastructure providers to deploy virtual networks and offer value-added services to end users. As we mentioned earlier, service providers often overbook substrate resources to cater for potentially high requirements, thus, service providers waste part of the purchased resources when there are only normal resource requirements. On the other hand, the amount of substrate resources that belong to an infrastructure provider is limited and constant over a relatively long time. To maximize its revenue, an infrastructure provider should make efficient use of the precious substrate resources, and accept as many virtual network requests as possible.

Consider an infrastructure provider $InP1$ that has a substrate link with a bandwidth capacity equal to 20 slots, and there are three service providers, $SP1$, $SP2$, and $SP3$. Each of the service providers wants to lease 8 slots in the substrate link for its own virtual link. Without opportunistic resource sharing, it is clear that $InP1$ can only accept two requests, since $8 \times 3 = 24 > 20$. If $InP1$ charges 1 dollar for 1 slot per hour, then the cost of an accepted request is 8 dollars/h, and $InP1$ can get a revenue of 16 dollars/h.

Next, let us apply opportunistic resource sharing to this example, and see whether there is any change in the cost and the revenue. After profiling experimentations, the service providers find that the 8 slots requirement is composed of a basic sub-requirement of 6 slots, and a variable sub-requirement of 2 slots, which are needed with probability 0.3. If this is the case, $InP1$ can accept all of the three requests by means of resource sharing. $InP1$ assigns 18 dedicated slots to the basic sub-requirements, and lets the variable sub-requirements share the remaining 2 slots. The collision probability can be easily obtained by:

$$1 - 0.8 \times 0.8 \times 0.8 - 3 \times 0.8 \times 0.8 \times 0.2 = 0.104$$

Since there are collisions for the variable sub-requirements, $InP1$ decreases the corresponding charge, e.g., 0.1 dollar for 1 slot per hour. Therefore, the cost of an accepted request is $(6 + 0.1 + 0.1) = 6.2$ dollars/h, which is lower than the previous charge, and the revenue of $InP1$ is $(6 + 0.1 + 0.1) \times 3 = 18.6$ dollars/h, which is much higher than the previous one.

In this example, we see that opportunistic resource sharing enables better utilization of physical resources, and hence, increases the revenues of InPs, and decreases the rents of SPs. We believe that opportunistic resource sharing can benefit all parties through reasonable pricing. We will not discuss how to set prices in this paper, as it is out of this paper's scope and deserves separate studies.

2 Proof of Theorem 1

Proof: Consider the following instance of the 3-partition problem [2]. There are positive integers B and m, and $3m$ positive integers a_i for $1 \le i \le 3m$ (called items), such that $B/4 < a_i < B/2$ for all i. The question is whether the items can be partitioned into m subsets, each of sum B (clearly, if this is possible, then each subset must contain three items).

Given an instance of the 3-partition problem, we construct an instance of the TSA problem as follows. Let $0 < p < 1$ (the reduction works for any such p), and $q = 1 - p$ (so $0 < q < 1$, as well). Let $N = (\lceil 10B/(pq) \rceil)^2$ (so $N \ge 100B^2/(p^2q^2)$). For $1 \le i \le 3m$, let $\varepsilon_i = a_i/N$, $p_i = p + \varepsilon_i$, $q_i = q - \varepsilon_i = 1 - p_i$. There are $3m$ variable sub-requirements from virtual links e_1, e_2, ..., and e_{3m}, where each e_i requires 1 time slot with probability p_i. We have:

$$0 < p < p_i < p + B/(2N) \le p + p^2q^2/(200B) < p + q = 1$$

and thus, since $q_i = 1 - p_i$, $0 < q_i < 1$.

We define the upper bound on the collision probability, i.e., p_{th}, for this instance by:

$$p_{th} = 1 - 3q^2 + 2q^3 + 2pqB/N + 10B^2/N^2$$

This value is indeed in $(0, 1)$, because:

$$\begin{aligned} & 3q^2 - 2q^3 - 2pqB/N - 10B^2/N^2 \\ & = q^2 + 2q^2 - 2q^3 - 2pqB/N - 10B^2/N^2 \\ & > q(q - 2pB/N - p^2q/(10N)) \\ & > q(q - 2B/N - q/10) \\ & > q(q - p^2q^2/(50B) - q/10) \\ & > q(q - q/50 - q/10) > 0 \end{aligned}$$

- *S. Zhang, Z.Z. Qian, and S.L. Lu are with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China. Z.Z. Qian is the corresponding author.*
  E-mail: zhangsheng@dislab.nju.edu.cn, {qzz,sanglu}@nju.edu.cn.
- *J. Wu is with the Computer and Information Sciences Department, Temple University, Philadelphia, PA 19122, USA.*
  E-mail: jiewu@temple.edu.
- *L. Epstein is with the Department of Mathematics, University of Haifa, Haifa 31905, Israel.*
  E-mail: lea@math.haifa.ac.il.

and

$$3q^2 - 2q^3 - 2pqB/N - 10B^2/N^2 < 3q^2 - 2q^3 \le 1$$

In the above, we have used that $3q^2 - 2q^3$ is monotonically non-decreasing $q \in (0, 1)$. The TSA problem is then translated into the following problem: whether it is possible to group the $3m$ sub-requirements into at most m sets, each with a collision probability of at most p_{th}.

We claim that it is possible to partition the items into m subsets of sum B if, and only if, it is possible to group the $3m$ sub-requirements into at most m sets, each with a collision probability of at most p_{th}.

We first compute the collision probability of a subset of three sub-requirements e_j, e_k, e_t. The probability that less than two of them occur is:

$$q_j q_k q_t + p_j q_k q_t + q_j p_k q_t + q_j q_k p_t$$

We have

$$\begin{aligned} &q_j q_k q_t = (q - \varepsilon_j)(q - \varepsilon_k)(q - \varepsilon_t) \\ &= q^3 - (\varepsilon_j + \varepsilon_k + \varepsilon_t)q^2 + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)q - \varepsilon_j\varepsilon_k\varepsilon_t \end{aligned}$$

and

$$\begin{aligned} &p_j q_k q_t = (p + \varepsilon_j)(q - \varepsilon_k)(q - \varepsilon_t) = pq^2 \\ &+ \varepsilon_j q^2 - (\varepsilon_k + \varepsilon_t)pq - (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t)q + \varepsilon_k\varepsilon_t p + \varepsilon_j\varepsilon_k\varepsilon_t \end{aligned}$$

Therefore,

$$\begin{aligned} &q_j q_k q_t + p_j q_k q_t + q_j p_k q_t + q_j q_k p_t = q^3 + 3pq^2 \\ &- 2(\varepsilon_j + \varepsilon_k + \varepsilon_t)pq + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t \end{aligned}$$

Then, the collision probability is 1 minus this number.

Next, we compute the lower bound on the collision probability of a set of at least four sub-requirements. Clearly, the probability that at least two sub-requirements happen is no less than the probability that at least two sub-requirements out of a subset of four arbitrarily chosen sub-requirements out of the set happen. Consider four sub-requirements, e_j, e_k, e_t, e_d. The probability that at most one of them occurs is:

$$q_j q_k q_t q_d + p_j q_k q_t q_d + q_j p_k q_t q_d + q_j q_k p_t q_d + q_j q_k q_t p_d$$

Since we have $q_j q_k q_t q_d < q^4$, and

$$p_j q_k q_t q_d < p_j q^3 < (p + B/(2N))q^3 = pq^3 + Bq^3/(2N)$$

Thus, the probability that at most one sub-requirement out of the four happens is less than

$$q^4 + 4pq^3 + 2Bq^3/N = 4q^3 - 3q^4 + 2Bq^3/N$$

We then show $4q^3 - 3q^4 + 2Bq^3/N \le 1 - p_{th}$, and thus we find that any set of at least four sub-requirements has a collision probability strictly above p_{th}.

Indeed $4q^3 - 3q^4 + 2Bq^3/N \le 3q^2 - 2q^3 - 2pqB/N - 10B^2/N^2$ holds, since this is equivalent to $2Bq^3/N + 2pqB/N + 10B^2/N^2 \le 3q^2(q-1)^2 = 3q^2p^2$, which holds since:

$$\begin{aligned} &2Bq^3/N + 2pqB/N + 10B^2/N^2 \\ &\le 2Bq^2/N + 2pqB/N + p^2q^2/(10N) \\ &\le 2Bq/N + Bq/N < p^2q^3/(30B) \\ &< p^2q^2 \end{aligned}$$

Therefore, a necessary condition for a valid partition of sub-requirements into m subsets is that every subset has exactly three sub-requirements. We then show that a subset of sub-requirements, e_j, e_k, e_t, can be grouped together if, and only if, $a_j + a_k + a_t \le B$. This will prove the correctness of the reduction.

We have $q^3 + 3pq^2 - 2(\varepsilon_j + \varepsilon_k + \varepsilon_t)pq + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t - (1 - p_{th}) = q^3 + 3pq^2 - 2(\varepsilon_j + \varepsilon_k + \varepsilon_t)pq + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t - 3q^2 + 2q^3 + 2pqB/N + 10B^2/N^2 = 2pqB/N - 2(\varepsilon_j + \varepsilon_k + \varepsilon_t)pq + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t + 10B^2/N^2$. We would like to show that this difference is non-negative if, and only if,

$$a_j + a_k + a_t \le B$$

which is equivalent to $\varepsilon_j + \varepsilon_k + \varepsilon_t \le B/N$.

We already have

$$\begin{aligned} &(\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t \\ &< 3(B/(2N))^2 + 2(B/(2N))^3 \le 5(B/(2N))^2 \end{aligned}$$

and

$$(\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) + 2\varepsilon_j\varepsilon_k\varepsilon_t > -3(B/2N)^2$$

Therefore, if $a_j + a_k + a_t \le B$, the difference is at least:

$$\begin{aligned} &2pqB/N - 2Bpq/N + (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) \\ &+ 2\varepsilon_j\varepsilon_k\varepsilon_t + 10B^2/N^2 \\ &> 10B^2/N^2 - 3(B/(2N))^2 > 0 \end{aligned}$$

and otherwise (that is, if $a_j + a_k + a_t \ge B + 1$, i.e., $\varepsilon_j + \varepsilon_k + \varepsilon_t \ge B/N + 1/N$), the difference is at most:

$$\begin{aligned} &2pqB/N - 2(B+1)pq/N + 10B^2/N^2 + 2\varepsilon_j\varepsilon_k\varepsilon_t \\ &+ (\varepsilon_j\varepsilon_k + \varepsilon_j\varepsilon_t + \varepsilon_k\varepsilon_t)(p - q) \\ &< 2pqB/N - 2Bpq/N - 2pq/N + 5(B/(2N))^2 + 10B^2/N^2 \\ &< 12B^2/N^2 - 2pq/N \\ &= (12B^2/N - 2pq)/N \\ &\le (12p^2q^2/100 - 2pq)/N < 0 \end{aligned}$$

The theorem follows immediately. □

3 Proof of Theorem 2

Proof: We first construct two special scenarios of our problem, then derive the approximation ratio bound.

Case I: we use a particular variable sub-flow, which requires d_{min} time slots and occurs with probability p_{min}, to replace all of the variable sub-flows. Then, the maximal allowable number vol_I of sub-flows that a substrate slot can be assigned to in this case is determined by:

$$1 - (1 - p_{min})^{vol_I} - vol_I \cdot p_{min} \cdot (1 - p_{min})^{vol_I - 1} = p_{th}$$

And the number of substrate slots required by all of the n sub-flows is $S_I = (n \cdot d_{min})/vol_I$.

Case II: we use another particular variable sub-flow, which requires d_{max} time slots and occurs with probability p_{max}, to replace all of the variable sub-flows. Then, the maximal allowable number vol_{II} of sub-flows that a substrate slot can be assigned to in this case is determined by:

$$1 - (1 - p_{max})^{vol_{II}} - vol_{II} \cdot p_{max} \cdot (1 - p_{max})^{vol_{II} - 1} = p_{th}$$

Similarly, the number of substrate slots required by all of the n sub-flows is $S_{II} = (n \cdot d_{max})/vol_{II}$.

It is straightforward to see that:

$$0 < S_I \le S_{opt} \le S_{cff} \le S_{II}$$

then:
$$\frac{S_{cff}}{S_{opt}} \leq \frac{S_{II}}{S_I} = \frac{d_{max} \cdot vol_I}{d_{min} \cdot vol_{II}}$$

The theorem follows immediately. □

4 Proof of Theorem 3

Proof: Let $Y = \sum_{i \in D_j} X_i$ and $\mu = E[Y]$, we have:

$$\begin{aligned} Pr(D_j) &= Pr[Y > 1] \leq Pr[Y \geq 1] \\ &= Pr[Y \geq (1+\delta)\mu] \quad (\text{let } \delta = 1/\mu - 1 > 0) \\ &\leq (e^{\delta}/(1+\delta)^{1+\delta})^{\mu} \quad (\text{chernoff bound}) \\ &= \mu e^{1-\mu} \leq \mu_{th} e^{1-\mu_{th}} = p_{th} \end{aligned}$$

The theorem follows immediately. □

5 Proof of Theorem 4

Proof: Given an instance of bin packing, we construct a corresponding instance of ETSA by letting μ_{th} be the bin size, p_1, p_2, ..., p_n be the sizes of n items, respectively, and $v_i = 1$ for all i. In doing so, we reduce bin backing to a special case of ETSA, and prove ETSA to be NP-hard. It is also easy to see that ETSA is in NP; the theorem follows immediately. □

References

[1] N. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[2] M. Gary and D. Johnson, "Computers and intractability: A guide to the theory of np-completeness," *WH Freeman and Company, New York*, 1979.