Multi-provider Service Negotiation and Contracting in Network Virtualization

Fida-E Zaheer[1], Jin Xiao[1], and Raouf Boutaba[1,2]

[1]David R. Cheriton School of Computer Science, University of Waterloo, Canada
[2]Division of IT Convergence Engineering, POSTECH, Republic of Korea

***Abstract*—Network virtualization environment (VNE) affords great business flexibility to the customers and the providers as multiple providers can jointly support a customer's virtual network. Under the current network model, a group of Infrastructure Providers (InPs) peer with each other to provide a packaged deal. Such a business arrangement is not customer-driven, does not promote fair market competition and does not ensure cost minimization. Furthermore, the on-demand nature of virtual networks requires efficient and automated service negotiation and contracting. In this paper, we present V-Mart. To the InPs, V-Mart offers an environment to participate in a faithful and fair competition over the VN resources; and to the SPs, it offers a customer-driven virtual resource partitioning and contracting engine. V-Mart uses a two-stage Vickrey auction model that is strategy-proof, flexible to diverse InP pricing models, and functions over heterogenous multi-commodity market that characterizes the NVE. Through analysis and simulation we show the flexibility and effectiveness of V-Mart.**

I. Introduction

Network virtualization has gained considerable attention [1]–[3] as it allows for deploying diverse network protocols and technologies customized for specific networked services and applications. In a network virtualization environment (NVE), the basic entity is a virtual network (VN), which is a logical topology composed of virtual nodes and virtual links. The provisioning of a VN involves the mapping/embedding of virtual nodes onto physical ones and virtual links onto physical links or paths. Once provisioned, a VN has the semblance of an actual physical network. Here, the role of the traditional ISPs is divided into two: *infrastructure providers (InP)* and *service providers (SP)*. Infrastructure providers are responsible for deploying and managing the substrate networks, i.e, the underlying physical resources. Service providers are customers to the InPs, and can lease virtual resources, possibly from multiple InPs, to form their virtual networks to provide services to *end users*. The owner of a VN, i.e, a SP, is free to implement services by selecting custom packet formats, routing protocols, forwarding mechanisms, and other control and management protocols. Moreover, a SP can serve as an InP to another SP, and so forth. Therefore, the business relation of NVE can be very complex. Furthermore, the lifetime of a VN is transient compared to traditional networks, as VNs can be freely setup and torn down based on service/application demand.

Consequently new service and network management challenges arise in NVE. From the network management perspective, there is a need for effective virtual network embedding, which deals with the efficient mapping of virtual resources to underlay resources of an InP. This is currently an active area of research [4]–[6]. From the service management perspective, it is important to address the issue of service negotiation and contracting among multiple InPs. This is especially true for NVE, where a much greater degree of business flexibility exists in terms of which InPs a SP will contract with. Although it may be possible for a small size virtual network to be fully embedded in single InP, it is much less likely for large inter-continental VNs. Indeed, inter-continental VNs are often provisioned among multiple network providers today. With the proliferation of VNE in the near future, we can expect a rather large number of InPs in the market, ranging from traditional underlay network providers to new 3rd party virtual network providers. Under the current network operational model, a group of collaborative InPs can negotiate among themselves to jointly host such a VN, however such business arrangement is not customer-driven in that the VN partitioning (among InPs) does not involve the customer in the negotiation process. Furthermore, it does not exhibit fair market properties, as it lacks free market competition and does not ensure price minimization for the customer.

In this paper, we present *V-Mart*, an open market model and enabling framework for automated service negotiation and contracting in NVE. To the InPs, V-Mart fosters an open and fair competition environment through auctioning; and to the SPs, it offers a customer-driven virtual resource partitioning and contracting engine. With V-Mart, a SP can disseminate a *Request for Quotation (RFQ)* when it desires to set up a VN, and any willing InPs may participate in a two-stage Vickrey auctioning process. V-Mart offers the customer (i.e. SP) a partitioning heuristic that minimizes the cost of VN setup, based on fair market value of the VN embedding cost. The heuristic not only considers intra-InP price but also the SP's preference for resource co-location and the high cost of inter-InP communication. Furthermore, V-Mart does not impose a particular pricing model on the InPs, but on the contrary supports diverse InP pricing policies ranging from resource-wise pricing to full network package pricing. Although V-Mart is designed for the NVE context, it is flexible enough to be applied to other distributed multi-provider service environment such as Cloud computing and Service-Oriented Architecture (SOA) infrastructures.

The remainder of this paper is organized as follows. Section

978-1-4244-5367-2/10/$26.00 ©2010 IEEE

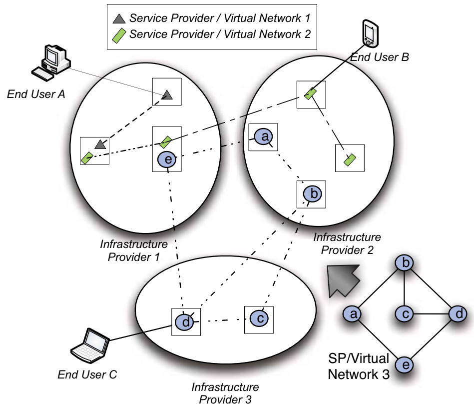

Fig. 1. The Business Model for an NVE.

II defines the business model for NVE. Section III provides an overview of V-Mart. In Section IV, we describe the problem of finding the optimal VN contract assignment. Section V describes the V-Mart auction model. In Section VI, we evaluate of V-Mart. Related works are presented in Section VII, and we conclude in Section VIII.

II. The Actors and Relations in NVE

In this section, we describe the actors in a NVE and their business relations. A detailed description can be found in [3].

A. Actors

There are three main actors in a NVE (Figure 1 illustrates an example interaction):

Infrastructure Provider (InP): Infrastructure providers deploy and manage the *physical network resources*. They divide the physical resources into multiple virtual ones and offer programmable interfaces to these virtual resources to their customers (service providers).

Service Provider (SP): Service providers lease *virtual resources* from multiple infrastructure providers to synthesize *virtual networks*.

End User: End users in the NVE are similar to their counterparts in current Internet.

B. Relations

Here we focus on SP-to-InP and InP-to-InP relations:

Vertical Relations are established between SPs and InPs, where the InP provides the necessary virtual resources to form the virtual network requested by the SP and generates revenue through provisioning, operation and maintenance of these virtual resources[1]. In addition to the traditional case where many customers are supported by a single provider (many-to-one relation), NVE also allows the flexibility of a single SP forming relations with multiple InPs for a single VN (one-to-many relation). Thus many-to-many vertical relation exists in NVE. In the context of this paper, we focus on one-to-many relations. We further note that an InP may in fact purchase resource from another InP, in this case the purchaser InP plays the role of a SP.

Horizontal Relations are formed between InPs, traditionally termed peering relations among network providers to facilitate end-to-end service delivery. In the context of NVE, a horizontal relation is formed between two InPs when each host a portion of the SP's virtual network. Under a market model, we consider two types of horizontal relations: *public relations* and *private relations*. Public relations are formed under the direction of a market mechanism (e.g. the result of an auction), where InPs find themselves co-hosting neighboring segments of the SP's VN. Private relations are formed among a group of InPs who decide to negotiate and cooperate in private. In a open market model such as the one V-Mart uses, public relations are the result of fair market competition while private relations are outside the design boundary of our framework. V-Mart considers each competing InP or a group of cooperative InPs as a single VN Bidder.

III. V-Mart Overview and Design

An inherent tussle exists between the SP and InP where the former wants to minimize its cost and the latter wants to maximize its revenue. In this context, it is highly beneficial for the SP to be able to divide its VN into multiple parts and assign to each InP in order to minimize cost. For example, assuming that all of the InPs in Figure 1 can exclusively provision virtual network 3. But the partitioning of $\{e\} \rightarrow InP1; \{a, b\} \rightarrow InP2; \{c, d\} \rightarrow InP3$ is cost minimizing. Our objective is to facilitate this negotiation process under a fair market model and to automate the VN partitioning and contracting process. Thus, our proposed framework enables the scenarios as prescribed in Figure 1.

A. Service Provider Requirements on a Virtual Network and Resources

Each virtual network has associated with it a set of requirements. This set includes configuration specifications and performance constraints [7]. Typical virtual node requirements include specification of minimum threshold values for CPU capacity, Queue size, availability, maximum value for mean time to repair (MTTR), etc. Requirements on virtual links include threshold values on bandwidth, delay, latency, packet loss, availability, etc. Performance requirements are often specified in terms of larger portions of the virtual network, i.e, a subgraph or the entire graph itself. Typical requirements include maximum values for end-to-end latency, end-to-end delay, average jitter, maximum jitter, and minimum value for network availability, etc.

B. V-Mart Design Objectives

V-Mart is designed with the following objectives in mind:

- **Flexible:** virtual networks are created dynamically on substrate resources from multiple InPs. A SP needs the flexibility to configure and operate its VN in a manner

[1]Throughout the rest of the paper, we use the term *virtual resources* to refer to both virtual nodes and virtual links

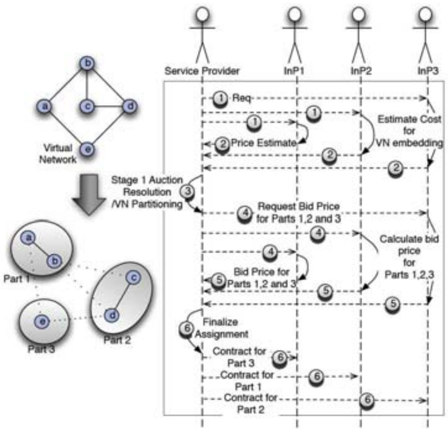

Fig. 2. The V-Mart Workflow

that best suits its purpose. Therefore, rather than choosing from a set of service offerings with pre-specified configurations and levels of performance from the InPs, the SP should have the freedom to determine the best contracting strategy among multiple InPs. This is supported by the V-Mart VN partitioning algorithm. At the same time, the InPs should have the flexibility to specify their resource pricing without the need to reveal their pricing models and private network information. This is supported by the V-Mart's two-stage auction model.

- **Fair market:** the negotiation among the SP and the InPs must be performed in such a way that the SP is protected from pricing manipulation of the InPs and the InPs are able to compete with each other in an open and truthful manner. To this end, we use a sealed one-sided Vickrey auction model (Section V) that is strategy-proof.
- **Automated:** the negotiation and contracting process is mostly automated in V-Mart, supporting quick on-demand VN setup.
- **Efficient:** the V-Mart framework must be efficient in its operation, both in terms of running time and the number of auctioning iterations. More specifically, the VN partitioning problem is NP-Hard and an efficient heuristic is necessary. Furthermore, some common auction models such as open one-sided bidding (e.g. eBay) is inefficient in the NVE context as it requires many iterations of bidder competition that requires the InPs to re-evaluate their embedding and pricing within each iteration.

C. Overview of V-Mart Operations

We briefly overview the V-Mart operations here, the technical detail on the VN partitioning algorithm is presented in Section IV and the auction model in Section V. Figure 2 illustrates a V-Mart workflow example.

Phase 1 - Request for Quotation (RFQ): the SP will formulate its VN request by sending out a RFQ to all interested VN Bidders (InPs). The RFQ includes the virtual topology and the requirements on the virtual resources. SP preference in the form of co-location constraints (defined in the next section) are also specified. This phase is step 1 in Figure 2.

Phase 2 - Resource Quote: each VN Bidder is expected to indicate the virtual resources it is willing to host and the corresponding hosting cost. This quotation is obtained with the following procedure. First, the VN Bidder performs embedding (mapping virtual resources to physical ones) of the VN. Second, the VN Bidder provides a price quote under the Vickrey model. The V-Mart framework influences how InPs can adopt their pricing strategies in this phase. We will discuss these specifics in Section V. Important features such as volume discount and package pricing are supported by V-Mart, and it is quite important for a VN Bidder to adopt its strategy to the market condition. This is step 2 in Figure 2.

Phase 3 - VN Partitioning: by the end of Phase 2, the SP obtains a set of price quotes for each virtual resource from the VN bidders. The SP now partitions the VN into multiple segments and attach them to specific VN Bidders. V-Mart provides a partitioning heuristic that performs this task automatically, aimed at both minimizing the total cost of the VN hosting and satisfying the SP's co-location constraints. This is step 3 in Figure 2.

Phase 4 - Final Offer: the list of segments obtained in Phase 3 is sent to all of the VN Bidders, as well as the winning VN Bidder of each segment and the winning Vickrey price. The VN Bidders make one final sealed bid that is upper bounded by the winning quote. The final winner of each segment is determined by the SP. Step 4 and 5 in Figure 2.

Phase 5 - Contracting: the SP contacts the winning VN Bidder of each segment and performs the final contracting and SLA generation. This phase is step 6 in Figure 2.

IV. VN PARTITIONING AMONG MULTIPLE PROVIDERS

Consider the case where there are K VN bidders competing for the virtual network, $G^V = (N^V, L^V)$, such that N^V and L^V are the set of virtual nodes and the set of virtual links respectively. Depending on its assignment to an InP, a virtual node can have different costs [2]. Therefore, each virtual node $n^V \epsilon N^V$ will have, associated with it a vector, $C^N(n^V) = \{c_k^N(n^V)|k = 1, ..., K\}$, where $c_k^N(n^V)$ is the cost for node n^V by bidder k. For the same reason, virtual links also have a vector of costs, where each element denotes the intra-domain (where both its end-points are hosted by the same VN bidder) link cost for a particular VN bidder. i.e, $C^L(l^V) = \{c_k^L(l^V)|k = 1, ..., K\}$, where $c_k^L(l^V)$ is the cost for link l^V by bidder k. In addition, each link also has associated with it an inter-domain cost, $C_{a,b}^I(l^V)$, which specifies the cost of provisioning it as an inter-domain link between bidders a and b, such that $a, b \epsilon K$ and $a \neq b$.

Therefore, the assignment of a VN among multiple providers is a variant of the k-cut problem, where the objective is to divide G^V into $P <= K$ subgraphs, $G_p^V = (N_p^V, L_p^V)$, where $N^V = \cup_{p \epsilon P} N_p^V$ and $L^V = (\cup_{p \epsilon P} L_p^V) \cup L_I^V$, where L_I^V

[2] In this section, we use the term *cost* to denote the price quote provided by a VN bidder

is the set of inter-InP links, such that the total cost $\mathbb{C}^{G^V}$, is minimized.

$$\mathbb{C}^{G^V} = \mathbb{C}^N + \mathbb{C}^L + \mathbb{C}^I$$

where $\mathbb{C}^N = \sum_{p=1}^{P} \sum_{n_p^V \epsilon N_p^V} c_p^N(\mathcal{M}_p^N(n_p^V))$,
$\mathbb{C}^L = \sum_{p=1}^{P} \sum_{l_p^V \epsilon L_p^V} c_p^L(\mathcal{M}_p^L(l_p^V))$,
$\mathbb{C}^I = \sum_{l_I^V \epsilon L_I^V} c_I^L(\mathcal{M}_I^L(l_I^V))$

$\mathcal{M}_p^N$, $\mathcal{M}_p^L$ and $\mathcal{M}_I^L$ are mapping functions such that $\mathcal{M}_p^N$: $N_p^V \to N^V$, $\mathcal{M}_p^L$: $L_p^V \to L^V$ and $\mathcal{M}_I^L$: $L_I^V \to L^V$

However, even when considering only the link weights, the k-cut problem for arbitrary k is NP-Complete [8]. V-Mart, therefore, uses a VN-Partitioning heuristic.

A. V-Mart Partitioning Heuristic

In this section we describe the VN Partitioning heuristic used by V-Mart to partition the virtual network topology among multiple VN bidders.

1) Co-Location Constraint: As a VN can be assigned to multiple, possibly competing, InPs, end-to-end performance is difficult to guarantee. However, a SP may have strict performance requirements on a group of resources. In these situations, a SP can specify that these virtual resources be assigned to a single VN bidder. This way guarantees can be achieved and violations can be resolved easily and quickly. Therefore, we introduce the concept of *Co-Location* constraints. Co-Location constraints on two nodes of the VN topology are expressed using a *Binary Co-Location Matrix*, Col, where $col(a, b) = \{0, 1 | a, b \epsilon N^V\}$ such that, if, $col(a, b) = 1$, then both a and b have to be assigned to the same InP. Note, that the co-location constraint is symmetric and transitive. Therefore, the co-location matrix divides the VN into a set of *islands*, which we call *V-Lets*, where each V-Let is composed of nodes which are to be assigned to the same bidder.

2) V-Let Graph: Each V-Let in the VN graph has to be assigned to the same partition. Therefore, the partitioning process, in its first step, forms a meta-graph, called the *V-Let Graph*, $G^M = (N^M, L^M)$, from the VN request graph.

3) Virtual Resource Costs: With K bidders participating in the auction, each V-let $n^M \epsilon N^M$ will have, associated with it, a K-dimensional cost of, $C^N(n^M) = \{c_k^N(n^M) | k = 1, ..., K\}$, where $c_k^N(n^M)$ is the cost quoted for n^M by bidder k. This value is set to infinity if InP k cannot or is not willing to provision V-Let n^M. Similarly, each meta link, $l^M \epsilon L^M$ has a intra-partition cost vector of, $C^L(l^M)$. A meta link can also be between two InP domain. Exact inter domain link costs are not readily available to the SP, as it involves two competing bidders. However, industry trends suggest that inter-domain communication cost is a magnitude higher than intra-domain. For example, Amazon EC2 [9] charges at most \$0.01 for one GB of intra-provider data transfers and a minimum \$0.10 for inter-provider. V-Mart assumes a constant inter-domain cost $\acute{C}^L$, per unit bandwidth requirement, valued at an order of magnitude higher than an average intra-domain link cost.

4) Initial Partition: V-Mart's VN partitioning heuristic starts with an initial partition and performs local improvements. We use two different initial partitions.

- **MinCost:** In this approach each V-Let is assigned greedily to the bidder with lowest quoted cost, i.e, V-Let n^M is assigned to partition k, where $c_k^N(n^M)$ is minimum for all $k <= K$.
- **MinCut:** Each bidder usually bids for more than one V-Let, i.e, a subgraph of the V-Let graph. The MinCut approach assigns all non overlapping subgraphs of the V-Let graph greedily to a bidder starting with the largest subgraph; ties are broken using the total quoted cost. This process is continued until no subgraph can be formed using the remaining V-Lets. The remaining V-Lets are greedily assigned using the MinCost approach.

Each approach has its advantages in particular scenarios. We explore these scenarios and provide a quantitative comparison of the two approaches in Section VI.

5) Local Improvements: Both the initial partition approaches assign V-Lets to bidders to minimize the V-Let provisioning cost, without considering the meta links. Local improvements are then made to minimize the cut size, i.e, the cost of inter domain links. However, as V-Lets are moved from their initial assignments, the cost of provisioning them increases. The VN partitioning process only moves a V-Let if the reduction in inter domain link cost is greater than the increase in the V-Let provisioning cost. V-Mart uses a *Gain* metric to decide a profitable re-location of V-Let, where, $Gain_{A,B}(a)$ = the gain in moving node a from bidder A to B = old cost - new cost = $\beta\times$ *Decrease in estimated meta link provisioning cost* - $\alpha\times$ *Increase in estimated island provisioning cost* = $\beta(\sum_{x \epsilon A} c_A^L(a, x) + \sum_{x \notin A} \acute{C}^L - \sum_{x \epsilon B} c_B^L(a, x) - \sum_{x \notin B} \acute{C}^L)$ - $\alpha(c_B^N(a) - c_A^N(a))$

Where $\acute{C}^L$ is the inter domain link cost, α is the weight factor for meta links and β is the weight factor for V-Lets.

The local improvements phase is an iterative process. In each iteration, all the V-Lets are individually considered and moved to a new partition if it produces a positive gain. The algorithm terminates when an iteration can provide no positive gain or after m iterations. Algorithm 1 shows the pseudocode for the local improvement phase.

V. The V-Mart Auction Model

Auction is an effective open negotiation mechanism for multiple competing buyers and sellers. It does not rely on price fixing by centralized authority and serves as a more fair alternative to provider-centric whole VN packaging. However, a correct auction model must be employed such that fair and faithful competition can be ensured and the resulting outcome is beneficial to both the buyers and the seller.

A. A Two-Stage Vickrey Auction Model

There exist many forms of auction models, such as the popular English auction with reservation, the Dutch auction, sealed high bid auction, Vickrey auction, etc. In general, a fair

Algorithm 1: Local Improvement / Refining V-Let Assignments

Form Initial Partition (MinCost or MinCut);
for $iteration \leftarrow 1$ **to** m **do**
  foreach $a \epsilon N^M$ **do**
    foreach *Partition* B*, such that* $a \not\epsilon B$ **do**
      $Gain(A, B) \leftarrow \beta(\sum_{x \epsilon A} c_A^L(a, x) + \sum_{x \notin A} \acute{C}^L - \sum_{x \epsilon B} c_B^L(a, x) - \sum_{x \notin B} \acute{C}^L)$ - $\alpha(c_B^N(a) - c_A^N(a))$
    end
    $maxGain \leftarrow max(Gain(A, B))$
    if $maxGain > 0$ **then**
      move a from A to B
    end
  end
  if *no V-Let is moved in last iteration* **then**
    return
  end
end

auction is *two-sided*, where the customers submit bids and the producers submit quotations, and a matching algorithm is run to produce the final result. When the customer or the producer cannot form an informed evaluation of the goods, *one-sided* auction is preferred. This is the case in NVE. The complexity involved in embedding and provisioning the virtual resource on a physical network is beyond the knowledge and expertise of a SP, who can only formulate a vague upper bound on the cost. It is then very important to have an auction model that ensures the InPs will offer prices proportional to the actual cost of hosting the virtual resources. The Vickrey auction model [10] is such a truthful mechanism. Under the Vickrey auction model, the VN Bidders will quote a price of hosting a virtual resource, and the lowest quote wins the auction, but the service is rendered at the price of the second lowest quote. The Vickrey model is strategy-proof, meaning that the only dominating strategy in this auction game is for each VN Bidder to quote a price proportional to the actual cost of hosting the service. The Vickrey auction model can be performed as either an open or sealed auction. The *open* auction has two issues. One, it is price minimizing with respect to large number of bidders. The dominating strategy is for the VN Bidder to quote a price exactly equal to cost, and thus gaining zero profit. This game is effective and strategy-proof, but does not offer fair market value to the InPs. Two, the number of auction iterations is large. In an attempt to optimize profit, each VN Bidder will descend its quote ϵ-small from the current winning quote during each iteration until it wins the bid or has a profit margin of zero. This is commonly known as the shilling effect. On the contrary, a *sealed* Vickrey auction is a single round auction that has all of the VN Bidders quote a price in secrecy. This model does not suffer from the price minimization issue due to the psychological effect of incomplete information.

Thus we arrive at the one-sided sealed Vickrey auction model for V-Mart. Each VN Bidder receives a RFQ from the SP and submits a per virtual resource price quote for each resource it is willing to host. We will discuss how this per resource price quote can be determined later in this section. The SP takes all the quotes for each virtual resource and modifies the quote of each VN bidder to the quote of the immediate following (in sequence) VN bidder. For example, for the quotes $\{A = 2, B = 3, C = 4\}$, the Vickrey quote is $\{A = 3, B = 4, C = 4\}$. the V-Mart model is strategy-proof. For brevity, we skip the proof here. We can see that a Bidder has no incentive to quote a price much higher than its cost, as it will only benefit its competitor. The revenue margin gained is the direct result of the cost differential among the Bidders for hosting the same virtual resource.

The result of this first stage Vickrey auction serves as the basis for our VN partitioning heuristic that minimizes cost. As the result, we obtain a set of VN segments with its associated total price and VN Bidder. In the second stage of our auction model, all the segment topologies, their total costs (termed *maximum reservation price*), and the identities of the associated VN Bidders (termed *owner of the insured bid*) are sent to all the VN Bidders in a single round sealed auction. Each VN Bidder is asked to provide a final price quote on each segment (if the VN Bidder is willing and/or able to do so), and this price quote is up bounded by the maximum reservation price. The SP takes all the bids and contracts each VN segment to the lowest VN Bidder of that segment at the quoted price. If the final price quote matches the maximum reservation price, the owner of the insured bid receives the contract. This stage is a one-sided sealed auction with maximum reservation.

Our V-Mart two-stage auction model is flexible in dealing with heterogenous correlated commodities, which the conventional auction models such as the Vickrey model is unable to. The VN partitioning heuristic accounts for the resource interdependence and the resulting price variation (i.e. the collateral cost of creating/removing inter-InP links when virtual resources are hosted in different VN Bidders). The strategy-proof first stage auction provides the necessary faithful cost information upon which a SP can effectively perform its cost minimization. The pricing models of a VN Bidder is supported by the second-stage auction, which we will discuss in the remainder of this section. The resulting contracting is then processed between the SP and the winning VN Bidders. We note some issues: 1) For the SP to be faithful, the first round bids should be secured either by audit via a trusted 3rd party, or by an open audit procedure at the end of the auction. 2) it is essential for the second-stage auction to have a maximum reservation price to ensure the validity of the first stage Vickrey auction. 3) We do not study long term strategy emergence here, as typically SP would not provision the same VN repeatedly. Furthermore, obtaining the strategy of a InP through observation over long term is not easy (and thus beneficial to the truthfulness of an auction) unless the physical resource topology and the embedding strategy of the InP is known, both of which are private information to the InPs.

B. The Pricing Model and Auction Strategy of a VN Bidder

Thus far, we have assumed that a VN Bidder is able to provide a per virtual resource price quote. Although there is great flexibility and advantages to per resource pricing model as implemented by a great majority of Cloud computing infrastructure (e.g. Amazon EC2, Google Apps, etc.), the pricing models of NVE are strongly influenced by the traditional network provider business model that operates quite differently from that of the application providers. In this subsection, we show two common pricing strategies: *volume discount* and *package pricing*. Both of these strategies are commonly used by sellers of multiple commodities.

Volume discount is a standard economic practice where the seller is willing to provide a discounted per unit pricing when a large enough volume of the commodity is purchased (e.g. wholesale). The discount is typically described as a discount function and reported to the SP. In the NVE context, it may be rather difficult to report this discount function to the SP. One reason is that the VN Bidder may not be willing to disclose its exact charging function to the buyer in order to guard against a competitor VN Bidder who can masquerade as a buyer and poach for its pricing model. Secondly, incorporating discount function in the VN partition heuristic increases the complexity of an already challenging problem. Instead, a VN Bidder's volume discount function can be applied in the second stage of the V-Mart auction model, where the VN has already been segmented and the exact resource count in each segment is public information. Hence in the first stage auction, the VN Bidder may quote a resource-wise price at non-discounted rate, while in the second stage auction quote a segment-wise price as modified by its discount function.

Package pricing is a common practice in heterogenous multi-commodity market, where a seller wishes to sell a package of commodities together at lower price (e.g. a whole dinning room set is cheaper than the sum of its individual parts). In the first stage of V-Mart auction, the VN Bidder would quote the price for a package of virtual resources and then map this package price to per resource price. This can be done by either computing the average price based on number of resources in the package, or by computing a weighted average of the resources based on the proportional cost of supporting each resource in the package. Although the latter method is preferred by V-Mart, it is understandable that a VN Bidder may not wish to disclose the cost of hosting any specific resource. In the second stage of V-Mart auction, the VN Bidder will quote the segment price based on its package price model. It is apparent that a mismatch in the VN Bidder's original package quote and the SP's segment partition will be problematic. We discuss how such mismatch is unlikely when the VN Bidder employs the right auction strategy.

We now discuss the aspect of strategy game play by the VN Bidders under different pricing models. To this end, the strategy move a VN Bidder makes in the first stage of V-Mart auction is critical as it strongly influences how the VN segments are generated by the SP. At first glance, this assertion appears to be contradictory to the concept of strategy-proof as we have stated: the one-sided sealed Vickrey auction model has a single dominate strategy which is to quote a price proportional to the cost. We now observe that both the volume discount and the package pricing models can only provide a cost estimate in the first stage of the auction. Or to put it another way, the exact cost of hosting cannot be known until after the VN partitioning.

First we examine the best strategy move of a VN Bidder under volume discount pricing. In the first round of auction, the VN Bidder can adopt a risk-averse stance or a risk-seeking stance. A risk-averse VN Bidder will estimate the cost of the resources at their non-discounted price and thus risk no chance of negative profit after the VN partitioning. A risk-seeking VN Bidder will estimate the cost of the resources already at a discounted price by assuming some of the final VN segments will contain at least the expected number of resources. Thus a negative cost could be incurred when this VN Bidder is the owner of an insured bid with smaller than expected segment size. On the other hand, a risk-seeking strategy is a best strategy move when the VN Bidder considers itself to be offering low discounted price in the market. Effectively, a risk-seeking VN Bidder can then corner the demand market especially when the other VN Bidders are risk-averse or does not provide equivalent price discounts. However, when all VN Bidders are Risk-seeking, we arrive at an inefficient system state where the sellers are selling at negative profit. Although such a state is cost minimizing for the SP, it is not fair market to the InPs.

A market that adopts package pricing is best represented by setting a low β/α ratio in the VN partitioning heuristic. Effectively, this produces high inter-InP link weight compared with node weight. Under high inter-InP link weight, the package of a VN Bidder will be preserved during the VN partitioning unless another VN Bidder is able to host majority of the same resources exclusively while offering a lower price quote. Thus mismatch between VN segment and a VN Bidder's package is not a significant problem. Therefore, the VN Bidder's best strategy is always to compute the per resource cost estimate based on the packaged price rather than based on non-packaged resource-wise cost estimate. This strategy provides the VN Bidder with the best chance of obtaining a VN segment matches its package.

VI. Performance Evaluation

In this section, we evaluate the performance of the VN partitioning algorithm and the relation of its parameters to market pricing models. We used flat and uniform random graphs generated by the GT-ITM tool [11] as the virtual request topologies, where each pair of nodes are randomly connected with a probability of 0.05. The quoted virtual resource cost provided by the bidders are uniformly distributed between 100 and 500. In each set of experiments conducted, we used 5 graphs of each size and took the average. Even though this setup may seem rather random, we believe the topology of VNE is much more flexible than current underlay topology and thus does not exhibit similar characteristics (e.g. power law property). Hence, we use random topology setup as a base study.

A comparison of our heuristic with an optimal exhaustive search is not computationally feasible. Therefore, we compared MinCost and MinCut with a *Random* solution. The

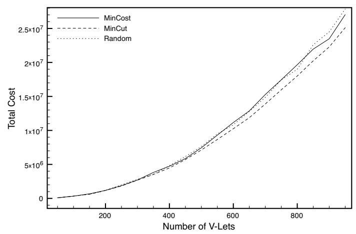

Fig. 3. Total Cost over Different Sizes of Graphs

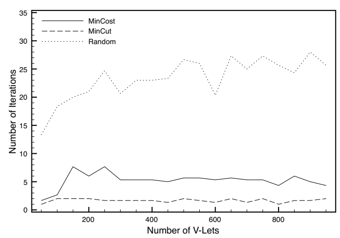

Fig. 4. Total Number of Iterations over Different Sizes of Graphs

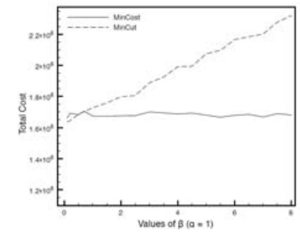

Fig. 5. Total Cost over Different Values of β/α

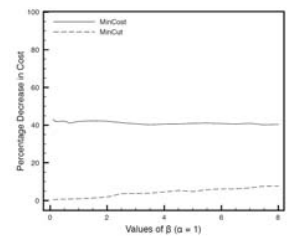

Fig. 6. Decrease in Total Cost from the Initial Configuration over Different Values of β/α

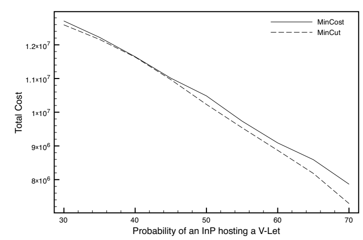

Fig. 7. Total Cost over Different Probabilities of an InP Desiring a V-Let

Random solution starts with a random initial partition and performs local improvements using the same objective as our heuristic. For each input, we ran the random solution four times and took the best result. We conducted three sets of experiments, to evaluate the performance of the heuristic, the impact of constants α and β, and the impact of InP provisioning probability. Experiment sets ii) and iii) are conducted using 500 node graphs.

i) *Performance of Heuristic:* Figure 3 shows the optimality of each respective starting configurations. We see that the total cost achieved starting from different configurations are comparable, with the MinCut performing the best ($\alpha = 5$ and $\beta = 1$). Figure 4 shows the total number of iterations for each starting configurations. It is apparent that random solution performs the worst, since it requires 4 runs. The performance difference of MinCost and MinCut is expected because of the particular α and β values used, as we will explain in experiment set ii.

ii) *Impact of α and β:* as the ratio of β/α increases, the performance of MinCut drastically decreases, as shown in Figure 5. Figure 6 depicts the percentage decrease in total cost from the initial partition. A lower percentage of decrease means a better initial configuration. These results are expected because, as β increases the V-Let cost factors more prominently in the overall cost, hence the decrease in cost by reducing the inter-domain cut size can no longer yield significant gain. This explains why MinCut performs better than MinCost in performance study i). The cross-over point is around 1.0. In practice, this ratio should be set according to the pricing model mix of the VN Bidders. In a market with high mix of package pricing bidders, the ratio should be much less than 1.0, while the converse is true when there is a high mix of resource-wise pricing models.

iii) *Impact of InP Provisioning Probability:* the InP provisioning probability is the probability of an InP willing to host a particular V-Let. When this probability value is high, each InP can host most or all parts of a VN. In these cases the MinCut algorithm produces lesser total costs than MinCost. However, as can be seen from Figure 7, MinCost produces better results when the percentage is low. Moreover, from Figure 8, it is evident that the initial solution for MinCost is closer to the final result when the probability is low. A high InP provisioning probability is a good representation of a pricing market where there is a high mix of InPs with volume discount pricing.

In summary, we observe that not only do the MinCut and MinCost configurations perform well compared with random initial configurations, but also offer significant performance gain. Furthermore, both the MinCut and MinCost configurations are needed as they perform best under different parameter configurations that are reflective of the various InP pricing model mix in the market.

VII. Related Work

Virtual network embedding problem [4]–[6], takes a resource and network management point of view, is complimentary to the problem addressed in this paper. These works

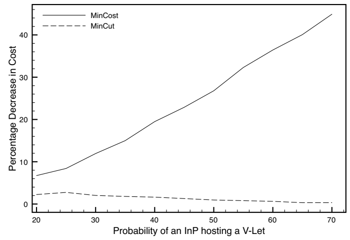

Fig. 8. Decrease in Total Cost from the Initial Configuration over Different Probabilities of an InP Desiring a V-Let

focus on the problem of mapping the virtual network resources (virtual nodes and virtual links) onto the underlying physical infrastructure in order to maximizing revenue for the InP and minimizing cost for the customer. They mostly address the problem from the InP's point of view, and although customer cost minimization is one of the objectives, without service negotiation, contracting and direct customer participation in the process, it is impossible to ensure fair market practices or minimum cost for the customers. To the best of our knowledge, there is no existing literature on the exact problem of designing an open market for service negotiation and contracting in a multi-provider virtual network environment.

Service management, SLA driven management, contracting and negotiations have been addressed in various multi-provider service environments including grid, utility computing and web services [12]–[14]. But the nature of services that these works address, or the scenarios they apply in, have two fundamental differences with an NVE. First, traditional SLA management does not consider the process of determining the most appropriate set of providers, it focuses on negotiation, specification and monitoring of the service level obligations or SLAs for already selected provider(s). Second, SLA management generally deals with independent commodities or services. Even when a commodity or service is composed of multiple ones, consuming a commodity or service from one provider does not affect the cost of the remaining portions.

Service contracting and negotiation aspects in a multi-provider scenario have been addressed in peer-to-peer services [15], [16]. In these works, the services on offer are well defined and providers offer different QoS levels at different prices for these services. Different auction-based mechanisms are used to reconcile the interests of both providers and customers. In an NVE, the customer (SP) enjoys a much higher degree of service customizability, which introduces new challenges that do not exist in a P2P service environment.

VIII. Conclusion

In this paper, we have presented V-Mart, an automated service negotiation and contracting framework for NVE. We showed that V-Mart is essential in providing a truthful and fair market competition environment for the SPs and InPs to reach a fair and mutually beneficial contract through a two-stage auction model and a VN partitioning heuristic. The V-Mart approach is flexible in accommodating the cost minimization requirement of the SPs as well as the diverse pricing models and strategies of the InPs; it is efficient in runtime performance and exhibits very small auctioning overhead.

For future work, we plan to implement and test the V-Mart system on an experimental NVE where the topology is operational rather than random, and address the various implementation, trust and reputation issues.

Acknowledgement

This work was jointly supported by the Natural Science and Engineering Council of Canada (NSERC) under its Discovery program, Cisco Systems, and WCU (World Class University) program through the Korea Science and Engineering Foundation funded by the Ministry of Education, Science and Technology (Project No. R31-2008-000-10100-0).

References

[1] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet impasse through virtualization," *Computer*, vol. 38, no. 4, pp. 34–41, 2005.
[2] J. Turner and D. Taylor, "Diversifying the Internet," in *GLOBECOM'05*, vol. 2, 2005.
[3] N. M. M. K. Chowdhury and R. Boutaba, "Network virtualization: State of the art and research challenges," *IEEE Communications Magazine*, vol. 47, no. 7, pp. 20–26, July 2009.
[4] M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual Network Embedding with Coordinated Node and Lin Mapping," in *28th Conference on Computer Communications (IEEE INFOCOM)*, Rio de Janeiro, Brazil, April, 2009.
[5] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: substrate support for path splitting and migration," *Computer Communication Review*, vol. 38, no. 2, pp. 17–29, 2008.
[6] Y. Zhu and M. Ammar, "Algorithms for assigning substrate network resources to virtual network components," in *Proceedings of the IEEE INFOCOM'06*, 2006.
[7] "AT&T Managed Internet Service (MIS)," http://new.serviceguide.att.com/mis.htm, 2009.
[8] O. Goldschmidt and D. S. Hochbaum, "A polynomial algorithm for the k-cut problem for fixed k," *Math. Oper. Res.*, vol. 19, no. 1, pp. 24–37, 1994.
[9] "Amazon Elastic Compute Cloud (Amazon EC2)," http://aws.amazon.com/ec2/.
[10] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," *Journal of Finance*, vol. 16, pp. 8–37, 1961.
[11] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, "How to model an internetwork," in *INFOCOM*, 1996, pp. 594–602.
[12] A. Keller and H. Ludwig, "The wsla framework: Specifying and monitoring service level agreements for web services," *J. Network Syst. Manage.*, vol. 11, no. 1, 2003.
[13] P. Bhoj, S. Singhal, and S. Chutani, "Sla management in federated environments," *Comput. Netw.*, vol. 35, no. 1, pp. 5–24, 2001.
[14] K. Czajkowski, I. T. Foster, C. Kesselman, V. Sander, and S. Tuecke, "Snap: A protocol for negotiating service level agreements and coordinating resource management in distributed systems," in *JSSPP*, 2002, pp. 153–183.
[15] D. Hausheer and B. Stiller, "PeerMart: The technology for a distributed auction-based market for peer-to-peer services," in *Proceedings of IEEE ICC'2005*, 2005, pp. 1583–1587.
[16] Z. Despotovic, J.-C. Usunier, and K. Aberer, "Towards peer-to-peer double auctioning," in *Hawaii International Conference on System Sciences*, 2004.