

# زنجیره‌سازی کارکردهای مجازی سرویس شبکه با در نظر گرفتن محدودیت منابع مدیریتی

## مهندسی فناوری اطلاعات - شبکه‌های کامپیوتری

---

پرهام الوانی

شهریور ۱۳۹۸

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دکتر بهادر بخشی



## فهرست

مقدمه

سابقه‌ی کارها

تعریف مساله

فرمول‌بندی و مدل‌سازی ریاضی مساله

راه‌حل پیشنهادی

## مقدمه

---

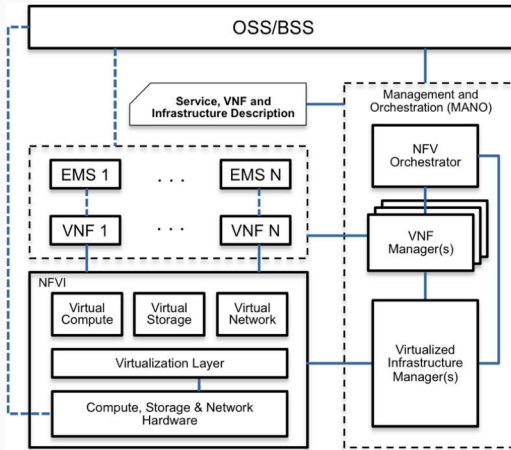
## مقدمه

- ◀ عدم انعطاف‌پذیری معماری فعلی شبکه
- ◀ در مجازی‌سازی کارکرد شبکه با استفاده از مجازی‌سازی منابع، می‌توان کارکردها را بر روی سرورهای استاندارد اجرا کرد و بهره‌وری منابع را افزایش داده و هزینه‌های انرژی را کاهش داد.
- ◀ زنجیره‌سازی کارکرد سرویس نیز امکان ایجاد زنجیره‌ای از کارکردها را به صورت پویا فراهم می‌کند.

## مقدمه

- ◀ با توجه به جداسازی زیرساخت از نرم‌افزار کارکردهای شبکه، نیاز به هماهنگی میان آنها ایجاد شده است.
- ◀ به صورت کلی تفاوت‌هایی که با توجه به فرآیند مجازی‌سازی کارکردهای شبکه ایجاد شده‌اند را می‌توان به ترتیب زیر دسته‌بندی نمود:
  - زیرساخت مجازی‌سازی شده
  - کارکردهای شبکه‌ای مجازی‌سازی شده
  - سرویس‌های شبکه‌ای

## مقدمه

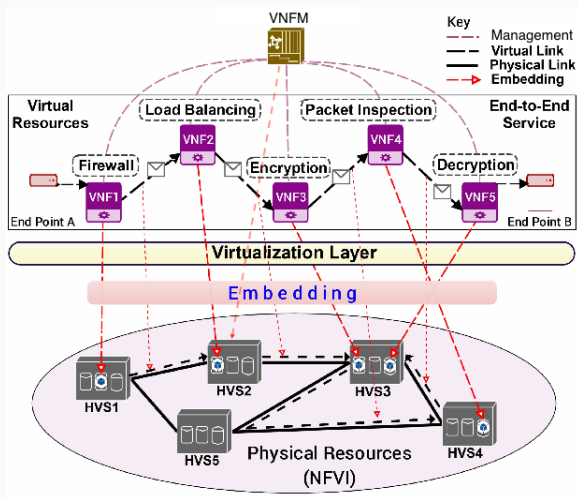


شکل ۱: معماری سطح بالای مجازی‌سازی کارکردهای شبکه

## مقدمه

- ◀ NFVO وظیفه‌ی استقرار زنجیره‌های کارکرد سرویس را برعهده دارد.
- ◀ VNFM مسئول چرخه‌ی زندگی کارکردهای مجازی شبکه می‌باشد.
- ◀ چرخه‌ی زندگی هر کارکرد مجازی شامل عملیات‌هایی همچون نمونه‌سازی، مقیاس‌کردن، به‌روزرسانی و پایان دادن می‌باشد.
- ◀ هر نمونه از کارکردهای مجازی شبکه نیاز دارد تحت مدیریت یکی از VNFM‌های موجود در شبکه باشد.

## چالش‌ها



◀ مدیریت و هماهنگی

◀ مصرف بهینه‌ی انرژی

◀ تخصیص منابع به

کارکردهای مجازی

◀ مسیریابی زنجیره‌های

کارکرد سرویس

◀ پذیرش زنجیره‌های

کارکرد سرویس

◀ به روزرسانی و مقیاس

کردن کارکردهای مجازی

سرویس



## سابقه‌ی کارها

---

## سابقه‌ی کارها

## جدول ۱: مقایسه مقالات پذیرش زنجیره‌های کارکرد سرویس

منبع	منابع	محدودیت	برخط	نگاشت	انتساب	اشتراک	تخصیص								
منبع	تخصیص یافته	ظرفیت پردازشی نمونه	یا برون خط	کارکرد و لینک	کارکرد لینک	یک نمونه	چند نمونه	دارد	ندارد	دارد	ندارد	دارد	ندارد	دارد	ندارد
#	other	MEM	BW	CPU	دارد	ندارد	برخط	برون خط	کارکرد لینک	یک نمونه	چند نمونه	دارد	ندارد	دارد	ندارد
۲۰۱۶-Eramo	—	✓	✓	✓	—	✓	—	✓	✓	✓	—	—	✓	—	✓
۲۰۱۷-Ghaznavi	—	✓	✓	✓	—	✓	✓	—	✓	—	✓	—	✓	—	✓
۲۰۱۷-Huang	—	✓	✓	✓	—	✓	✓	—	✓	—	✓	—	✓	—	✓
۲۰۱۷-VANDEDEH	—	—	—	—	✓	—	—	✓	—	—	—	—	—	—	—
۲۰۱۷-VANDEDEH capacity	—	—	—	—	✓	—	—	✓	—	—	—	—	—	—	—
پژوهش حاضر	✓	✓	✓	✓	—	✓	—	✓	✓	✓	—	—	✓	✓	—

## سابقه‌ی کارها

- ◀ این مقاله مساله‌ی جایگذاری VNFMs را مطرح می‌کند.
- ◀ این مقاله فرض می‌کند زنجیره‌های جایگذاری شده‌اند و هر در بازه‌ی زمانی می‌توانند بازنگاشت شوند.
- ◀ این مساله قصد دارد با در نظر گرفتن هزینه‌های عملیاتی مساله‌ی بازنگاشت VNFMs را در بازه‌های زمانی را حل کند.

**AbuLebdeh2017**

## تعریف مساله

---

## تعریف مساله

بیشینه‌سازی سود حاصل از پذیرش زنجیره‌های کارکرد سرویس با در نظر گرفتن کارکرد سرویس با در نظر گرفتن نیاز برخی از نمونه‌های کارکرد مجازی شبکه به VNFM.

## تعریف مساله

- ◀ توپولوژی زیرساخت شامل پنهای باند لینک‌ها و ظرفیت NFVI-PoP‌ها، موجود است.
- ◀  $n$  تقاضای زنجیره کارکرد سرویس به صورت کامل و از پیش مشخص شده داریم.
- ◀ هر تقاضا شامل نوع و تعداد نمونه‌های مجازی، پنهای باند لینک‌های مجازی و توپولوژی نمونه‌های مجازی می‌باشد.

## تعریف مساله

- ◀ نمونه‌ها بین زنجیره‌ها به اشتراک گذاشته نمی‌شوند.
- ◀ محدودیت ظرفیت لینک‌ها
- ◀ محدودیت توان پردازش سرورهای فیزیکی با توجه به میزان حافظه و تعداد پردازنده‌ها
- ◀ برخی از سرورهای فیزیکی نمی‌توانند سرورهای فیزیکی مشخصی را مدیریت کنند.
- ◀ برخی از سرورهای فیزیکی توانایی پشتیبانی از کارکردهای مجازی را ندارد.
- ◀ تنها برخی از نمونه‌های کارکردهای مجازی نیاز به مدیریت دارند.

## تعریف مساله

- ◀ برای مدیریت یک‌دست و آسان‌تر زنجیره‌ها و در عین حال جمع‌آوری راحت‌تر خطاها، برای هر زنجیره یک VNFM تخصیص می‌دهیم.
- ◀ VNFM ها می‌توانند بین زنجیره به اشتراک گذاشته شوند.
- ◀ هر نمونه از VNFM ها می‌تواند تعداد مشخصی از نمونه‌های کارکرد مجازی شبکه را سرویس دهد.
- ◀ برای ارتباط میان هر نمونه از VNFM ها و VNF ها پهنای باند مشخصی رزرو می‌گردد.
- ◀ در صورتی که NFVI-PoP بتواند از VNFM پشتیبانی نماید، می‌توان به هر تعداد که ظرفیت آن اجازه می‌دهد بر روی آن VNFM نصب نمود.



## چالش‌ها و نوآوری‌های مساله

- ◀ در نظر گرفتن نیازمندی نمونه‌های کارکرد مجازی به یک VNF
- ◀ در نظر گرفتن نیازمندی تاخیر برای لینک‌های مدیریتی
- ◀ تخصیص منابع مدیریتی به زنجیره‌ها و مسیریابی ارتباط مدیریتی
- ◀ جایگذاری و مسیریابی توانان زنجیره‌های کارکرد سرویس

## معیار و نحوه‌ی ارزیابی

- ◀ مدل‌سازی مساله
- ◀ حل مساله‌ی بهینه در ابعاد کوچک
- ◀ پیاده‌سازی راه‌حل مکاشفه‌ای
- ◀ معیار مقایسه این راه حل سود حاصل از پذیرش تقاضاهای زنجیره‌های کارکرد سرویس می‌باشد.
- ◀ مقایسه‌ی نتایج راه‌حل مکاشفه‌ای با جواب بهینه

## فرمول‌بندی و مدل‌سازی ریاضی مساله

---

## فرمول‌بندی

## پارامترهای مساله

$memory(k)$	required RAM of VNF instance with type $k$ in GB
$core(k)$	required CPU cores of VNF instance with type $k$
$memory$	required RAM of VNFM in GB
$core$	required CPU cores of VNFM
$capacity$	maximum number of VNF instances that VNFM can handle
$len(h)$	number of VNF instances in $h$ th SFC request

## فرمول‌بندی

## پارامترهای مساله

$type(v, k)$	assuming the value 1 if the VNF instance $v$ has type $k$
$bandwidth(u, v)$	required bandwidth in link from VNF instance $u$ to $v$
$\hat{bandwidth}$	required bandwidth in management link
$radius$	maximum neighborhood distance for instance management

## فرمول‌بندی

## پارامترهای مساله

$licenseFee$	VNFM license fee that must pay for each VNFM
$vnfSupport(w)$	assuming the value 1 if the physical server $w$ can support VNF instances
$isManageable(k)$	assuming the value 1 if the type $k$ needs a manager
$notManagableBy(w1, w2)$	assuming the value 1 if the physical server $w1$ cannot manage by physical server $w2$

## فرمول‌بندی

## متغیرهای تصمیم‌گیری

- $x_h$  binary variable assuming the value 1 if the  $h$ th SFC request is accepted; otherwise its value is zero
- $y_{wk}$  the number of VNF instances of type  $k$  that are used in server  $w \in V_s^{PN}$
- $z_{vw}^k$  binary variable assuming the value 1 if the VNF node  $v \in \cup_{i=1}^T V_{i,F}^{SFC}$  is served by the VNF instance of type  $k$  in the server  $w \in V_s^{PN}$

## فرمول‌بندی

متغیرهای تصمیم‌گیری

- $\bar{y}_w$  the number of VNFMs that are used in server  $w \in V_s^{PN}$
- $\bar{z}_{hw}$  binary variable assuming the value 1 if  $h$ th SFC is assigned to VNFM on server  $w \in V_s^{PN}$



## فرمول‌بندی

هدف اصلی مساله پذیرش بیشترین تعداد تقاضا می‌باشد. در اینجا فرض می‌کنیم پذیرش هر تقاضا سودی منحصر به فرد خواهد داشت. بنابراین تابع هدف به شکل زیر می‌باشد:

$$\max \sum_{h=1}^T x_h \quad (1)$$

## فرمول‌بندی

محدودیت حافظه نودها

$$\sum_{k=1}^F y_{wk} memory(k) + \bar{y}_w me\bar{m}ory \leq N_{ram}^{PN}(w) \quad \forall w \in V_s^{PN} \quad (2)$$

محدودیت تعداد پردازنده‌های نودها

$$\sum_{k=1}^F y_{wk} core(k) + \bar{y}_w c\bar{o}re \leq N_{core}^{PN}(w) \quad \forall w \in V_s^{PN} \quad (3)$$

## فرمول‌بندی

اگر VNF،  $v$  توسط VNF instance نوع  $k$  روی سرور  $w$  سرویس شود می‌بایست VNF instance نوع  $k$  روی سرور  $w$  فعال شود. توجه شود که اشتراک گذاری VNF ها پشتیبانی نمی‌گردد.

$$\sum_{v \in \bigcup_{i=1}^T V_{i,F}^{SFC}} z_{vw}^k \leq y_{wk} \quad \forall w \in V_s^{PN}, \forall k \in [1, \dots, F] \quad (4)$$

## فرمول‌بندی

اگر تقاضای  $h$ ام پذیرفته شده باشد می‌بایست تمام VNF node های آن سرویس شده باشند.  
یک VNF حداکثر یکبار سرویس داده شود.

$$x_h = \sum_{k=1}^F \sum_{w \in V_s^{PN}} z_{vw}^k \quad \forall v \in V_{h,F}^{SFC}, \forall h \in [1, \dots, T] \quad (5)$$

## فرمول‌بندی

اگر تقاضای  $h$ ام پذیرفته شده باشد می‌بایست توسط یک VNFم سرویس شده باشد. توجه شود که این محدودیت اجازه‌ی تخصیص بیش از یک VNFم به زنجیره نمی‌دهد.

$$x_h = \sum_{w \in V_s^{PN}} \bar{z}_{hw} \quad \forall h \in [1, \dots, T] \quad (6)$$

## فرمول‌بندی

اگر SFC،  $i$  توسط VNFم روی سرور  $w$  سرویس شود می‌بایست یک VNFم سرور  $w$  برای آن فعال شود.

$$\sum_{h=1}^T \bar{z}_{hw} \leq \bar{y}_w \quad \forall w \in V_s^{PN} \quad (7)$$

محدودیت ظرفیت سرویس‌دهی VNFم

$$\sum_{i=1}^T \bar{z}_{iw} * len(i) \leq capacity \quad \forall w \in V_s^{PN} \quad (8)$$

## فرمول‌بندی

اگر  $VNF$ ،  $v$  توسط نمونه‌ای نوع  $k$  روی سرور  $w$  سرویس می‌شود می‌بایست این  $VNF$  از نوع  $k$  ام باشد.

$$z_{vw}^k \leq type(v, k) \quad \forall w \in V_s^{PN}, \forall k \in [1, \dots, F], \forall v \in \cup_{i=1}^T V_{i,F}^{SFC} \quad (9)$$

## فرمول‌بندی

متغیرهای تصمیم‌گیری

- $\tau_{ij}^{(u,v)}$  binary variable assuming the value 1 if the virtual link  $(u, v)$  is routed on the physical network link  $(i, j)$
- $\bar{\tau}_{ij}^v$  binary variable assuming the value 1 if the management traffic of VNF node  $v$  is routed on the physical network link  $(i, j)$



## Flow Conservation

$$\sum_{(i,j) \in E^{PN}} \tau_{ij}^{(u,v)} - \sum_{(j,i) \in E^{PN}} \tau_{ji}^{(u,v)} = \sum_{k=1}^F z_{ui}^k - \sum_{k=1}^F z_{vi}^k$$

$$\forall i \in V_S^{PN}, (u, v) \in E_h^{SFC}, h \in [1, \dots, T] \quad (10)$$

## Flow Conservation

$$\sum_{(i,j) \in E^{PN}} \bar{\tau}_{ij}^v - \sum_{(j,i) \in E^{PN}} \bar{\tau}_{ji}^v = \sum_{k=1}^F z_{vi}^k - \bar{z}_{hi} \quad \forall i \in V_S^{PN}, v \in V_{h,F}^{SFC}, h \in [1, \dots, T] \quad (11)$$

## فرمول‌بندی

محدودیت ظرفیت لینک‌ها

$$\sum_{v \in \bigcup_{i=1}^T V_{i,F}^{SFC}} \bar{\tau}_{ij}^v * bandwidth + \sum_{(u,v) \in \bigcup_{i=1}^T E_i^{SFC}} \tau_{ij}^{(u,v)} * bandwidth(u,v) \leq C_{ij}$$

$$\forall (i,j) \in E^{PN}$$

$$(12)$$

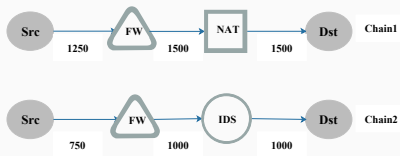
## فرمول‌بندی

شعاع همسایگی تضمین می‌کند که زمان سرویس‌دهی توسط VNFMs ها در یک بازه مشخص (از نظر تعداد گام) خواهد بود.

$$\sum_{(i,j) \in E^{PN}} \bar{\tau}_{ij}^v \leq radius \quad \forall v \in \cup_{i=1}^T V_{i,F}^{SFC} \quad (13)$$

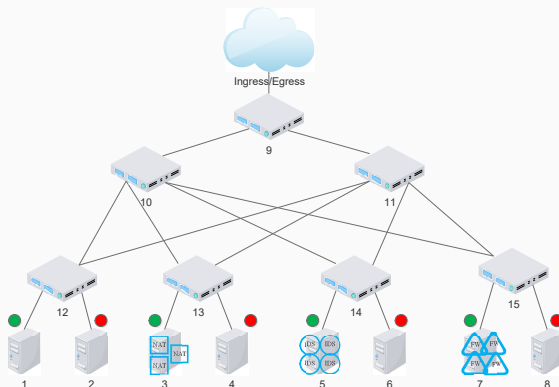
## مساله ی نمونه

زنجیره های زیر را به عنوان تقاضاها در نظر می گیریم.



## مساله‌ی نمونه

فرض می‌کنیم مرکز داده‌ای دارای توپولوژی زیر می‌باشد.



## راه‌حل پیشنهادی

---

## راه‌حل پیشنهادی

- ◀ مساله‌ی اصلی یک مساله‌ی NP-Hard می‌باشد.
- ◀ برای حل مساله در زمان معقول برای ابعاد بزرگ نیاز به یک الگوریتم مکاشفه‌ای می‌باشد.
- ◀ از ایده‌ی الگوریتم **Bari2015** برای جایگذاری زنجیره‌ها شروع می‌کنیم.



## ◀ Joint Service Deployment - Manager Placement

- ◀ زنجیره‌ها را با استفاده از الگوریتم **Bari2015** جایگذاری می‌کنیم.
- ◀ در زمان انتخاب مجموعه‌ی امکان‌پذیر محدودیت‌های مساله را اعمال می‌کنیم.
- ◀ بعد از جایگذاری هر زنجیره VNFم آن را انتخاب می‌کنیم.
- ◀ برای انتخاب VNFم اولویت با نمونه‌هایی است که ظرفیت آن‌ها کامل استفاده نشده است.
- ◀ در بین VNFم‌هایی که ظرفیت خالی دارند اولویت با نمونه‌هایی است که منابع پردازشی بیشتری دارند.

## eJSD-MP

- ◀ الگوریتم پیشنهادی JSD-MP زمان اجرای زیادی دارد که می‌توان آن را کاهش داد.
- ◀ الگوریتم پیشنهادی eJSD-MP از برون‌خط بودن مساله استفاده نمی‌کند.
- ◀ برای استفاده از ویژگی برون‌خط بودن مساله زنجیره‌ها را بر اساس قیمت‌شان مرتب می‌کنیم.
- ◀ برای کاهش زمان اجرای الگوریتم نسب مشخصی از زنجیره‌ها را با الگوریتم first-fit جایگذاری می‌کنیم.

## ارزیابی

## ارزیابی

فرمول‌بندی ارائه شده بر روی نرم‌افزار cplex که محصول شرکت IBM بوده و برای حل مسائل برنامه‌ریزی خطی و ... استفاده می‌شود، به زبان جاوا پیاده‌سازی شده و تست گشت.



## مراجع