Survivable Virtual Network Embedding with resource sharing and optimization

Shuopeng Li, Mohand Yazid Saidi, Ken Chen

Établissement de Réseaux Virtuels Protégés avec Partage et Optimisation de Ressources

Shuopeng LI
L2TI, Institut Galilée
Université Paris 13
Villetaneuse, France
li.shuopeng@univ-paris13.fr

Mohand Yazid SAIDI
L2TI, Institut Galilée
Université Paris 13
Villetaneuse, France
saidi@univ-paris13.fr

Ken CHEN
L2TI, Institut Galilée
Université Paris 13
Villetaneuse, France
ken.chen@univ-paris13.fr

***Résumé*—La virtualisation permet de disposer de plusieurs réseaux logiques indépendants par le partage des ressources d'un réseau substrat. Pour assurer un déploiement optimal des réseaux virtuels, un *mappage* efficace entre les ressources logiques réclamées par les réseaux virtuels d'une part, et d'autre part, les ressources physiques appartenant au réseau substrat doit être déterminé. Ce problème de mappage, appelé VNE (Virtual Network Embedding), est NP complet et est intensivement traité dans la littérature. Pour assurer une continuité de service, même après une panne, les réseaux virtuels doivent implanter la protection. Un des enjeux et défis majeurs est de protéger les réseaux virtuels en optimisant les ressources allouées dans le réseau substrat . Dans cet article, nous présenterons d'abord les principales approches VNE et la protection des réseaux classiques. Nous fournirons ensuite une analyse synthétique et critique des méthodes de protection des réseaux virtuels.**

I. Introduction

La virtualisation des réseaux [1] est une technique qui permet de créer plusieurs réseaux logiques qui partagent des ressources physiques appartenant à un même réseau physique dit *substrat*. La virtualisation permet non seulement de mieux utiliser les ressources mais aussi de séparer les flux pour mieux les contrôler. Par exemple, nous pouvons évaluer un nouveau protocole sur un réseau virtuel dédié et configurer un autre réseau virtuel pour un service particulier telle que la diffusion vidéo.

Un réseau virtuel (*Virtuel Network* : **VN**) est un réseau logique construit au dessus d'un réseau substrat. Il est constitué d'un ensemble de nœuds virtuels interconnectés par des liens virtuels. Un nœud virtuel est supporté par un nœud substrat choisi le cas échéant parmi une liste de candidats, tandis qu'un lien virtuel est formé d'un ou plusieurs chemins dans le réseau substrat.

Il y a deux acteurs principaux dans un réseau virtuel : i) le fournisseur de service (*Service Provider* : **SP**) qui a pour rôle de créer et gérer les réseaux virtuels (pour satisfaire ses besoins de services) ; ii) le fournisseur d'infrastructure (*Infrastructure Provider*, **InP**) dont le rôle est de fournir les ressources substrat nécessaires à la satisfaction des requêtes de création de réseaux virtuels émises par les SP. L'INP assure aussi le bon fonctionnement des VN.

L'établissement d'un réseau virtuel consiste à identifier les ressources du réseau substrat permettant de réaliser la topologie du VN demandé, c'est-à-dire ses nœuds et liens virtuels, avec les capacités réclamées (capacités de traitements des nœuds, bandes passantes, etc.). Il s'agit donc d'un problème de recherche de la correspondance (**mappage**) entre le réseau virtuel et une partie du réseau substrat. Ce problème est appelé **VNE** (*Virtual Network Embedding*) et est NP-difficile [2], [3].

Comme dans les réseaux classiques, la protection contre les pannes est désirée dans les réseaux virtuels afin d'assurer la continuité de service. Elle consiste à rechercher des routes de secours capables de router le trafic des communications affectées suite à une panne.

Bien que les méthodes classiques de protection restent, en principe, applicables aux réseaux virtuels, des modifications et/ou extensions de ces dernières sont nécessaires pour mieux répondre aux besoins spécifiques des réseaux virtuels résumés ci-après :

- La protection d'un seul réseau virtuel conduit généralement à la protection de plusieurs connexions entre couples de nœuds. Les méthodes classiques de protection hors ligne ne sont souvent pas efficaces pour protéger les réseaux virtuels puisqu'elles requièrent des temps de calcul élevés alors que les méthodes en ligne ne sont efficaces que pour la protection d'une seule connexion ;
- Il est possible de protéger les nœuds virtuels en migrant leurs trafics vers d'autres nœuds équivalents du réseau substrat.

Dans cet article, nous présenterons une analyse synthétique et critique de diverses méthodes de protection dédiées aux réseaux virtuels. Nous commencerons par introduire le problème VNE (section 2) et la protection classique (section 3) avant de nous focaliser sur la protection des réseaux virtuels et ses spécificités (section 4). Nous y décrirons diverses méthodes de protection de réseaux virtuels proposées dans la littérature en fournissant à chaque fois leurs avantages et inconvénients. La section 5 donnera nos conclusions.

II. Mappage des réseaux virtuels

Afin de permettre la création d'un réseau virtuel, le réseau substrat doit fournir des ressources pour supporter aussi bien les nœuds virtuels que les liens virtuels. Pour mieux utiliser les ressources du réseau substrat, ces dernières doivent être allouées d'une manière optimale aux réseaux virtuels. Ceci revient à résoudre le problème VNE qui est un problème de recherche combinatoire sous contraintes (sur les liens et les

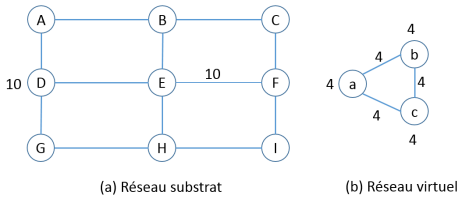

FIGURE 1. Réseau substrat et réseau virtuel.

nœuds) dont l'objectif est d'optimiser un ou plusieurs critères (bande passante totale, puissance de calcul, etc.). Selon le contexte, certains critères peuvent se transformer en contraintes et vice-versa. Par exemple, la bande passante sur un lien peut être une contrainte (contrôle d'admission) ou un objectif (remplissage maximal d'un lien).

A. *Formulation du problème*

Nous donnons ci-après quelques définition et notations (c.f. [4]) utiles à la compréhension du problème VNE.

Réseau substrat : Le réseau substrat est généralement modélisé comme un graphe non orienté $G^S(N^S, L^S)$, où N^S et L^S représentent respectivement l'ensemble des nœuds et des liens substrat. A un nœud substrat $n^s \in N^S$, nous associons différents attributs comme la puissance de calcul $cpu(n^s)$ et la localisation géographique $loc(n^s)$. De manière similaire, nous associons à tout lien substrat $l^s \in L^S$ divers attributs comme la bande passante $bw(l^s)$ et le délai $d(l^s)$.

Réseau virtuel : Le réseau virtuel est également modélisé comme un graphe non orienté $G^V(N^V, L^V)$. A tout nœud du graphe n^v sont associées des critères et/ou contraintes comme la puissance de calcul $cpu(n^v)$ et la localisation géographique $loc(n^v)$. De même, à chaque lien l^v sont associées des demandes liées à la QoS comme le délai $d(l^v)$ et la bande passante $bw(l^v)$). Généralement, un réseau virtuel G^v a une durée de vie limitée.

Mappage : Le problème de mappage (VNE) consiste à trouver une réalisation du réseau virtuel G^v sur un sous-ensemble du réseau substrat G^s de telle sorte que les contraintes/critères soient vérifiés/optimisés. A chaque nœud virtuel, nous faisons correspondre un seul nœud substrat vérifiant les contraintes (ex. la localisation) et optimisant les critères (par exemple, la puissance de calcul). De même, un lien virtuel l^v est mappé vers un ensemble de chemins de telle sorte que les contraintes soient vérifiées sur chacun des chemins (par exemple, le délai de chaque chemin inférieur ou égal à $d(l^v)$) et les critères optimisés (par exemple bande passante cumulée de tous les liens formant l'ensemble des chemins).

Sur la figure 1 (a) est illustré un exemple de réseau substrat. Ce dernier est constitué de 9 nœuds disposant chacun d'une puissance de calcul de 10 unités et 12 liens bidirectionnels de capacités égal à 10 unités. Sur la figure 1 (b) est illustré un exemple de requête d'un réseau virtuel. Cette dernière requiert 3 nœuds a, b et c d'une même puissance de calcul de 4 unités. Les nœuds sont tous interconnectés entre eux par des liens disposant d'une capacité de bande passante égale à 4 unités. En plus des contraintes sur les puissances de calcul et les bandes passantes, les requêtes exigent que les nœuds virtuels a, b et c soient respectivement supportés par des nœuds substrat appartenant aux ensembles $\{D\}$, $\{F, I\}$ et $\{H, A\}$.

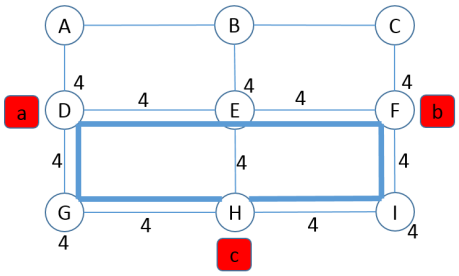

FIGURE 2. Exemple de mappage d'un réseau virtuel vers un réseau substrat.

Pour satisfaire la requête du réseau virtuel de la figure 1 (b), un mappage de cette requête vers le réseau substrat de la figure 1(a) est déterminé et est illustré sur la figure 2. Dans cet exemple, les nœuds virtuels a, b et c sont respectivement mappés vers les nœuds substrat D, F et H alors que les liens virtuels a-b, b-c et c-a ont été mappés respectivement vers les chemins D-E-F, F-I-H et H-G-D. Notons que pendant la durée de vie du réseau virtuel de la figure 1(b), les capacités de traitements de tous les nœuds substrat D, E, F, G, H et I sont diminuées de 4 unités. Idem pour les bandes passantes disponibles sur les liens D-E, E-F, F-I, I-H, H-G et G-D qui sont diminuées de 4 unités.

Problème d'optimisation : VNE est un problème d'optimisation sous contraintes, ces dernières peuvent être d'ordre applicatif, technologique, économique, etc. Un des critères majeurs d'optimisation consiste à *minimiser* les ressources (ex. la bande passante cumulée) substrat allouées à chaque nouveau VN.
Ci après une fonction objective typique visant à optimiser le revenu :

$$R(G^V) = \sum_{l^v \in L^v} \sum_{l^s \in L^S} bw(l^s, l^v) + \sum_{n^v \in N^v} cpu(n^v) \quad (1)$$

où $bw(l^s, l^v)$ désigne la quantité de bande passante dédiée au lien virtuel l^v sur le lien substrat l^s.

B. *Méthodes de résolution du VNE*

Les méthodes de mappage des réseaux virtuels vers les réseaux substrat peuvent être groupées en deux catégories : méthodes de mappage disjoint (en deux étapes) et méthodes de mappage conjoint (en une étape).

1) mappage disjoint : Avec le mappage disjoint, le mappage des nœuds est effectué à la première étape, avant la seconde étape qui consiste à mapper les liens. Cette séparation a pour but d'accélérer les calculs (en réduisant la complexité notamment).

mappage des nœuds : Pour réduire la complexité du problème, un algorithme glouton est souvent utilisé [5], [6]. Un ensemble de nœuds substrat vérifiant les contraintes de la location géographique est d'abord choisi. Ensuite, le mappage des nœuds virtuels vers les nœuds substrat appartenant à l'ensemble déterminé précédemment est effectué de manière à optimiser une fonction combinant des critères de nœuds et

de liens. Une fonction typique, proposée dans [6], pour un nœud substrat n^s est donnée ci-après :

$$H(n^s) = cpu(n^s) \sum_{l^s \in L(n^s)} bw(l^s) \quad (2)$$

où $L(n^s)$ désigne des liens substrat adjacents de n^s.
Cette fonction prend en compte non seulement les puissances de calcul disponibles sur les nœuds substrat, mais aussi les bandes passantes disponibles sur leurs liens substrat adjacents. Elle est assez efficace pour éviter les goulets d'étranglement et est moins coûteuse en termes de temps de calcul.

mappage des liens : Le mappage des liens correspond à un problème d'allocation de flux entre des nœuds déterminés puisqu'il intervient après le mappage des nœuds. Si un seul chemin substrat est acceptable et utilisable pour mapper un lien virtuel, ce problème est NP-complet et rentre dans la catégorie des problèmes dits *Unsplittable Flow Problem* (UFP). Par contre, si l'on autorise le mappage d'un lien virtuel vers plusieurs chemins substrat (le trafic du lien virtuel est alors réparti sur ces chemins) [6], le mappage correspondra au problème dit *Multi-commdity Flow* (MCF) qui est solvable en un temps polynomial.

Cette approche est très difficile à implémenter puisque le partitionnement d'un trafic en plusieurs flux peut induire des dé-séquencements des messages à l'arrivée. Il est à signaler qu'en cas de migration d'un nœod virtuel vers un autre nœud substrat, le mappage de liens doit être refait [6].

2) Mappage conjoint : Le mappage en 2 étapes (nœuds puis liens) souffre d'un manque de collaboration entre les deux phases qui pourrait conduire à des solutions moins optimales. Le procédé de mappage des nœuds et des liens en une seule étape tente de remédier à cet inconvénient.

Dans [7], un graphe substrat augmenté par l'ajout de nœuds virtuels est déduit. Les nœuds virtuels sont interconnectés aux nœuds substrat pouvant les supporter avec des méta-liens. Après cette étape d'ajout, le problème VNE peut être résolu en mappant uniquement les liens.

Cheng et al. [8] proposent un algorithme de mappage basé sur la connaissance de la topologie du réseau. Les nœuds sont classés en fonction de leurs puissances de calcul et des bandes passantes de leurs liens adjacents (une variante de l'équation (2) est utilisée). Le graphe substrat est ensuite converti en arbre de parcours en largeur avec des valeurs de classement des nœuds. Le mappage des nœuds et des liens se fait sur cet arbre en adoptant le stratégie de retour en arrière (*backtracking*) en cas de contraintes non satisfaites.

3) VNE dynamique : Outre des approches indiquées ci-dessus, la communauté se penche sur le cas de requêtes de VN dynamiques. En effet, pour satisfaire les exigences de certaines applications temps réel, les ressources (ex. puissance de calcul sur les nœuds) d'un VN sont amenées à évoluer au fil du temps, ce qui nécessiterait une reconfiguration dynamique et rapide du mappage. Ce problème a été traité dans [9] qui l'a formulé par la programmation linéaire en entiers mixtes avec l'objectif de minimiser le coût de reconfiguration.

III. Protection dans les réseaux classiques

Avec le succès et le déploiement large d'applications temps réel (VoIP, visioconférences, etc.), la protection contre les pannes est de plus en plus désirée voire nécessaire. Par un calcul des chemins de secours capables de recevoir et de router le trafic des communications affectées suite àune panne, la protection permet d'assurer la continuité de service.

Différentes méthodes ont été proposées [10], [11], [12], [13] pour protéger les réseaux classiques. Parmi ces méthodes, certaines assurent la disponibilité des ressources après une pannes, d'autres sont capables de faire face à plusieurs pannes, etc. Nous donnons ci-après une classification des méthodes de protection selon divers critères.

A. Couche OSI d'implémentation de la protection

Selon le type du réseau physique, la protection peut être implantée d'une manière simple et efficace au niveau de la couche Physique ou la couche Liaison de Données. Il est bien connu qu'à la couche Physique, la topologie en anneau est protégée par la mise en place d'un double anneau (avec sens de transmission contraire) qui assure une double protection : contre la panne sur un anneau (panne d'un émetteur par exemple) ou une coupure des 2 anneaux (coupure physique par exemple).

Afin de mieux protéger les réseaux optiques *WDM*, [10] propose de construire et pré-configurer des anneaux, appelés P_cycles. Chaque chemin optique traversant un lien $a \rightarrow b$ est donc protégé grâce à l'établissement d'un circuit (un P_cycle, $b \rightarrow a... \rightarrow b$) incluant les points a et b.

Bien que les méthodes de protection décrites précédemment s'avèrent efficaces en termes de rapidité de récupération et d'utilisation des ressources, ces dernières restent spécifiques à certains types de réseaux et ne sont pas applicables dans le contexte général (réseaux hétérogènes, réseaux virtualisés, etc.). Ceci a motivé l'intérêt pour les méthodes de protection de niveaux supérieurs, notamment la protection au niveau IP ou MPLS. Dans ces réseaux, la protection est rendue avec le calcul et éventuellement la configuration de chemins redondants reliant des couples de nœuds appartenant aux chemins primaires.

Noton qu'une panne d'un composant au niveau Physique ou Liaison de Données peut entraîner la panne de plusieurs composants au niveau Réseau. Trois type de risques de panne doivent donc être protégés : nœud (node risk), lien (link risk) ou groupe de liens (Shared Risk Link group ou SRLG).

B. Protection (proactive) versus Restauration (réactive)

Avec la restauration [14], aucun calcul de chemins de secours n'est effectué avant l'apparition de pannes. La récupération est donc réalisée au moment de la détection d'une panne avec le calcul et la configuration de nouveaux chemins de secours pour router le trafic des communications affectées par la panne. Ce type de méthodes optimise l'utilisation des ressources puisqu'aucune ressource n'est allouée aux chemins de secours avant les pannes. Cependant, elle ne garantit pas le succès de la récupération puisque les ressources disponibles peuvent être insuffisantes pour établir tous les chemins de secours nécessaires à la récupération. De plus, la restauration

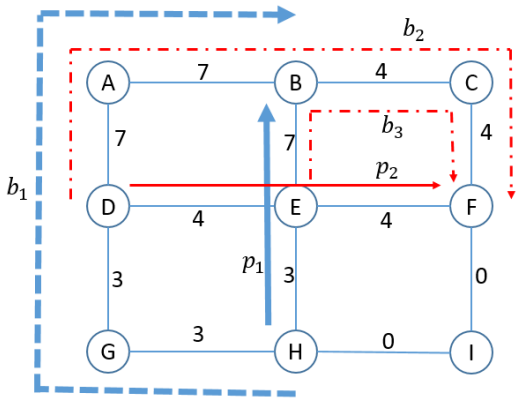

FIGURE 3. Protection globale et locale partagée

n'est pas efficace pour protéger les applications temps réel puisque ses délais de récupération sont élevés.

Pour réduire les délais de récupération et assurer le succès de la récupération d'une panne, la protection est souvent préférée à la restauration. Avec la protection, les chemins de secours sont calculés et souvent pré-configurés avant toute panne.

C. Protection globale versus Protection locale

Avec la protection globale, tout chemin primaire est protégé par un chemin de secours disjoint et reliant la source à la destination. Lorsqu'une panne survient, un message de notification de la panne est envoyé à la source qui basculera le trafic du chemin primaire vers le chemin de secours. Ce type de protection augmente le délai de récupération notamment lorsque la source est loin du nœud détectant la panne, génère du trafic pour notifier la panne et cause une perte d'un volume élevé de données dans les réseaux haut débit.

Sur la figure 3, un premier chemin primaire $p_1 = H \rightarrow E \rightarrow B$ de 3 unités de bande passante est établi. Ce chemin est globalement protégé par un chemin de secours $b_1 = H \rightarrow G \rightarrow D \rightarrow A \rightarrow B$. A la détection de la panne du lien $E \rightarrow B$, le nœud E envoie un message de notification d'erreur vers la source H qui basculera le trafic du chemin primaire p_1 vers le chemin de secours b_1.

Pour améliorer la protection et réduire le délai de récupération, la protection locale propose de configurer un chemin de secours sur chaque nœud du chemin primaire. Deux types de chemins de secours locaux ont été définis [11] : next hop (NHOP) et next next hop (NNHOP). Un chemin NHOP est établi entre un nœud du chemin primaire appelé Point of Local repair (PLR) et un nœud situé en aval du PLR appelé merge point (MP). Il protège et contourne le lien situé en aval du PLR sur le chemin primaire. Un chemin NNHOP est établi entre un PLR et un nœud primaire MP situé en aval du prochain nœud du PLR. Il protège et contourne les lien et nœud situés en aval du PLR sur le chemin primaire. Lorsqu'une panne est détectée par un PLR, ce dernier bascule localement et rapidement le trafic du chemin primaire correspondant vers son chemin de secours.

Sur la figure 3, un second chemin primaire $p_2 = D \rightarrow E \rightarrow F$ de 4 unités de bande passante est établi. Ce chemin est protégé par deux chemins de secours : $b_2 = D \rightarrow A \rightarrow B \rightarrow C \rightarrow F$ et $b_3 = E \rightarrow B \rightarrow C \rightarrow F$. Le premier chemin de secours b_2 (NNHOP) protège contre la panne des nœud E et lien $D \rightarrow E$ alors que le second chemin de secours b_3 (NHOP) protège contre la panne du lien $E \rightarrow F$. À la détection de la panne du lien $E \rightarrow F$, le nœud PLR E bascule localement et rapidement le trafic du chemin primaire p_2 vers le chemin de secours b_3 sans envoi de message de notification de la panne. De même, à la détection d'une panne affectant le lien D-E ou le nœud D, le PLR D bascule le trafic du chemin primaire p_2 vers le chemin de secours b_2.

D. Protection dédiée versus Protection partagée

Dans la protection dédiée, aucun partage de ressources (bande passante par exemple) n'est effectué entre les chemins. Chaque chemin (primaire ou secours) dispose de ses propres ressources qu'il n'est pas possible d'allouer à un autre chemin.

Avec l'adoption de l'hypothèse pratique de pannes simples (une seule panne à la fois), les chemins de secours protégeant contre des risques de panne différents ne peuvent pas être activés en mêmes temps. Avec la protection partagée, ces chemins sont amenés à partager leur ressources afin de les mieux utiliser. Pour déterminer la quantité de bande passante de secours nécessaire sur un lien λ pour faire face à la panne du risque r, la notion de coût de protection δ_r^λ a été définie :

$$\delta_r^\lambda = \sum_{b_r^\lambda \in B_r^\lambda} bw(b_r^\lambda)$$

Où B_r^λ est l'ensemble des chemins de secours protégeant contre la panne du risque r et traversant le lien λ.
Avec la définition du coût de protection δ_r^λ, nous déterminons la quantité de bande passante de secours G^λ à réserver sur un lien λ pour faire face à n'importe qu'elle panne comme suit :

$$G^\lambda = \max_r \delta_r^\lambda = \max_r \sum_{b_r^\lambda \in B_r^\lambda} bw(b_r^\lambda)$$

Sur la figure 3, le lien $D \rightarrow A$ est traversé par deux chemins de secours b_1 et b_2. En rappelant que b_1 et b_2 protègent respectivement contre les ensembles de risque de pannes $\{H$-E, E et E-$B\}$ et $\{D$-E, $E\}$, nous déterminons les coût de protection sur le lien $D \rightarrow A$ comme suit :
$\delta_{H-E}^{D\rightarrow A} = \delta_{E-B}^{D\rightarrow A} = 3$, $\delta_{D-E}^{D\rightarrow A} = 4$, $\delta_{E}^{D\rightarrow A} = 4+3 = 7$
En conséquence, la quantité de bande passante de secours $G^{D\rightarrow A}$ à allouer sur le lien $D \rightarrow A$ correspond à :
$G^{D\rightarrow A} = \max(\delta_{H\rightarrow E}^{D\rightarrow A}, \delta_{E\rightarrow B}^{D\rightarrow A}, \delta_{D\rightarrow E}^{D\rightarrow A}, \delta_{E}^{D\rightarrow A}) = 7$
Notons que les étiquettes sur les liens de la figure 3 illustrent les quantités de bande passante allouées sur chaque lien pour l'ensemble des chemins primaires et de secours.

E. Protection hors ligne versus Protection en ligne

La protection hors ligne est rendue lorsque la matrice du trafic est connue alors que la protection en ligne est réalisée pour offrir rapidement une protection aux demandes de connexion arrivant en ligne et non connues à l'avance. La protection hors ligne permet l'optimisation des ressources mais requiert des temps de calculs élevés. Elle est donc limitée à la protection des connexions permanentes et à la ré-optimisation des routes.

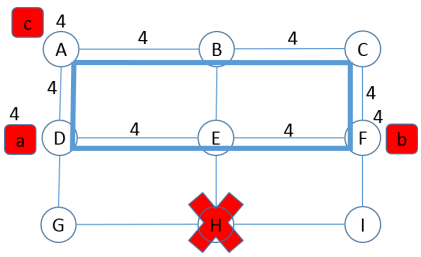

FIGURE 4. protection de nœud virtuel a

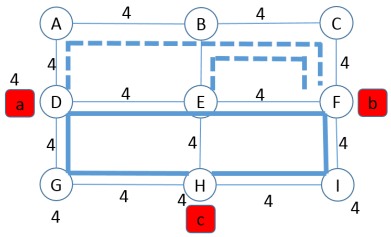

FIGURE 5. réservation de ressource pour le risque de lien virtuel a-b

F. *Optimisation conjointe versus Optimisation disjointe*

Les chemins de secours peuvent être calculés après avoir déterminé les chemins primaires (optimisation disjointe) ou en même temps (optimisation conjointe). Évidemment, le calcul simultané des chemins primaires et de leurs secours rationalisent les ressources mais conduit à l'utilisation de chemins primaires non optimaux en l'absence de pannes.

IV. PROTECTION DES RÉSEAUX VIRTUELS

Comme dans les réseaux classiques, la protection est également désirée pour des réseaux virtuels. La particularité dans ce genre de protection est qu'il y a deux acteurs, à savoir le fournisseur de services (SP) qui gère le réseau virtuel et l'opérateur d'infrastructure (InP) qui fournit les ressources. Certains auteurs [4] parlent du problème SVNE (pour *Survivable VNE*). Si la protection peut se fait de manière indépendante par le SP (par exemple en demandant un VN qui incorpore déjà une certaine redondance) ou l'InP (qui protège de manière classique ses ressources, et, par ce biais, les VN qu'elles supportent), il est évident que seule une considération combinée, intégrant les souhaits de protection de SP et les ressources physiques à mettre en œuvre, permet d'obtenir une vraie solution optimisée. Dans la suite, nous proposons une analyse synthétique des différents plans d'actions possibles, ainsi que certaines méthodes proposées

A. *Synthèse des approches*

1) Éléments à protéger : Protéger un VN revient à protéger ses nœuds et ses liens qui correspondent à des ressources différentes.

Une des particularités des VN est l'existence des nœuds virtuels qui sont supportés par un nœud substrat. Pour pallier la panne d'un nœud virtuel, un nœud substrat de secours doit être choisi pour supporter le nœud en panne. Ceci revient donc à remapper le réseau virtuel sur le réseau substrat [15], [16], [17], [18] en éliminant le nœud substrat en panne. Pour accélérer les calculs, certaines approches se contentent de remplacer le nœud substrat supportant le nœud virtuel défaillant par un autre nœud substrat (ou le même s'il y a suffisamment de ressource) et de remapper les liens virtuels adjacents au nœud défaillant vers le nœud de remplacement. Par exemple, lors de la panne du nœud H de la figure 2, le nœud virtuel c sera migré vers le nœud surbstrat A pour satisfaire la contrainte de localisation. En conséquence, les deux liens virtuels adjacents au nœud substrat H seront remappés sur les chemins A-D et A-B-C-F.

Un lien virtuel correspond à un ou plusieurs chemins substrat. Un chemin substrat est une chaîne de liens substrat avec des nœuds intermédiaires. La panne d'un lien virtuel peut donc être due à la panne d'un lien substrat ou à celle d'un nœud intermédiaire substrat. La protection d'un lien virtuel [19], [4], [20], [21], [22] vise à réparer le chemin substrat supportant le lien virtuel tout en optimisant différents métriques.

Sur la figure 5 est illustrée la réservation de ressources pour le risque de lien a-b. Un chemin substrat D-A-B-C-F est choisi pour protéger localement le lien virtuel a-b (supporté par le chemin primaire D-E-F) contre la panne du nœud E.De même, le chemin E-B-C-F protège le lien substrat E-F du lien virtuel a-b. L'établissement de ces chemins de secours conduit aux réservations de ressources illustrées sur la figure 5 si l'on considère que les liens et nœuds virtuels réclament tous 4 unités de puissance de calcul ou de bande passante.

2) Niveau de protection : Vu qu'il y a deux topologies, celle du VN et celle du réseau substrat, les deux niveaux de protection ci-dessous sont faisables.

a) Protection au niveau virtuel : Certains auteurs [6] proposent d'utiliser le splittage de liens comme moyen de protection. Ce genre d'approche a l'avantage de ne pas demander de configuration spéciale, mais n'est pas efficace en termes de ressource. De manière plus générale, on peut intégrer la protection dans un problème VNE en y ajoutant des contraintes ou éléments supplémentaires par rapport au problème VNE initial. Par exemple, [18], [15] ajoutent des nœuds et des liens virtuels de secours sur le réseau virtuel afin de protéger contre la panne d'un nœud virtuel. Le nouveau réseau virtuel avec les nœuds et liens supplémentaires est traité dans un problème VNE. Le mappage de ce nouveau réseau virtuel fournit la protection au réseau virtuel initial. Dans [4], des liens virtuels disjoints et complémentaires aux liens virtuels primaires ont été rajoutés à la formulation du VNE pour offrir la protection.

b) Protection au niveau substrat : On peut considérer un réseau virtuel comme une collection de sources de trafics. Dans des réseaux classiques, par exemple MPLS, une requête contient un seul flux de bout en bout. Par contre, une requête de réseau virtuel est composé de plusieurs flux. L'optimisation de la protection de plusieurs flux en ligne (conjointement avec des flux existants) est plus difficile. [19] utilise l'algorithme de ré-acheminement rapide d'IP (*IP fast reroute*) pour des liens avec une charge important (nombre de VN sur un lien supérieur à un seuil). [20] protège des liens substrat localement avec partage de ressource en optimisant conjointement la totalité des ressources primaires et secours des liens virtuels.

3) Métrique à optimiser : On peut aussi examiner les méthodes de protection selon leurs critères d'optimisation.

a) Bande passante additionnelle : La bande passante du lien est une des métriques/contraintes les plus importantes. La protection des liens réclame nécessairement de la bande passante additionnelle. Dans [21], une méthode d'optimisation disjointe est proposée, celle-ci minimise séparément la bande passante primaire et secours additionnelle. Guo et al. [20] optimise la bande passante additionnelle mais les bande passantes de secours et primaire sont optimisées dans un seul problème LP.

b) Puissance de calcul : La puissance de calcul sur un nœuds joue le même rôle que la bande passante sur un lien. Pour mieux utiliser cette ressource, la protection des nœuds [15] doit minimiser les puissances de calcul sur les nœuds virtuels primaires et de secours. Lorsque les ressources sont insuffisantes pour protéger un lien virtuel, une des extrémités du lien devrait être migrée vers un nœud de secours : la puissance de calcul peut être utilisée comme critère pour le choix de ce nœud de secours [21].

c) Délai : L'optimisation de la bande passante additionnelle peut conduire à des chemins qui peuvent être arbitrairement longs. Pour certains type d'application, l'optimisation des délais[23] est nécessaire et est plus importante que celle de la bande passante.

d) Taux de protection : Il est prévisible que l'utilisation des VN sera de plus en plus répandue et un même réseau substrat sera appelé à supporter de nombreux VN, alors que ses ressources sont limitées. Il se peut que tous les VN ne peuvent pas être protégés et la question d'une protection différentielle se pose. Rahman et al. [4] proposent de maximiser le taux de protection en minimisant la pénalité de violation de la protection qu'ils proposent d'associer aux liens virtuels.

e) Fiabilité : Le critère de fiabilité du réseau virtuel peut être mesuré par la probabilité qu'une panne l'affecte. Optimiser ce critère revient à réduire le risque de coupure des services offerts sur le VN.

Ce critère a été étudié par Soualah et al. [24] qui proposent de minimiser la probabilité de panne de l'ensemble des équipements utilisés par un VN, ce qui permettra d'augmenter la durée de vie de ce VN. Pour ce faire, la probabilité de défaillance de tout équipement a été formulée par une fonction inversement proportionnelle à l'age d'un équipement.

4) Centralisé versus distribué : La protection peux être fournie de manière centralisée ou distribuée. Les deux approches sont différents et chacune a ses avantages et inconvénients.

a) centralisé: Une entité centrale se charge de calculer les protections. C'est l'approche la plus répandue pour protéger les réseaux virtuels.

b) distribué: Avec cette approche, le calcul des nœuds et routes de secours est établi par plusieurs éléments du réseau. Étant donné le succès des architectures centralisées de réseaux virtuels et la difficulté de calcul et d'optimisation distribués, rares sont les approches de protection distribuée [16].

B. Classification des méthodes

Les méthodes de protection de réseaux virtuels peuvent être classées suivant différents critères. Sur la table I, nous les avons classé selon les critères décrits précédemment dans la section IV-A.

Dans la suite de cette section, nous donnerons des exemples de méthodes de protection de réseaux virtuels pour chaque catégorie de critères de la table I.

1) Protection de liens virtuels : Un lien virtuel peut être protégé globalement par un ou plusieurs autres liens virtuels ou localement en utilisant des détours locaux pour les nœuds et liens substrat qui le constituent.

Dans [6], les auteurs proposent de protéger un lien virtuel mappé sur plusieurs chemins substrat en redirigeant le trafic de ces chemins affectés vers les autres en cas de panne. Par exemple, si un lien virtuel l^v réclamant 20 unités de bande passante est mappé vers deux chemins substrat p_1 et p_2 de 10 unités de bande passante chacun, toute panne affectant le chemin p_1 sera réparée en remappant l'intégralité du lien l^v vers le second chemin substrat p_2. Cette méthode a l'avantage de ne requérir aucun nouveau calcul pour fournir la protection mais ne garantit pas la disponibilité des ressources pour pallier une panne. De plus, la probabilité qu'une panne affecte plusieurs chemins substrat supportant le même lien virtuel n'est pas négligeable, ce qui peut faire échouer la récupération.

Dans [4], Rahman et al. proposent une méthode de protection dédiée et proactive (**PROACTIVE**) des liens virtuels. Pour chaque lien virtuel, deux chemins substrat disjoints sont déterminés et pré-réservés : un chemin primaire utilisé en l'absence de pannes et un chemin de secours activé suite à une panne affectant le chemin primaire. Afin de maximiser le taux des liens virtuels protégés, les auteurs associent à chaque lien non protégé une pénalité qui dépend de l'importance du lien dans le réseau virtuel correspondant. De cette manière, une protection efficace est fournie au réseau virtuel en minimisant la somme des pénalités des liens qui le constituent. Notons que pour réduire la complexité de la protection, [4] propose de séparer la bande passante disponible sur un lien en deux pools séparés : un pool primaire pour allouer la bande passante aux chemins primaires et un pool de secours pour allouer la bande passante aux chemins de secours. De cette manière, les phases de mappage et de protection peuvent être séparées et exécutées en un temps polynomial [1].

Cette méthode de protection présente les désavantages de la protection dédiée au niveau réseau virtuel, à savoir des délais de récupération élevés et un gaspillage de ressources.

2) Niveau substrat : Pour pallier les désavantages de la méthode précédente, une deuxième méthode de protection locale est proposée dans [4]. Dans cette proposition, plusieurs détours locaux sont pré-calculés pour protéger un lien substrat supportant un lien virtuel. Lorsqu'une panne survient, un programme linéaire minimisant les pénalités est exécuté pour choisir l'ensemble des détours locaux vérifiant le contrôle d'admission (contraintes de la bande passante) et permettant la récupération.

Bien que cette méthode de protection pré-calcule les détours locaux, elle ne réserve aucune ressource avant la survenue d'une panne. Elle ne garantit donc pas le succès de la récupération.

1. [4] a formulé en MIP ce même problème pour le cas de calcul conjoint.

	Nœud, Lien	Métrique	Proactive vs. Reactive	Niveau	Partagé vs dédié	Centralisé vs. distribué
Rahman[4] PROACTIVE	Lien	Taux de protection	Proactive	Virtuel	Dédié	Centralisé
Rahman[4] HYBRID	Lien	Taux de protection	Reactive	Substrat	/	Centralisé
Guo[20] SOD_BK	Lien	bande passante	Proactive	Substrat	Partagé	Centralisé
Guo[20] SP_BK	Lien	bande passante	Proactive	Statique		Centralisé
Yu[21] P_SVIMA	Lien	bande passante	Proactive	Virtuel	Partagé	Centralisé
Yu[21] MP_SVIMA	Lien	bande passante + charge de calcul	Proactive	Virtuel	Partagé	Centralisé
Yu[15] K-redundant	Nœud	Charge de calcul	Proactive	Virtuel	Partagé	Centralisé
Guo[17]	Nœud	bande passante	Proactive	Virtuel	Partagé	Centralisé
Yu[19]	Lien	bande passante	Réactive	Substrat	/	Centralisé
Zhang[23]	Lien	Délais	Réactive	Virtuel	/	Centralisé
Houidi[16]	Nœud	Charge de calcul	Réactive	Virtuel	/	Distribué
Soualah[24]	Nœud, Lien	Fiabilité	Réactive	Virtuel	/	Centralisé

TABLE I. CLASSIFICATION DES MÉTHODES

3) Protection Partagée : Pour garantir le succès de la récupération, les ressources doivent être réservées à l'avance. Dans [20], les auteurs proposent de protéger localement les liens substrat supportant un lien virtuel tout en pré-réservant et partageant les ressources. Après une première phase de mappage des nœuds virtuels, les phases de mappage et de protection des liens virtuels sont conjointement exécutées. Pour ce faire et pour réduire la complexité du problème, un ensemble de chemins primaires et de secours est pré-calculé. Cet ensemble sert d'entrée pour un programme linéaire dont l'objectif est de minimiser les allocations de la bande passante tout en équilibrant la charge.

Cette méthode, dite SOD_BK, n'est pas optimale et n'explore qu'un sous ensemble du domaine de recherche des chemins puisque ceux-ci sont déterminés à l'avance et avant toute panne.

Une seconde méthode de protection locale et statique, dite SP_BK, est proposée dans [20]. Cette méthode a pour but de déterminer le flux maximum primaire qui peut être protégé sur chaque lien substrat. Pour ce faire, le revenu (c.f. équation (1)) est optimisé en acceptant un maximum de requêtes de réseau virtuels protégés. Cette méthode a pour but de déterminer le ratio de bande passante de secours à réserver sur chaque lien substrat afin de fournir la protection. Évidemment la pré-réservation de la bande passante de secours pour des requêtes de protection de VN qui risquent de ne jamais arriver conduit à un gaspillage notamment dans les réseaux moins surchargés. Cela peut aussi induire des échecs d'établissement de VN alors que les ressources sont suffisantes.

4) optimisation de la bande passante et de la charge de calcul : Yu et al. proposent une méthode appelée P_SVIMA [21] pour protéger globalement les liens virtuels. Ainsi, tout chemin substrat primaire est protégé par un autre chemin substrat qui lui est disjoint. Afin de mieux utiliser les ressources, les chemins de secours correspondent aux plus courts chemins en termes de bande passante de secours additionnelle. Cette méthode de protection est simple et rapide mais présente un taux de protection relativement faible notamment dans les topologies de réseau qui ne sont pas très connectées.

Lorsque la protection d'un lien est infaisable avec P_SVIMA pour cause d'un manque de ressources, une deuxième méthode de protection MP_SVIMA est exécutée. Cette dernière détermine un nœud de secours pour l'une des extrémités du lien non protégé et calcule un nouveau mappage (utilisé comme secours) pour tous ses liens adjacents. Évidemment, pour garantir suffisamment de ressources après une panne, ces dernières doivent être pré-réservées sur les nœuds et liens de secours. Une fonction d'objectif visant à optimiser la puissance de calcul et les bandes passantes des liens adjacents au nœud de secours est déterminée et est utilisée. Cette seconde méthode de protection est efficace pour éviter les goulets d'étranglement et augmenter le taux de protection mais elle est coûteuse à implémenter (difficulté de basculement vers un autre nœud virtuel, échanges de messages, etc.).

5) Protection des nœuds : Yu et al. [15] propose une approche, dite *K-redundant*, pour la protection des nœuds virtuels. Cette approche enrichit le réseau virtuel en lui ajoutant *K super* nœuds virtuels reliés par des liens virtuels aux nœuds à protéger. Pour une meilleure utilisation des ressources, la valeur du paramètre *K* doit dépendre de la topologie du réseau substrat, du réseau virtuel et des critères d'optimisation.

Cette approche permet le partage de ressources puisqu'un *super* nœud virtuel pourrait permettre de protéger plusieurs nœuds virtuels.

Avec cette approche, la requête de réseau virtuel est modifiée et est augmentée pour assurer le remplacement de tout nœud virtuel tombé en panne. Les ressources sont réservées sur les liens reliant les super nœuds aux nœuds virtuels. De cette manière, la protection des nœuds virtuels revient à trouver un mappage efficace du réseau virtuel augmenter vers le réseau substrat. Ce problème a été formulé en MIP dans [15] et une heuristique de protection disjointe est proposée.

6) Distribué : Houidi et al. [16] proposent une approche adaptative et distribuée pour faire face à trois types de panne : nœud virtuel (dégradation de performances), nœud substrat et lien substrat. Dans cette approche dite multi-agents, tout nœud substrat supporte un agent chargé du monitorage, de la supervision et de l'extraction des attributs dynamiques du nœud correspondant et de ses liens adjacents. Les attributs dynamiques d'un nœud correspondent à toutes les informations évoluant dans le temps et nécessaires au processus de calcul d'un mappage d'un réseau virtuel vers un réseau substrat. Ces informations incluent la charge du nœud, son coût, les capacités des liens adjacents, leurs coûts, etc.

Après avoir regroupé les nœuds substrat ayant le plus de similitude dans les mêmes clusters, l'approche multi-agents propose de réagir à une panne d'un nœud virtuel en réessayant de mapper ce nœud en panne sur le même nœud substrat. Si cette procédure échoue, l'approche propose d'étendre la recherche à tous les nœuds substrat appartenant au cluster supportant le nœud en panne. De cette façon, le processus de réparation de la panne est simplifié (échange restreint de

messages) et accéléré (délai de récupération réduit) puisque seuls les nœuds substrat proches et ayant des propriétés similaires au nœud en panne sont consultés. Une fonction de dissimilitude avec le nœud virtuel en panne est calculée pour chaque nœud substrat du cluster afin d'en choisir le meilleur. Cette fonction est assez simple et a pour objectif de déterminer un nœud substrat ayant le plus de similitude dans ses attributs dynamiques avec le nœud virtuel en panne. Le processus de réparation se termine par la migration des liens virtuels du nœud virtuel en panne vers son nœud de remplacement.

L'évaluation de performances montre que cette approche est apte à réparer les pannes en un temps légèrement inférieur à la seconde dans un réseau substrat d'une dizaine de nœuds. Cette approche n'est donc pas applicable dans le contexte d'applications temps réel surtout qu'elle ne garantit pas la disponibilité des ressources après une panne. En effet, les ressources ne sont pas pré-réservées sur les nœuds et liens de secours. De plus, cette approche ne permet pas d'optimiser les ressources d'autant plus que la fonction de dissimilitude combine linéairement les attributs au lieu de les optimiser ensemble.

V. Conclusion

Les réseaux virtuels prennent un rôle de plus en plus important dans l'architecture des réseaux et services. Leur protection en cas de pannes est devenu un problème crucial pour garantir la continuité de service. Ce problème est lié à la fois au problème VNE et au problème de protection des réseaux maillés, qui sont NP-difficiles.

Dans cet article, nous avons présenté et analysé quelques unes des heuristiques que nous estimons représentatives de l'état de l'art. La plupart de ces méthodes adoptent une approche centralisée, alors que le réseau, par nature, est un système distribué et les VN sont très dynamiques. Ceci est compréhensible dans la mesure où la dissémination d'informations est souvent très coûteuse et pas facile à réaliser. L'avènement de SDN (*Software Defined Networks*), qui facilite la gestion centralisée des informations et la mise en place des actions coordonnées, devra permettre une meilleure optimisation des ressources, c'est donc une piste qui mériterait d'être explorée.

Références

[1] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the internet impasse through virtualization," *Computer*, no. 4, pp. 34–41, 2005.

[2] A. Belbekkouche, M. Hasan, and A. Karmouch, "Resource discovery and allocation in network virtualization," *Communications Surveys & Tutorials, IEEE*, vol. 14, no. 4, pp. 1114–1128, 2012.

[3] A. Fischer, J. F. Botero, M. Till Beck, H. De Meer, and X. Hesselbach, "Virtual network embedding : A survey," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 4, pp. 1888–1906, 2013.

[4] M. R. Rahman and R. Boutaba, "Svne : Survivable virtual network embedding algorithms for network virtualization.," *IEEE Transactions on Network and Service Management*, vol. 10, no. 2, pp. 105–118, 2013.

[5] Y. Zhu and M. H. Ammar, "Algorithms for assigning substrate network resources to virtual network components," in *INFOCOM*, vol. 1200, pp. 1–12, 2006.

[6] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding : substrate support for path splitting and migration," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 17–29, 2008.

[7] N. M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual network embedding with coordinated node and link mapping," in *INFOCOM*, pp. 783–791, IEEE, 2009.

[8] X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J. Wang, "Virtual network embedding through topology-aware node ranking," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 2, pp. 38–47, 2011.

[9] G. Sun, H. Yu, V. Anand, and L. Li, "A cost efficient framework and algorithm for embedding dynamic virtual network requests," *Future Generation Computer Systems*, vol. 29, no. 5, pp. 1265–1277, 2013.

[10] W. D. Grover and D. Stamatelakis, "Cycle-oriented distributed preconfiguration : ring-like speed with mesh-like capacity for self-planning network restoration," in *Communications, 1998. ICC 98.*, vol. 1, pp. 537–543, IEEE, 1998.

[11] P. Pan, G. Swallow, and A. Atlas, "Fast reroute extensions to rsvp-te for lsp tunnels," 2005.

[12] L. Csikor, J. Tapolcai, and G. Rétvári, "Optimizing igp link costs for improving ip-level resilience with loop-free alternates," *Computer Communications*, vol. 36, no. 6, pp. 645–655, 2013.

[13] J. Tapolcai and G. Rétvári, "Router virtualization for improving ip-level resilience," in *INFOCOM, 2013 Proceedings IEEE*, pp. 935–943, IEEE, 2013.

[14] K. Murakami and H. S. Kim, "Optimal capacity and flow assignment for self-healing atm networks based on line and end-to-end restoration," *IEEE/ACM Transactions on Networking (TON)*, vol. 6, no. 2, pp. 207–221, 1998.

[15] H. Yu, V. Anand, C. Qiao, and G. Sun, "Cost efficient design of survivable virtual infrastructure to recover from facility node failures," in *Communications (ICC), 2011 IEEE International Conference on*, pp. 1–6, IEEE, 2011.

[16] I. Houidi, W. Louati, D. Zeghlache, P. Papadimitriou, and L. Mathy, "Adaptive virtual network provisioning," in *Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures*, pp. 41–48, ACM, 2010.

[17] B. Guo, C. Qiao, J. Wang, H. Yu, Y. Zuo, J. Li, Z. Chen, and Y. He, "Survivable virtual network design and embedding to survive a facility node failure," *Lightwave Technology, Journal of*, vol. 32, no. 3, pp. 483–493, 2014.

[18] W.-L. Yeow, C. Westphal, and U. C. Kozat, "Designing and embedding reliable virtual infrastructures," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 2, pp. 57–64, 2011.

[19] Y. Yu, C. Shan-zhi, L. Xin, and W. Yan, "Rmap : An algorithm of virtual network resilience mapping," in *Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 7th International Conference on*, pp. 1–4, IEEE, 2011.

[20] T. Guo, N. Wang, K. Moessner, and R. Tafazolli, "Shared backup network provision for virtual network embedding," in *Communications (ICC), 2011 IEEE International Conference on*, pp. 1–5, IEEE, 2011.

[21] H. Yu, V. Anand, and C. Qiao, "Virtual infrastructure design for surviving physical link failures," *The Computer Journal*, 2012.

[22] H. Yu, V. Anand, C. Qiao, and H. Di, "Migration based protection for virtual infrastructure survivability for link failure," in *Optical Fiber Communication Conference*, Optical Society of America, 2011.

[23] X. Zhang, C. Phillips, and X. Chen, "An overlay mapping model for achieving enhanced qos and resilience performance," in *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2011 3rd International Congress on*, pp. 1–7, IEEE, 2011.

[24] O. Soualah, I. Fajjari, N. Aitsaadi, and A. Mellouk, "Pr-vne : Preventive reliable virtual network embedding algorithm in cloud's network," in *Global Communications Conference (GLOBECOM), 2013 IEEE*, pp. 1303–1309, IEEE, 2013.