

Vers des algorithmes efficaces pour la prévention de l'inconsistance entre les tables de routage

Lieu	Équipe Réseaux, ICube (UMR CNRS 7357)
Encadrant	Pascal MERINDOL (merindol@unistra.fr)

Contexte

Avec le développement des applications temps-réel sur Internet, telles que la télévision, la voix sur IP et les jeux en ligne, les fournisseurs d'accès à Internet doivent faire face à des contraintes de plus en plus fortes quant aux performances de leurs services. Ces contraintes se traduisent sous la forme de conventions de service [1] (SLA) et définissent le niveau de service attendu d'un opérateur à travers diverses métriques, comme les pertes de paquets ou la disponibilité du réseau. Actuellement, les interruptions du service sont principalement causées par des changements topologiques, lesquels sont pourtant des événements courants dans les réseaux IP de grande taille. Une part non négligeable de ces changements résulte d'opérations de maintenance et de migration [2]. De telles interventions sont généralement effectuées de nuit, afin de limiter leur impact sur le trafic, bien que cela représente un coût supplémentaire pour l'opérateur. Une autre source de changement topologique peut être la modification intentionnelle des poids de certains liens dans les réseaux internes des opérateurs, afin d'optimiser le routage en fonction des fluctuations du trafic [3]. Enfin, les changements imprévus [4], tels que la panne d'un lien ou d'un routeur, sont également une source importante de problèmes de convergence, mais leur impact peut être limité grâce à des techniques de re-routage rapide [5]. Chacun de ces changements force les routeurs à recalculer leurs tables de routage, faisant ainsi entrer le réseau dans un état transitoire durant lequel des boucles de routage peuvent apparaître. Ce phénomène peut causer une dégradation du service offert, et est donc particulièrement regrettable quand il survient à la suite d'une opération planifiée.

Francois et al. [6, 7] ont notamment proposé un mécanisme garantissant l'absence de boucle transitoire lors de la convergence d'un protocole de routage à état des liens, suite à la modification d'un lien dans le réseau (ajout, suppression ou changement de poids). Cette approche consiste en l'injection d'une séquence de reconfigurations du poids du lien, et ne nécessite donc aucune modification des standards existants. En termes algorithmiques, cette solution se base sur la fusion des graphes de routage avant et après la modification du réseau, afin de détecter les boucles transitoires pouvant potentiellement apparaître lors de la convergence. Dans le cas d'une augmentation du coût d'un lien, une séquence d'incrémentations intermédiaires est ensuite calculée, de sorte que, pour une destination donnée, les routeurs les plus "éloignés" du lien impacté se mettent à jour avant les routeurs plus proches. L'algorithme utilisé fonctionne sur un mode essai-erreur glouton dont l'objectif, à chaque étape, est d'appliquer le plus grand incrément possible. Une extension de cet algorithme pour l'ajout ou de la suppression d'un routeur entier est possible [8]. Néanmoins, il est à noter que cette situation ajoute au problème une nouvelle dimension : il ne s'agit plus maintenant de considérer un seul lien, mais l'ensemble des liens sortants d'un noeud ainsi que les interactions entre ceux-ci. Des modifications indépendantes sur plusieurs liens peuvent entraîner la formation d'un nouveau genre de boucle transitoire, plus délicat à détecter. Naturellement, il serait possible de retirer (ou d'ajouter) un à un chacun de ces liens, de sorte que le problème pourrait se réduire à l'application de k séquences de modification de lien, k étant le degré du noeud concerné, mais dans ce cas la minimalité de la séquence n'est pas garantie. Une autre solution plus élégante consiste à modifier uniformément le poids sur chacun des liens. Il s'agirait alors d'une modification de la distance offerte par le noeud, lequel pourrait être assimilé à un unique lien et traité comme tel. Néanmoins, de telles solutions ne profitent pas pleinement de la liberté offerte par la possibilité de modifier ces poids à la fois simultanément et de manière hétérogène. Elle ne répondent donc pas à un objectif de minimisation du nombre d'étapes intermédiaires. Dès lors, une autre technique, propre à cette situation, a du être envisagée. Il est possible, à partir d'une extraction efficace des contraintes garantissant l'absence de boucle transitoire, de construire une séquence de taille minimale satisfaisant chacune de ces contraintes. Contrairement à la technique utilisée pour la modification d'un

lien, ce nouvel algorithme commencera par calculer les derniers vecteurs à injecter (avec les valeurs les plus élevées). En effet, l'ensemble des contraintes n'exhibant qu'une relation d'ordre partiel, le mode essai-erreur utilisé jusqu'ici ne permet pas de satisfaire efficacement les contraintes. Le but de ce stage est d'évaluer plusieurs solutions algorithmiques efficaces permettant de construire une séquence minimale pour tout type d'opérations (lien et noeud voire davantage).

Sujet

Il a été démontré que le problème de reconfiguration mentionné dans le paragraphe précédent pouvait être résolu en temps polynomial à l'échelle d'un lien comme d'un noeud entier. Cependant, plusieurs sous-problèmes restent ouverts : quelle est la nature du problème à l'échelle d'un ensemble de liens indépendants ? quelle est sa complexité exacte en fonction de chaque sous-objectif ? et, plus modestement, quels sont les algorithmes les plus efficaces (en termes de complexité moyenne et au pire) en fonction du contexte (outil de management ou déploiement sur les routeurs) ? C'est au moins à cette dernière question que devra répondre l'étudiant retenu pour ce stage. En particulier, il réalisera cette étude en s'intéressant au temps de calcul observé sur des graphes de terrains réalistes.

Après s'être documenté en détail sujet, l'étudiant devra fournir les travaux suivants :

- spécification, implémentation et comparaison de plusieurs algorithmes de calcul de séquence de mise à jour ;
- utilisation de générateurs de topologie et de topologies inférées ou réelles pour évaluer en pratique la complexité des algorithmes suscités ;
- émulations et expérimentations des bénéfices de la solution dans un environnement applicatif réaliste et inter-domaine [9] ;
- comparaison avec les alternatives [10] et ajustements des solutions proposées.

Références

- [1] J. Martin and A. Nilsson. On Service Level Agreements for IP Networks. In Proceedings of IEEE INFOCOM'02, volume 2, pages 855 – 863, New York, NY, USA, June 2002.
- [2] Laurent Vanbever, Stefano Vissicchio, Cristel Pelsser, Pierre Francois, and Olivier Bonaventure. Lossless Migrations of Link-State IGPs. IEEE/ACM Transactions on Networking, 20(6) :1842–1855, December 2012.
- [3] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS Weights in a Changing World. IEEE Journal on Selected Areas in Communications, 20(4) :756 –767, May 2002.
- [4] Athina Markopoulou, Gianluca Iannaccone, Supratik Bhattacharyya, Chen-Nee Chuah, Yashar Ganjali, and Christophe Diot. Characterization of Failures in an Operational IP Backbone Network. IEEE/ACM Transactions on Networking, 16 :749–762, August 2008.
- [5] Amund Kvalbein, Audun Fosselie Hansen, Tarik Čičić, Stein Gjessing, and Olav Lysne. Multiple routing configurations for fast ip network recovery. IEEE/ACM Trans. Netw., 17(2) :473–486, April 2009.
- [6] M. Shand and S. Bryant. A Framework for Loop-Free Convergence. RFC 5715, IETF, January 2010.
- [7] Francois Clad, Pascal Merindol, Jean-Jacques Pansiot, Pierre Francois, and Olivier Bonaventure. Graceful Convergence in Link-State IP Networks : A Lightweight Algorithm Ensuring Minimal Operational Impact. IEEE/ACM Transactions on Networking, 2013.
- [8] Francois Clad, Pascal Merindol, Stefano Vissicchio, Jean-Jacques Pansiot, and Pierre Francois. Graceful Router Updates for Link-State Protocols. In IEEE ICNP, 2013.
- [9] Laurent Vanbever, Stefano Vissicchio, Luca Cittadini, and Olivier Bonaventure. When the cure is worse than the disease : the impact of graceful igp operations on bgp. In IEEE INFOCOM, 2013.
- [10] P. Francois and O. Bonaventure. Avoiding Transient Loops During the Convergence of Link-State Routing Protocols. IEEE/ACM Transactions on Networking, 15(6) :1280 –1292, December 2007.