



Cooperative Congestion Control dans NDN

Adrien Thibaud, Julien Fasson, Fabrice Arnal, Renaud Sallantin, Emmanuel Dubois, Emmanuel Chaput

► **To cite this version:**

Adrien Thibaud, Julien Fasson, Fabrice Arnal, Renaud Sallantin, Emmanuel Dubois, et al.. Cooperative Congestion Control dans NDN. Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, Sep 2020, Lyon, France. hal-02696871

HAL Id: hal-02696871

<https://hal.archives-ouvertes.fr/hal-02696871>

Submitted on 1 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Cooperative Congestion Control dans NDN

A. Thibaud¹, J. Fasson¹, F. Arnal², R. Sallantin², E. Dubois³ et E. Chaput¹

¹{adrien.thibaud,jfasson,chaput}@enseeiht.fr, T SA/Universit  de Toulouse; INP; IRIT; F-31017 Toulouse, France

²{fabrice.arnal,renaud.sallantin}@thalesaleniaspace.com, Thales Alenia Space, France

³emmanuel.dubois@cnes.fr, CNES, France

Named Data Networking (NDN) est l'une des architectures d'Information Centric Network (ICN). Pour r cup rer du contenu, il utilise du multi-source, du multi-chemin ainsi que des caches opportunistes sur les routeurs. Ces propri t s fournissent de nouvelles opportunit s pour am liorer la Qualit  d'Exp rience des utilisateurs finaux (QoE). Cependant, la gestion de plusieurs flux, eux-m mes utilisant du multi-chemin, est tr s complexe. L'objectif de notre travail est de proposer un cadre   ce probl me en d finissant trois principes qu'une solution de gestion de ces flux devrait prendre en compte. Les n uds devront coop rer, superviser leurs files d' mission et g rer intelligemment les capacit s de multi-chemins de NDN. Ces trois  l ments sont au c ur de notre proposition : Cooperative Congestion Control (CCC). Plus qu'une solution, CCC est plus une ossature modulaire o  chaque principe peut  tre impl ment  de multiples fa ons. L'objectif final est de r partir  quitablement les flux sur le r seau et de maximiser la QoE des utilisateurs. Nous  valuons CCC par simulation avec ndnSIM, pour ensuite le comparer avec les solutions propos es dans l' tat de l'art.

Mots-cl s : ICN, NDN, Flux multi-chemins, Contr le de Congestion,  quit  max-min, QoE

1 Introduction et Objectifs

L'Information-Centric Network (ICN) est un nouveau paradigme qui red finit la couche r seau. Celle-ci est orient e sur le contenu plut t que sur les h tes comme dans IP. Parmi les architectures ICN, Named Data Networking (NDN) [Zt14] affiche de belles promesses. En effet, elle associe les avantages du protocole HTTP et des r seaux de livraison de contenus (CDN) et pair   pair (P2P) directement dans la couche de niveau 3. C'est une architecture *receiver-driven*, o  l'utilisateur est   l'initiative de la communication et demande explicitement chaque morceau de la donn e (nomm  chunk) qu'il veut recevoir. Comme dans les r seaux P2P, NDN peut r cup rer une donn e depuis plusieurs sources. De plus, NDN g re nativement la r cup ration multi-chemins de ces contenus. Enfin, comme dans les CDN, les n uds peuvent mettre en cache une donn e qu'ils estiment populaire, de mani re opportuniste ou non.

Notre d fi est, ici, de d finir une solution qui g re les flux de donn e multi-chemins de mani re  quitable tout en maximisant l'utilisation du r seau. Ainsi nous maximisons la Qualit  d'Exp rience des utilisateurs (QoE) sans pour autant avantager ou d savantager un type d'utilisateurs. Dans les r seaux IP, deux m thodes duales et antagonistes coexistent : TCP qui a pour but d'augmenter le d bit jusqu'  la congestion et les Active Queue Management (AQM) qui ont pour but de pr venir la congestion en r duisant le d bit. En plus de fournir l'opportunit  de g rer les multi-chemins, NDN permet une coop ration entre les m canismes de bout en bout, tel que TCP, et les m canismes au saut par saut, tel que les AQM. Dans ce papier, nous proposons une nouvelle solution, Cooperative Congestion Control [TFA⁺20] (CCC), qui se base justement sur ces propri t s.

2  tat de l'art

Comme dans les r seaux IP, il existe deux grande familles de solutions pour g rer les congestions : les m thodes bout en bout et les techniques saut par saut. Il existe des solutions hybrides qui sont une combinaison des deux pr c dentes. Interest Control Protocol [CGM12] (ICP) est une adaptation de TCP pour NDN. Du   la caract ristique *receiver-driven* de NDN, la fen tre de congestion est maintenue chez l'utilisateur et compte les Interests en transit dans le r seau. Les paquets de donn e valident les Interest, comme les

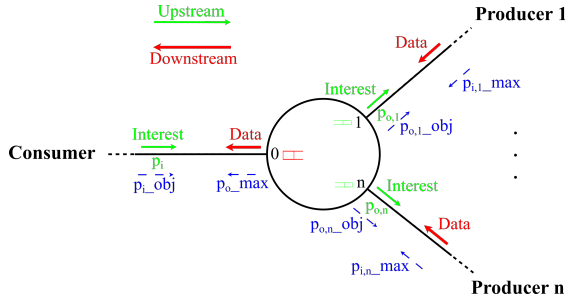


FIGURE 1: Contexte d'un nœud

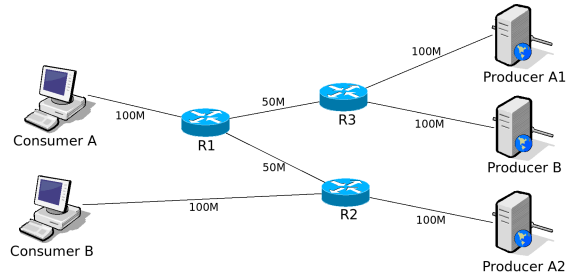


FIGURE 2: Topologie de test

acquitements de TCP. Practical Congestion Control Scheme [SYZZ16] (PCON) est une solution hybride qui allie un mécanisme de bout en bout et un autre au saut par saut. Ce dernier, sur chaque nœud utilise des AQM sur les files d'émission pour détecter la congestion avant qu'une perte ne se produise, marque les paquets et fait du partage de charge. Comme pour ICP, le mécanisme de bout en bout utilise une fenêtre de congestion sur les Interests. Cependant, en plus de réagir lorsqu'une demande expire (Retransmission TimeOut), il réduit la taille de sa fenêtre de congestion aussi lorsqu'il reçoit une notification de congestion. Dans cette étude, nous allons comparer notre solution avec les combinaisons "ICP + PCON-FS", où l'utilisateur utilise ICP de bout en bout et où les nœuds utilisent la partie saut par saut de PCON, et "PCON-CS + DRF", où l'utilisateur utilise PCON de bout en bout et où les nœuds utilisent la *forwarding strategy* Dynamic Request Forwarding [CGM⁺13] (DRF) (cf. une précédente étude [TFA⁺19] pour le choix de ces deux combinaisons). Le but de DRF n'est que de faire de l'équilibrage de charge : un nœud utilisant DRF fait en sorte d'utiliser tous les chemins à sa disposition de la même manière.

3 Cooperative Congestion Control

3.1 Principes

Nous avons conçu Cooperative Congestion Control (CCC) comme une méthode coopérative pour distribuer des flux concurrents sur le réseau. Elle adopte une approche purement saut par saut, gère le rythme d'émission des Interests et vise à éviter la congestion tout en utilisant efficacement les propriétés multi-chemins de NDN. Notre solution repose sur trois principes : (i) une coopération entre les nœuds qui prend la forme d'échange d'objectifs de flux et de contraintes réseaux, (ii) une surveillance des files d'émission pour estimer leur statuts et (iii) une prise de décisions intelligente faite à l'aide des informations des deux premiers principes pour allouer de la bande passante aux différents flux.

3.2 Comportement des nœuds

La Fig. 1 représente un nœud NDN avec un flux. Les Interests arrivent sur ce nœud au rythme p_i dans le sens montant (*Upstream*). Il y a n interfaces possibles pour atteindre un producteur de la donnée et le flux est donc partagé entre ces différents chemins. Les Interests sortent ensuite du nœud au rythme $p_{o,k}$ pour l'interface k . Connaissant la taille du paquet de donnée, CCC est capable d'estimer le débit utilisé par ce flux sur chaque interface. Comme expliqué précédemment, notre premier principe consiste en une coopération entre les nœuds. Les nœuds *Upstream* envoient les objectifs pour chaque flux ($p_{i,obj}$) et les nœuds *Downstream* envoient les contraintes réseaux pour ces mêmes flux ($p_{i,k,max}, \forall k$). L'objectif d'un flux représente le besoin de l'utilisateur et a tendance à faire augmenter le rythme réel, tandis que les contraintes représentent les limitations réseaux et ont tendance à faire diminuer le rythme réel. Le deuxième principe est la surveillance des files d'émission. D'une part, en utilisant une AQM, le nœud est capable de détecter si la file est en passe d'être congestionnée. D'autre part, avec les contraintes sortantes de chacun des flux, il est capable d'estimer le débit utilisé par ceux-ci. Il peut ainsi en déduire la quantité de débit inutilisée sur cette interface. Le dernier principe se fonde sur les deux premiers pour réagir : gestion des informations reçues (objectifs et contraintes) et prise de décision. Pour les objectifs, le nœud doit respecter la règle de conservation de l'objectif : le besoin entrant doit être égal à la somme des besoins sortants (pas

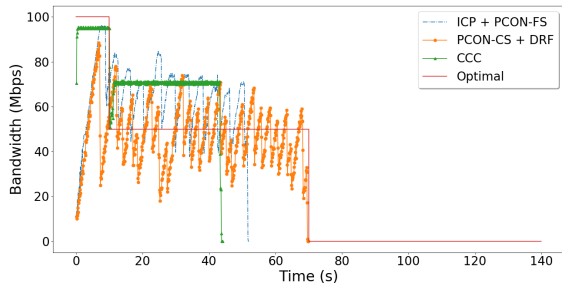


FIGURE 3: Débit du flux A

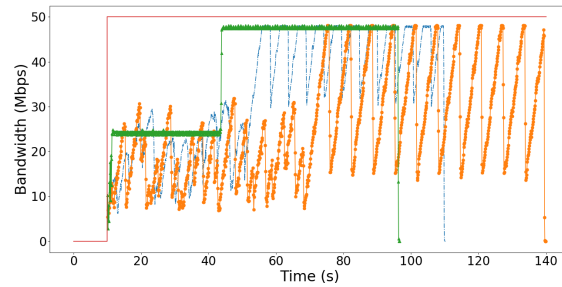


FIGURE 4: Débit du flux B

de création de besoin dans le réseau). Pour les contraintes, le nœud doit agréger les contraintes entrantes dans la contrainte sortante. De plus, quand une congestion est détectée localement, le nœud diminue la contrainte sortante de chaque flux. Si le nœud estime que le lien est sous utilisé il va partager équitablement le débit restant entre les flux actifs (tout en respectant les contraintes entrantes). La contrainte sortante est donc inférieure ou égale à la somme des contraintes entrantes. Finalement, le nœud décide de comment faire l'équilibrage de charge. Pour cela, il doit seulement respecter le fait que le rythme d'émission réel des Interests soit inférieur ou égal à la contrainte reçue pour chacune des interfaces. Le détail complet des implémentations de ces principes sont disponibles dans le papier d'origine [TFA⁺20].

4 Évaluations

Nous avons implémenté CCC dans le simulateur ndnSIM [Mt16]. Notre code comme l'environnement de test sont mis à disposition*. La topologie utilisée pour les tests est celle présentée sur la Fig. 2. Les deux utilisateurs ont des demandes supérieures aux capacités du réseau. L'utilisateur A commence ses requêtes à partir de $t = 0s$, tandis que l'utilisateur B commence à $t = 10s$. Les Fig. 3 et 4 montrent les débits obtenus par les utilisateurs A et B, respectivement. La courbe rouge montre la distribution équitable optimale (en prenant en compte tout le réseau et en utilisant le critère de *max-min fairness*). Pendant la période où le flux A est seul, il peut théoriquement obtenir 100Mbps de débit grâce aux deux chemins disponibles. Les deux combinaisons "ICP + PCON-FS" et "PCON-CS + DRF" convergent lentement mais arrivent finalement presque à cette valeur. CCC est lui capable de l'atteindre très rapidement. Lorsque le flux B commence et durant la phase de cohabitation, les deux flux utilisent le lien R1-R3 de notre topologie. Le flux A reste seul sur son second chemin. La courbe optimale indique que les deux flux doivent obtenir 50Mbps chacun. Pour les trois solutions, le débit du flux A diminue effectivement quand le flux B commence mais il reste supérieur aux 50Mbps. En conséquence, le débit du flux B reste lui inférieur à ces mêmes 50Mbps. Ceci s'explique par le fait que ces trois solutions assurent une équité au niveau du lien : le lien partagé (R1-R3) est équitablement partagé entre les flux actifs, soit environ 25Mbps chacun. Toutefois, dans ce cas-là, le flux A dispose d'un autre chemin et son débit total atteint environ 75Mbps. Ce type d'équité est un bon point de départ et notre solution fait aussi bien que les solutions existantes sur ce point. Cependant, dans le futur, nous aimerions atteindre une équité globale, comme celle de la courbe optimale. Les courbes de débits des combinaisons "ICP + PCON-FS" et "PCON-CS + DRF" montrent le même type d'oscillations que TCP. Ce phénomène est directement lié à la gestion de la fenêtre de congestion qui va diminuer drastiquement lorsqu'une congestion est détectée. À l'inverse, notre solution est très stable et est capable de rapidement converger lorsqu'un nouveau flux commence ou termine. Cela est du au fait que CCC ne va pas augmenter de manière aveugle le débit des flux et va assigner les contraintes telles qu'aucune congestion ne peut se produire. En effet, contrairement à une solution de bout en bout, chacun des nœuds connaît le débit qu'il peut atteindre et la capacité dont il dispose. Le seul cas où une congestion apparaît est lorsqu'un nouveau flux commence, mais celle-ci est détectée localement à l'aide des AQM et les valeurs des contraintes sont réduites pour résoudre ce problème. Lorsque le flux A se termine, CCC est capable de rapidement ré-allouer la bande passante pour que le flux B utilise ensuite la totalité de la capacité du chemin. Pour les deux autres

*. <https://gitlab.tesa.prd.fr/athibaud/ccc-codebase>

solutions, avec fenêtre de congestion, l'augmentation est progressive et le phénomène oscillatoire reste présent. Avec CCC, les utilisateurs sont donc capables de récupérer leur contenu en un temps réduit et d'avoir un débit global plus important.

5 Conclusions

Dans ce papier, nous avons décrit le problème d'une distribution multi-flux sur une topologie multi-chemins. Nous avons ensuite proposé une solution, Cooperative Congestion Control, qui, basée sur NDN, s'attaque à ce problème. Elle fonctionne de manière distribuée : chaque nœud du réseau, consommateurs et producteurs inclus, vont s'échanger des objectifs et des contraintes, mesurer l'utilisation locale de leurs files d'émission et utiliser toutes ces informations pour allouer équitablement leur bande passante entre les différents flux actifs. L'implémentation de ces trois principes est très modulable. Dans le papier original, nous avons proposé un découpage en six algorithmes différents avec une implémentation de base pour chacun d'entre eux. Cette première implémentation prouve que la conception de CCC fonctionne. Notre solution est capable d'utiliser le réseau au maximum de sa capacité, diminuant ainsi le temps de récupération global des contenus par nos utilisateurs. La surveillance local des files d'émission permet de détecter rapidement lorsqu'une congestion arrive ou lorsqu'un surplus de débit est disponible. Le temps de réaction est par conséquent court. Comme pour les solutions de l'état de l'art, CCC ne garantit qu'une équité de lien. C'est-à-dire que les flux en compétition sur un lien vont recevoir une part égale du débit de ce lien. Comme travaux futurs, nous prévoyons d'étudier l'impacte de différentes implémentations pour nos algorithmes afin de tendre vers une équité plus proche d'une *max-min fairness* globale.

Références

- [CGM12] G. Carofiglio, M. Gallo, and L. Muscariello. Icp : Design and evaluation of an interest control protocol for content-centric networking. In *2012 Proceedings IEEE INFOCOM Workshops*, pages 304–309, March 2012.
- [CGM⁺13] G. Carofiglio, M. Gallo, L. Muscariello, M. Papalini, and S. Wang. Optimal multipath congestion control and request forwarding in information-centric networks. In *2013 21st IEEE International Conference on Network Protocols (ICNP)*, pages 1–10, Oct 2013.
- [Mt16] S. Mastorakis and *al.* ndnSIM 2 : An updated NDN simulator for NS-3. Technical Report NDN-0028, Revision 2, NDN, November 2016.
- [SYZZ16] K. Schneider, C. Yi, B. Zhang, and L. Zhang. A practical congestion control scheme for named data networking. In *Proceedings of the 3rd ACM Conference on Information-Centric Networking, ACM-ICN '16*, pages 21–30, New York, NY, USA, 2016. ACM.
- [TFA⁺19] Adrien Thibaud, Julien Fasson, Fabrice Arnal, Renaud Sallantin, Emmanuel Dubois, and Emmanuel Chaput. An analysis of NDN Congestion Control challenges. In *International Conference on Hot Information-Centric Networking (HotICN 2019)*, Chongqing, China, December 2019.
- [TFA⁺20] Adrien Thibaud, Julien Fasson, Fabrice Arnal, Renaud Sallantin, Emmanuel Dubois, and Emmanuel Chaput. Cooperative Congestion Control in NDN. In *2020 IEEE International Conference on Communications (ICC) : Next-Generation Networking and Internet Symposium (IEEE ICC'20 - NGNI Symposium)*, Dublin, Ireland, June 2020.
- [Zt14] L. Zhan and *al.* Named data networking. *SIGCOMM Comput. Commun. Rev.*, 44(3) :66–73, July 2014.