

# Un algorithme de routage robuste basé sur l'optimisation vers des réseaux économes en énergie pour le cloud computing

Dingde Jiang·Zhengzheng Xu·Jindi Liu·  
Wenhui Zhao

© Springer Science+Business Media New York 2015

**Abstrait**Cet article étudie le problème de routage dans les réseaux économes en énergie pour le cloud computing. Nous proposons un algorithme de routage robuste pour atteindre la meilleure efficacité énergétique du réseau, qui est basé sur un problème d'optimisation. Pour atteindre le routage hautement économe en énergie dans les réseaux écoénergétiques pour le cloud computing, le lien de faible utilisation est transformé en état de veille pour économiser l'énergie du réseau. Dans le même temps, le trafic de liaison faible est agrégé au lien avec une utilisation élevée pour améliorer l'utilisation du lien et pour endormir les liens autant que possible. Nous présentons une méthode optimisée de lien dormant pour maximiser le nombre de liens dormants. En ciblant la robustesse du réseau, une stratégie adaptative de poids est proposée pour réduire la congestion de la liaison et améliorer la robustesse du réseau.

**Mots clés**Cloud computing·Réseaux économes en énergie· La modélisation·Mesure de réseau·Routage

## 1. Introduction

Selon les statistiques de l'Agence internationale de l'énergie (AIE), la consommation mondiale d'électricité des TIC représentait 8 % de la consommation mondiale d'énergie, et la tendance est à la hausse [1]. En particulier, le trafic réseau augmente de façon exponentielle et contient des fonctionnalités beaucoup plus complexes qu'auparavant [2,3]. Cela conduit les entreprises de télécommunications et les fournisseurs de services Internet à accorder plus d'attention au problème de l'efficacité énergétique des réseaux câblés. La hausse des prix de l'énergie et le nombre croissant d'utilisateurs ont intensifié l'urgence de la recherche sur les questions d'efficacité énergétique, en particulier pour les réseaux de cloud computing. Les réseaux de cloud computing sont basés sur les infrastructures de réseau actuelles. Cependant, d'autres études montrent que la méthode de gestion de l'énergie des routeurs et des équipements de liaison est déraisonnable dans le réseau actuel. La consommation énergétique de ces appareils représente environ un quart de la consommation énergétique totale d'Internet. Par conséquent, le sujet de l'efficacité énergétique des réseaux de cloud computing est devenu un sujet brûlant dans la communauté actuelle des réseaux.

Ces dernières années, de nombreux chercheurs ont étudié l'efficacité énergétique des réseaux sous différents aspects. Bolla et al. ont proposé une stratégie d'optimisation de la consommation électrique des équipements en détectant la charge de trafic dans le réseau hétérogène [4]. Cianfrani et al. a présenté une méthode de gestion de l'efficacité énergétique pour le réseau IP [5], qui a utilisé l'algorithme de routage OSPF pour baliser l'état du lien, puis pour fermer le lien avec une faible charge de trafic afin d'améliorer l'efficacité énergétique. Les deux méthodes envisageaient de transférer le trafic de liaison des liaisons à faible utilisation vers les liaisons à forte utilisation, afin de réduire la consommation d'énergie dans l'ensemble du réseau lorsqu'il répondait à la solidité de la charge du réseau, mais pas d'analyse et

---

D. Jiang (✉)·Z. Xu·J. Liu·W. Zhao College of Information Science and Engineering, Northeastern University, Shenyang 110819, Chine Courriel : jiangdd@mail.neu.edu.cn

Z. Xu  
École d'économie et de gestion, Université normale d'AnQing, AnQing 246011, Chine

recherche spécifiquement sur l'efficacité énergétique. Restrepo et al. ont proposé un algorithme de gestion de l'énergie dans l'ingénierie du trafic, fermant dynamiquement les liaisons à faible utilisation pour réduire la demande d'énergie [6]. Chiaraviglio et al. ont proposé une approche heuristique pour calculer le nombre minimum de liens qui peuvent être endormis pour une matrice de trafic connue [7]. Ils ont utilisé les informations de trafic pour minimiser la consommation d'énergie du réseau [8]. Cianfrani et al. a proposé un schéma de routage économe en énergie basé sur le protocole de routage OSPF [9]. Cependant, les réseaux de cloud computing supportent une charge plus lourde et plus de trafic en rafale. Cela conduit au fait que les utilisations de liens en eux sont déséquilibrées. Certains liens dans les réseaux de cloud computing supportent des charges plus lourdes tandis que d'autres liens sont suffisamment exploités. De plus, le problème d'efficacité énergétique des réseaux [dix, 11], le caractère social des réseaux [12], et les propriétés dynamiques du trafic réseau [13] ajoutent la difficulté d'effectuer le cloud computing efficace. La mobilité des utilisateurs de terminaux [14] et les comportements anormaux des périphériques réseau [15] peut également conduire à l'échec des tâches de cloud computing. Dans un tel cas, ces méthodes dans les études actuelles sont très difficiles à fournir le routage efficace et robuste pour améliorer l'efficacité énergétique des réseaux de cloud computing.

Dans cet article, un algorithme de routage robuste et économe en énergie, appelé algorithme de routage robuste basé sur l'optimisation (ORRA), est proposé pour résoudre le problème de stabilité et d'efficacité énergétique dans le réseau informatique en nuage actuel. Afin de répondre aux demandes de QoS, les réseaux ISP typiques utilisent généralement la conception redondante pour faire face aux situations inattendues de congestion du réseau et d'échec de routage. Bien qu'une telle conception puisse résoudre les problèmes de QoS, il en résulte un problème de faible efficacité énergétique. Contrairement aux méthodes précédentes, nous examinons comment concevoir un algorithme de routage pour répondre aux exigences de routage et de QoS ainsi que pour obtenir un système hautement économe en énergie et réseau de cloud computing robuste. Nous convertissons le lien de faible utilisation en état de veille pour économiser l'énergie du réseau de cloud computing. Dans ce cas, le routage hautement économe en énergie pour les réseaux de cloud computing peut être atteint. Simultanément, le trafic de liaison faible est agrégé au lien avec une utilisation élevée pour améliorer l'utilisation du lien et pour endormir les liens autant que possible. Pour résoudre ce problème, une méthode de sommeil de liaison optimisée est proposée pour maximiser le nombre de liens dormants dans le réseau informatique en nuage. Et nous proposons une stratégie adaptative de poids pour réduire la congestion des liens et assurer la robustesse du réseau pour le cloud computing. Les résultats de la simulation indiquent que l'algorithme proposé est efficace et réalisable pour le réseau informatique en nuage économe en énergie. Le trafic de liaison faible est agrégé au lien avec une utilisation élevée pour améliorer l'utilisation du lien et pour endormir les liens autant que possible. Pour résoudre ce problème, une méthode de sommeil de liaison optimisée est proposée pour maximiser le nombre de liens dormants dans le réseau informatique en nuage. Et nous proposons une stratégie adaptative de poids pour réduire la congestion des liens et assurer la robustesse du réseau pour le cloud computing. Les résultats de la simulation indiquent que l'algorithme proposé est efficace et réalisable pour le réseau informatique en nuage économe en énergie. Le trafic de liaison faible est agrégé au lien avec une utilisation élevée pour améliorer l'utilisation du lien et pour endormir les liens autant que possible. Pour résoudre ce problème, une méthode de sommeil de liaison optimisée est proposée pour maximiser le nombre de liens dormants dans le réseau informatique en nuage. Et nous proposons une stratégie adaptative de poids pour réduire la congestion des liens et assurer la robustesse du réseau pour le cloud computing. Les résultats de la simulation indiquent que l'algorithme proposé est efficace et réalisable pour le réseau informatique en nuage économe en énergie.

Le reste de cet article est organisé comme suit. Nous effectuons l'énoncé du problème dans la Sect. 2. Section 3 discute et dérive de notre méthode de routage robuste pour le réseau de cloud computing économe en énergie. Section 4 présente les résultats et l'analyse de la simulation. Nous terminons ensuite notre travail dans la Sect. 5.

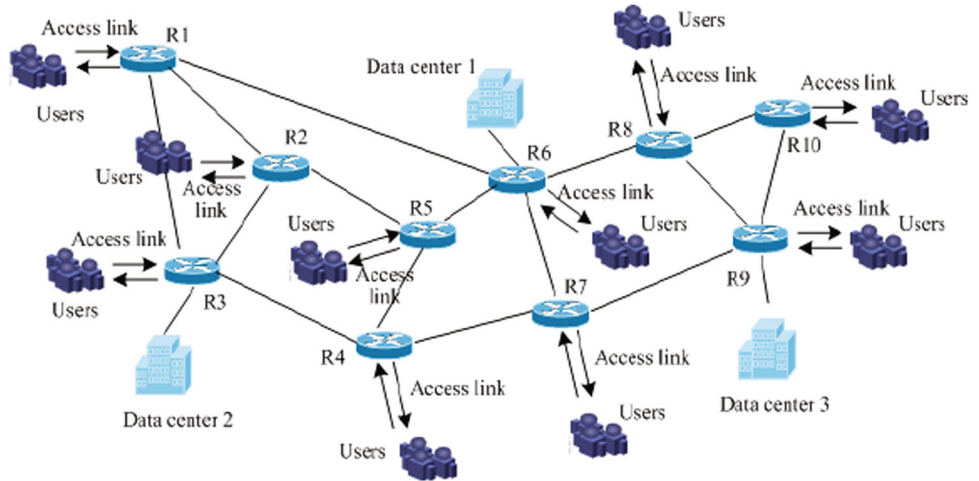
## 2 Énoncé du problème

Ici, nous étudions le problème d'efficacité énergétique et de routage robuste dans le réseau fédérateur de cloud computing qui connecte différents centres de données et utilisateurs. Chiffre 1 trace le modèle de réseau de cloud computing, où les utilisateurs incluent des utilisateurs individuels et des utilisateurs d'entreprise, ils sont connectés à des réseaux de base de cloud computing via des liens d'accès pour visiter des centres de données situés à différents emplacements géographiques. En outre, les migrations de types de ressources virtuelles (telles que les machines virtuelles, les processeurs virtuels, les mémoires virtuelles, les services virtuels, etc.) entre différents centres de données ont été réalisées via le réseau fédérateur de cloud computing de la Fig. 1. Les demandes de service des utilisateurs pour le cloud computing et les migrations de ressources virtuelles maintiennent la dynamique importante dans le temps. Ainsi, dans les réseaux fédérateurs de cloud computing indiqués à la Fig. 1, l'utilisation de chaque lien est déséquilibrée. Dans un tel cas, certains liens du réseau informatique en nuage sont probablement très sollicités et deviennent même congestionnés, mais d'autres liens sont éventuellement utilisés avec une efficacité moindre. Par conséquent, les périphériques réseau peuvent consommer une grande quantité d'énergie pour maintenir le fonctionnement normal du réseau de cloud computing. En conséquence, l'ensemble du réseau fonctionne de manière peu économe en énergie. Pour le modèle de réseau de cloud computing, il est très important de trouver une méthode de routage pour mettre en œuvre la communication robuste et hautement économe en énergie.

De la Fig. 1, nous pouvons extraire la topologie du réseau de base du cloud computing et l'instantané d'utilisation des liens illustré à la Fig. 2, où différentes largeurs de ligne indiquent l'utilisation de lien observée dans un certain instantané. À partir de la topologie de réseau pour le cloud computing illustrée à la Fig. 2, nous pouvons voir que les liens  $R1-R6$ ,  $R2-R3$ , et  $R7-R9$  détiennent l'utilité supérieure tandis que d'autres liens ne sont pas suffisamment exploités. Surtout pour les liens  $R1-R2$ ,  $R2-R5$ ,  $R4-R5$ ,  $R5-R6$ ,  $R4-R7$ ,  $R6-R7$ , et  $R8-R10$ , ils ont l'utilisation beaucoup plus faible dans une certaine observation. Les études actuelles montrent que le cas tel qu'indiqué dans la Fig. 2 existe pratiquement. De plus, pour les liaisons à faible utilisation, elles consomment toujours une grande quantité d'énergie. Pour assurer des communications fiables, les réseaux informatiques en nuage actuels adoptent tous la conception redondante. En conséquence, cette situation d'efficacité énergétique inférieure devient beaucoup plus grave. Par conséquent, comment augmenter de manière appropriée l'efficacité de chaque lien et réduire leur consommation d'énergie est très important.

À cette fin, nous pouvons exprimer le réseau illustré à la Fig. 2 comme modèle graphique  $G(V, E, O)$ , où  $V$  est l'ensemble des nœuds du réseau,  $E$  est l'ensemble des liens du réseau, et  $O$  représente le poids des liens. Laisser  $Net$  être les cardinalités de  $V$  ( $N = |V|$ ) et  $E$  ( $L = |E|$ ), représentant le nombre de nœuds et de liens dans le réseau, respectivement. Dans le présent article, nous nous concentrons uniquement sur les liens redondants dormants, mais nous n'envisageons pas de fermer les nœuds. Et ainsi nous définissons l'efficacité énergétique de l'ensemble du réseau comme la

**Fig. 1** Modèle de réseau informatique en nuage reliant différents centres de données et utilisateurs d'accès



transmettre sans problème des bits d'information sur la consommation d'énergie par joule, à savoir

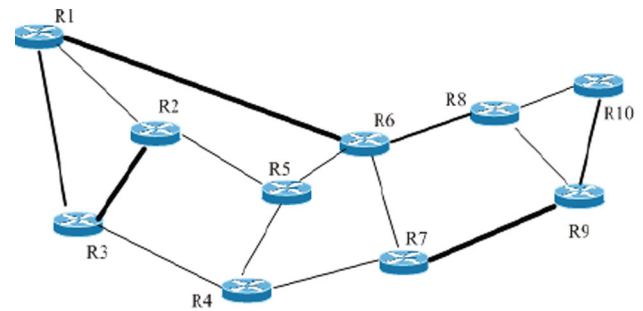
$$\eta = \frac{a}{E_j} = \frac{\sum_{N_s \in V} \sum_{N_d \in V} \int_{T_{ij}} P(d, t) dt}{\sum_{L_{ij} \in E} \sum_{N_s \in V} \sum_{N_d \in V} \int_{T_{ij}} P(d, t) dt} \quad (1)$$

où  $a$  désigne le débit total des réseaux,  $E_j$  représente la consommation énergétique totale des réseaux,  $P_{ij}(t)$  représente la puissance instantanée du lien  $L_{ij}$ ,  $h_{Dakota du Sud}(t)$  est le flux instantané du routeur  $N_s \in V$  pour  $N_d \in V$ ,  $P_{ij}$  indique la puissance moyenne du lien  $L_{ij}$  au cours de la période  $t$ ,  $h_{Dakota du Sud}$  signifie le débit moyen du routeur d'accès  $N_s$  pour  $N_d$  au cours de la période  $t$ , et  $\rho$  (qui est généralement défini comme une valeur constante) exprime la puissance des nœuds du réseau.

Équation (1) formule que l'efficacité énergétique du réseau est principalement associée à la quantité d'informations transmises et à la consommation d'énergie du réseau. Lors de la transmission de la même quantité d'informations dans une unité de temps, plus la consommation d'énergie du réseau est faible, plus l'efficacité énergétique du réseau est élevée, et vice versa. Ici, nous prenons en considération la réduction de la consommation d'énergie du réseau pour l'envoi d'informations sur l'unité afin d'augmenter l'efficacité énergétique du réseau. Dans ce qui suit, nous présenterons un algorithme de mise en veille de lien optimisé pour mettre en veille les liens du réseau autant que possible sous la condition préalable d'assurer une communication normale, afin d'améliorer l'efficacité énergétique de l'ensemble du réseau.

### 3 Routage robuste pour les réseaux de cloud computing

Dans cette section, nous allons dériver notre algorithme de routage robuste optimal (ORRA) pour les réseaux de cloud computing. **Chiffre 3** désigne le schéma fonctionnel de l'algorithme ORRA proposé dans cet article. L'algorithme ORRA proposé comprend sys-



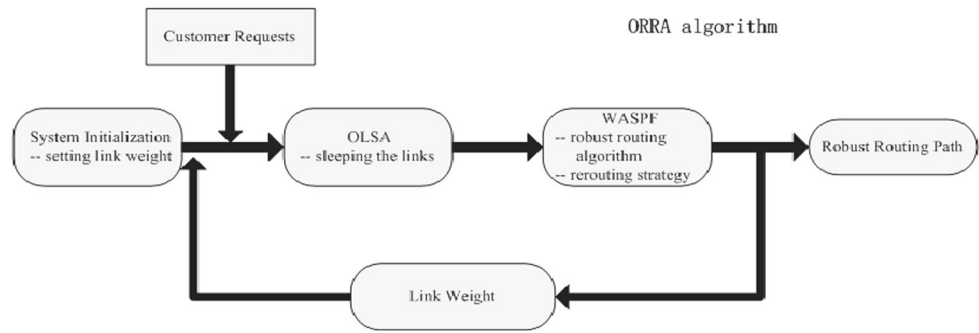
**Figure 2** Topologie du réseau fédérateur de cloud computing et instantané de l'utilisation des liens

module d'initialisation tem, module OLSA (Optimization Link Sleeping Algorithm), module WASPF (Weighted Adaptive Shortest Path First), module de rétroaction sur les pondérations des liens, module de sortie de chemin de routage robuste et interface de demande utilisateur. Nous pouvons voir sur la Fig. 2, lorsque l'utilisateur envoie une requête, les paramètres obtenus du module d'initialisation du système seraient transmis au module OLSA pour endormir les liens redondants ; puis les nouvelles informations de topologie du réseau ont été transmises au module WASPF, en utilisant l'algorithme ORRA pour acheminer le flux de trafic, en produisant des paramètres de chemin de routage et de poids de liaison robustes. Les paramètres de poids de lien seraient utilisés comme paramètres de rétroaction pour les demandes des utilisateurs de la prochaine fois. Les paramètres d'initialisation du système sont obtenus en analysant les données historiques, qui ont été définies de manière appropriée en fonction des conditions réelles du réseau. Dans ce qui suit, nous allons discuter en détail de l'algorithme ORRA.

#### 3.1 Mise en veille optimale des liens

Ici, nous proposons notre OLSA pour laisser dormir les liens de manière appropriée. Pour réduire le nombre de liens actifs, nous exploitons la *moisve* pour exprimer un processus de révision de lien basé sur le mécanisme des degrés d'exportation [9]. En trouvant un nœud avec le maximum

**Figure 3** Schéma fonctionnel de l'algorithme ORRA



degré en tant qu'exportateur le routeur (ER) dans le réseau et recherche le nœud avec le degré maximum dans les nœuds voisins de ER en tant qu'importateur le routeur (IR), un *déplacer* processus est celui du changement de sens de liaison de IR à ER. Calculez l'arbre du plus court chemin (SPT) de ER et IR, respectivement, puis comparez  $SPT(IR)$  et  $SPT(ER)$  pour générer l'arbre de chemin modifié (MPT). En comparant le MPT et le SPT d'IR, nous pouvons dormir les liens redondants. Comme mentionné dans [9], le nombre de liens redondants dormants peut être défini comme le poids du déplacement, à savoir

$$k = \{k_{je}, 1 \leq j \leq L\}. \quad (2)$$

Ici, nous utilisons l'idée de compatibilité de déplacement pour trouver l'ensemble de liens dormants minimum. La compatibilité de déplacement est définie comme [9] : étant donné quelques  $m_1(je_1, e_1)$  et  $m_2(je_2, e_2)$ , si la *bouge moi* peut encore être effectuée après l'exécution de la *bouge moi*, le *mois*  $m_2$  est dit compatible avec la *bouge moi*. La relation de compatibilité est symétrique, c'est-à-dire que si  $m_1$  est compatible avec  $m_2$ , donc  $m_2$  et  $m_1$  sont compatibles. Ensuite, nous nous concentrerons sur la manière de maximiser le nombre de liens dormants dans le réseau. Le problème peut être transformé en un problème de sac à dos typique, qui peut obtenir la solution optimale par un algorithme glouton heuristique. La fonction objectif et les contraintes sont exprimées comme suit :

$$\begin{cases} \text{maximum} & \sum_{je=1}^L k_{je} m_{je}, \\ \text{St} & m_{je} + m_{ji} \leq 1, \quad \forall (je, j) \in C \\ & m_{je} \in \{0, 1\}, \quad \forall je \in M, \end{cases} \quad (3)$$

où C est la relation de compatibilité de  $m_{je}$  et  $m_{ji}$ , M représente l'ensemble de tous les coups possibles, et  $k_{je}$  exprime le poids du déplacement du lien. Le lien déplace le poids  $k_{je}$  peut être obtenu par l'Eq. (2), puis nous pouvons maximiser l'ensemble des liens dormants. Équations (2,3) montrent notre algorithme de sommeil de lien optimisé (pour court OLS).

### 3.2 Plus court chemin adaptatif pondéré en premier

dans cette sous-section, nous discutons de notre WASPF. Pour réaliser le routage robuste, nous proposons maintenant une stratégie avec un poids de lien

routage robuste adaptatif, qui peut modifier dynamiquement les poids des liens du réseau. Selon cette stratégie, nous pouvons modifier la topologie de routage du réseau, réaliser l'équilibrage de charge et améliorer la robustesse du réseau. Cette stratégie mesure la robustesse des réseaux de communication à l'aide de la criticité du réseau qui peut être directement mappée sur la robustesse du réseau. Plus la criticité du réseau est petite, plus le réseau est robuste, et vice versa. Comme discuté dans [16], la criticité du réseau peut être défini comme :

$$\tau = \sum_{je} \sum_j \tau_{ij} = 2n \sum_{je} \frac{j_{je}}{Tr(L)}, \quad (4)$$

où  $n$  est le nombre de nœuds du réseau,  $\tau_{ij}$  dénote la criticité du lien  $je$ ,  $L^+$  est l'inverse de la matrice de Moore-Penrose transformée du Laplacien  $L$ ,  $L^+ = [l_{je}^+]$ ,  $j_{je}$  est le diagonal élément finaux dans la matrice de transformation inverse, et  $Tr(\cdot)$  représente l'opération cumulative.

En normalisant la criticité du réseau  $\tau$  dans l'éq. peut (4), nous obtenir l'équation ci-dessous:

$$\tau = \frac{2}{n-1} Tr(L^+), \quad (5)$$

où  $\tau$  désigne la criticité normalisée du réseau. La capacité de la bande passante du lien est liée à la criticité. Lorsque la capacité de la bande passante du lien est modifiée, la criticité du lien est également modifiée [16]. Pour le lien à forte charge de trafic, la capacité de bande passante du lien disponible est relativement faible, donc le risque de choisir ce lien à router est plus grand. C'est-à-dire que les performances de ce lien deviennent le goulot d'étranglement de l'ensemble du réseau. Ainsi, nous pouvons faire correspondre la criticité du lien à la valeur du poids du lien, de sorte que le lien réseau puisse être utilisé de manière uniforme pour réduire la fluctuation du taux d'utilisation des liens. Lorsque la capacité de la bande passante du lien dans le réseau diminue  $c_{ij}$ , alors la criticité du réseau changera de  $\tau$ . Mettre à jour la valeur du poids du lien  $w_{ij}$  par la formule suivante :

$$\begin{cases} \tau = n(n-1)\tau, \\ \tau = \tau - \tau_0, \\ w_{ij} = \alpha n \tau + w_{0ij}, \end{cases} \quad (6)$$

où  $\eta$  est la criticité initiale dans un intervalle de temps  $J$ ,  $\eta$  est la criticité du réseau pour la fin de  $J$ ,  $un$  est une constante qui a été fixée en fonction de la situation réelle du réseau de simulation,  $n$  est le nombre de nœuds du réseau,  $w_0$  est la valeur initiale du poids du lien de l'intervalle de temps  $J$ , et  $w_j$  est la valeur du poids du lien pour la fin de l'intervalle de temps  $J$ .

Nous proposons une stratégie de routage adaptative robuste comme suit :

- Trouver la matrice de pondération des liens  $O$  du réseau selon Eq. (6);
- Construire le meilleur itinéraire de routage basé sur  $O$  par l'algorithme du plus court chemin de Dijkstra.

### 3.3 Algorithme ORRA

Nous discutons maintenant de notre routage robuste à haute efficacité énergétique pour les réseaux de cloud computing, à savoir ORRA. Comme décrit dans [9], afin de mettre en avant notre algorithme ORRA, nous considérons deux cas. Dans le premier cas, nous ignorons la matrice de trafic et les exigences de QoS, appelée ORRA simplifiée. Dans le second cas, nous considérons la matrice de trafic et les exigences de QoS, appelée ORRA avancée. L'ORRA simplifié peut être exprimé comme le MCP classique [17]. L'ORRA avancé peut être résolu par l'algorithme glouton heuristique basé sur le problème du sac à dos 0-1 [18]. En outre,

Pour l'ORRA simplifié, parce qu'il ignore la matrice de trafic et les exigences de QoS, il suffit de maximiser l'efficacité énergétique du réseau sous la prémisses de réaliser une communication réseau normale et de s'assurer que le réseau a une bonne robustesse. Comment maximiser l'ensemble dormant de liens redondants et trouver la relation entre la criticité du réseau et l'efficacité énergétique de l'ensemble du réseau est le problème clé de l'ORRA simplifié.

Étant donné un modèle de réseau  $G(V, E, O)$ , le taux d'utilisation des liens est défini comme :

$$tu_{ij} = \frac{F_{ij}}{C_{ij}} \quad (7)$$

où  $tu_{ij}$  désigne le taux d'utilisation des liens de  $L_{ij}$ ,  $C_{ij}$  présente la capacité du lien  $L_{ij}$ , et  $F_{ij}$  indique le débit moyen du lien  $L_{ij}$  au cours de la période  $J$ .

Laissiez  $tu$  être la matrice de variation du lien réseau, après un certain nombre de demandes d'utilisateurs dans  $J$ ,  $tu_0$  et  $tu_j$  sont la valeur initiale et la valeur finale de la période de temps  $J$ , respectivement). L'équation suivante peut être obtenue :

$$tu_j = tu_0 + tu_{ij}, L_{ij} \in E \quad (8)$$

Selon le problème d'optimisation décrit par Eq. (3), nous pouvons obtenir l'ensemble maximal des liens dormants. Cependant, après avoir mis en sommeil les liens de redondance, les performances du réseau seront plus ou moins affectées. Nous devons donc trouver un algorithme de routage adapté pour assurer la robustesse optimale

des performances du réseau. Équation (6) reflète la relation entre la robustesse du réseau et le poids du lien. Nous utilisons la criticité du réseau pour mesurer la stabilité du réseau, qui peut calculer le poids de liaison du réseau à un état de performance stable. Et puis nous exploitons le poids du lien comme entrée de l'algorithme de Dijkstra pour obtenir un chemin de routage robuste. De plus, nous employons également l'idée de reroutage pour rediriger la demande vers un autre lien lorsque la capacité du lien ne peut pas répondre à la demande de l'utilisateur. Ainsi, la probabilité de refus du réseau peut être réduite. Grâce à l'analyse ci-dessus, le problème ORRA simplifié peut être exprimé par l'équation ci-dessous :

$$\begin{cases} \text{maximum} & \eta^- \\ \text{St} & \\ & \forall N_s, N_d \in V \& \forall \overline{h_{Dakota du Sud}} > 0, \text{Connecte-moi sur}(N_s, N_d) \\ & L_{ij} \in E, \quad = 1 \quad c(f_{ij}) = w \sum_{N_s \in V} \overline{h_{Dakota du Sud}} \\ & \forall L_{ij} \in E, \quad \overline{F_{ij}} = \sum_{N_s \in V} \overline{h_{Dakota du Sud}} \\ & 0 \leq tu \leq 1, \end{cases} \quad (9)$$

où  $\overline{h_{Dakota du Sud}}$  représente le débit moyen du routeur d'accès  $N_s$  pour  $N_d$  au cours de la période  $J$ ,  $\overline{h_{Dakota du Sud}}$  désigne le débit moyen allant par lien  $L_{ij}$ , et  $c(f_{ij})$  est le coût du lien pour fournir la charge de trafic du lien  $F_{ij}$ .

Afin d'obtenir le chemin de routage réseau robuste optimal et de maximiser l'efficacité énergétique du réseau, nous devons déterminer une série de paramètres. OLSA basé sur l'éq. (3) est relativement simple dans la définition des paramètres par rapport à d'autres mécanismes de mise en veille de liaison. C'est une convergence automatique, et non sensible aux paramètres d'initialisation. Il est important de définir un intervalle de temps approprié, tout en respectant les contraintes de limite de capacité de liaison et de nombre de requêtes utilisateurs. Le réglage de  $de$  est simple dans le processus de simulation proprement dit, car l'impact de  $de$  peut être compensée dans le processus de calcul de l'efficacité énergétique dans le réseau de cloud computing.

Le réglage de la capacité de liaison affectera également les performances de l'algorithme. Dans cet article, nous définissons une relation linéaire entre la capacité du lien et la valeur initiale du poids du lien, à savoir  $c_{ij} = k \cdot w_{0ij} + un$  ( $k$  et  $un$  sont constants). Par conséquent, nous pouvons définir la capacité du lien en fonction du poids du lien, juste en termes de  $ket un$ . Pour définir la valeur initiale du poids du lien, nous utilisons les informations de la matrice de trafic historique du lien pour déterminer l'état de chaque lien, puis calculons la valeur de flux moyenne sur une période de temps. Ensuite, nous exploitons cette valeur moyenne pour définir le poids du lien. Le pseudo-code de l'algorithme ORRA est présenté dans l'algorithme 1.

L'ORRA avancé considère la contrainte de charge maximale de la matrice de trafic sur la base de l'ORRA simplifié, de manière à répondre aux exigences de QoS de l'utilisateur. La matrice de trafic représente la demande de trafic entre la source et

nœuds de destination, qui est exprimé comme  $M$ . Pour tous les nœuds, deux nœuds quelconques constitueront une paire de nœuds source et destination. Et avoir un chemin de routage robuste entre deux nœuds quelconques calculé par la stratégie de routage robuste adaptative ci-dessus. Voici un exemple de contrainte de capacité de liaison réseau avec  $m$  nœuds et  $j$  liens.

Un réseau avec  $m$  nœuds, alors il y a  $m^2$  des chemins adaptatifs robustes entre les nœuds source et destination. Chaque chemin est constitué par un lien différent, et pour chaque lien, la capacité du lien est la somme de toutes les requêtes des utilisateurs à travers les chemins adaptatifs robustes incluant ce lien. Et puis on peut facilement atteindre l'équation ci-dessous:

$$C_{je} = \sum_{(Dakota \text{ du Sud})} m_{Dakota \text{ du Sud}}, \forall s \in V, \forall d \in V, \forall je \in PWASPF(Dakota \text{ du Sud}), (dix)$$

où  $m_{Dakota \text{ du Sud}}$  est l'élément de la matrice de trafic  $M$ .

---

Algorithme 1 : ORRA pour un créneau horaire

---

Saisir:  $R$ (route i ngmat rix),  $T$ (matrice de trafic)  
 Sortir:  $P$ (robustrout i ngpath)  
 Ensemble:  $W$ (lienwhuit),  $UR$ (utilisateur s smat rix des besoins),  
 $C_{max}$ (capacité de liaison)  
 % appellent l'algorithme OLS aux liens dormants [ $new\_O, sl$ ] =  $OLS(R, W)$ ; %  $new\_O$   
 affiche les poids des nouveaux liens %  $sl$   
 affiche le nombre de liens dormants  
 $m = Constant$ ; % définir le nombre d'exigences d'utilisateurs  
 pour  $j = 1, 2, \dots, m$  faire  
 % appellent l'algorithme de Dijkstra  
 $Chemin = Dijkstra(W, Source, Dest i nat i on)$ ; % le plus court chemin adaptatif pondéré en premier si  $Chemin = 0$  faire  
 $k = longueur(Chemin)$ ;  
 % appellent l'algorithme de criticité  
**marque1** :  $Cr1 = Criticité(C_{max})$ ;  
 $temp = C_{max}$ ;  
 % modifier les poids des liens pour  $j = 1$  à  $k-1$  faire  
 si  $C_{max}(Chemin(j), Chemin(j+1)) - UR(bande) > low_{je}$  suis faire  
 $C_{max}(Chemin(j), Chemin(j+1)) = UR(bande)$   
 Autre  
 % processus de réacheminement  
 aller à **marque1**;  
**fin**  
**Fin**  
 % appellent l'algorithme de criticité  $Cr2 = Criticité(C_{max})$ ;  
 $= (Cr2 - Cr1)$ ; % de différence de criticité  
 pour  $p = 1$  à  $k-1$  faire  
 $W(Chemin(p), Chemin(p+1)) = W(Chemin(p), Chemin(p+1)) * 100$ ; % mettre à jour la fin des poids des liens

**fin**  
**Fin %ORRA**

---

Après avoir déplacé le lien en fonction de l'algorithme de sommeil de lien optimisé proposé ci-dessus, la capacité des liens du réseau changera. Pour le moment, nous sommes en mesure d'obtenir la nouvelle capacité de liaison en termes de nouvelle matrice de routage par Eq. (dix). Il est défini comme  $c_{je} \in E$ , où  $E$  représente l'ensemble des nouveaux liens réseau calculés par l'algorithme de sommeil de lien optimisé.

rythme. Le problème ORRA avancé consiste à faire  $c_{je}$  moins que la capacité de liaison maximale par défaut, à savoir

$$C_{je} \leq C_{maximum}(je), \forall je \in E \quad (11)$$

où  $C_{maximum}(je)$  est la capacité maximale du lien  $je$ . La condition de contrainte représentée par la formule (11) est ajoutée à l'équation. (9) pour résoudre ce problème. Par conséquent, nous pouvons réaliser l'ORRA avancé proposé. Autrement dit, en ajoutant la contrainte de capacité de liaison indiquée dans l'équation. (11) à l'algorithme de sommeil de liaison optimisé illustré dans l'algorithme 1, nous pouvons alors surmonter ce problème et obtenir les chemins de routage robustes qui peuvent répondre aux exigences de QoS des utilisateurs.

Jusqu'à présent, nous avons proposé notre algorithme de routage robuste ORRA pour atteindre la plus grande efficacité énergétique du réseau du réseau de cloud computing. En transformant les liens redondants du réseau en état de veille, en simplifiant la topologie de routage du réseau, en réduisant le nombre de liens actifs, ORRA peut réduire la consommation d'énergie de transmission de certaines informations et par conséquent augmenter l'efficacité énergétique du réseau. Dans la section suivante, nous allons utiliser une série de simulation pour valider ses performances.

#### 4 Résultat de la simulation et analyse

Afin de valider l'ORRA proposé, nous menons une série d'expériences de simulation. Nous utilisons la topologie du réseau et les données de trafic du réseau dorsal GEANT dans le processus de simulation, qui contient 23 nœuds de routeur et 74 liens. EAR-OSPF dans [9] est signalé comme un algorithme de routage économe en énergie. Nous comparerons ici les algorithmes EAR-OSPF et ORRA. Pour valider davantage notre algorithme ORRA, nous analysons également l'algorithme OLS basé sur OSPF (en abrégé OLS-OSPF). Dans nos expériences numériques, nous allons discuter des performances des trois algorithmes.

Pour vérifier la relation entre l'efficacité énergétique et le nombre de demandes d'utilisateurs, nous avons réalisé une expérience de simulation correspondante, obtenant le résultat de simulation illustré à la Fig. 4. L'axe horizontal de la Fig. 4 représente les 30 expérimentations, le nombre de requêtes utilisateur augmentant de 20 à chaque expérimentation. Les informations transmises par requête sont les mêmes et l'efficacité énergétique du réseau est égale au rapport entre la quantité d'informations et la consommation d'énergie du réseau. De la Fig. 4, on peut voir ça



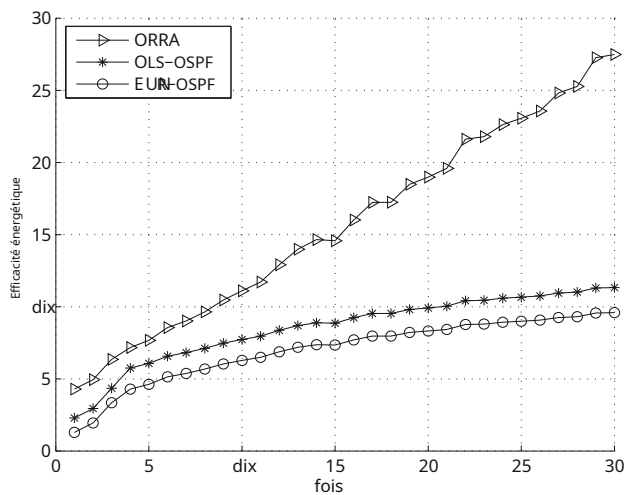


Figure 4 Efficacité énergétique de l'ORRA

avec l'augmentation des demandes des utilisateurs, l'efficacité énergétique augmente également progressivement. C'est nettement raisonnable. Selon l'ORRA, plus les demandes des utilisateurs augmentent, plus l'utilisation du lien est élevée. Par conséquent, le débit des réseaux est rapidement augmenté, mais en raison du schéma de sommeil d'ORRA, la consommation d'énergie du réseau s'ajoute lentement. Et ainsi l'efficacité énergétique du réseau devient importante avec l'augmentation des demandes des utilisateurs. Pour OLS-OSPF et EAR-OSPF, nous obtenons également des conclusions similaires. Cependant, contrairement à ORRA, OLS-OSPF et EAR-OSPF ont une efficacité énergétique beaucoup plus faible bien que celle-ci augmente avec la croissance du nombre de demandes des clients. Cela indique qu'ORRA peut effectuer la livraison de paquets à haute efficacité énergétique.

Nous analysons maintenant les performances de trois algorithmes. Figure 5 est l'histogramme d'utilisation des liens de trois algorithmes, l'axe horizontal représentant la distribution de l'utilisation des liens, à savoir  $(0, 0, 1]$  désignant la probabilité de liens dont l'utilisation est inférieure à 10 %,  $(0, 1, 0, 2]$  indiquant l'utilisation des liens entre 10 et 20 %, et d'autres exprimant une signification similaire. L'axe vertical de la Fig. 5 représente le nombre de liens utilisés. De la Fig. 5, ORRA détient les meilleures performances dans la plage de 0,1 à 0,4. Cela indique que les méthodes de routage ORRA peuvent maintenir la majeure partie de l'utilisation des liaisons entre 20 et 40 % de distribution. L'utilisation de la liaison n'est ni trop élevée pour entraîner une congestion du réseau et des pannes de liaison, ni trop faible pour entraîner un gaspillage. Cela montre qu'ORRA présente une bien meilleure utilisation des liens que OLS-OSPF et EAR-OSPF.

Chiffre 6 montre la distribution du nombre de demandes d'utilisateurs en échec en raison d'une capacité de bande passante insuffisante des liens. L'axe horizontal de la Fig. 6 représente les intervalles de temps. L'axe vertical de la Fig. 6 indique le nombre de bogues de demandes d'utilisateurs qui peuvent être acheminés avec succès mais la capacité de bande passante de liaison du chemin de routage ne répond pas aux exigences de l'utilisateur. Nous pouvons facilement trouver à partir de la Fig. 6 ce

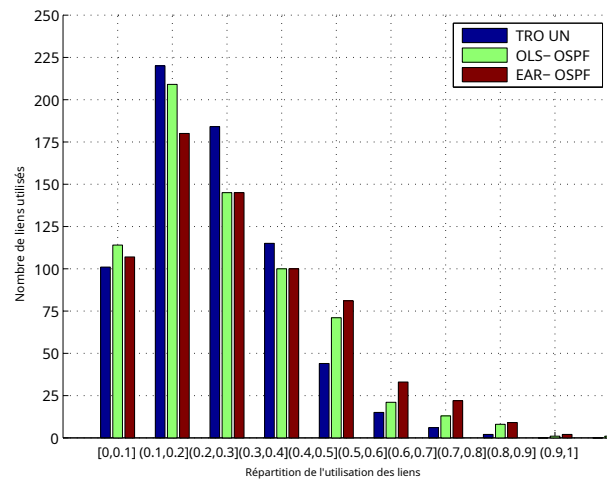


Figure 5 Utilisation de lien de trois algorithmes

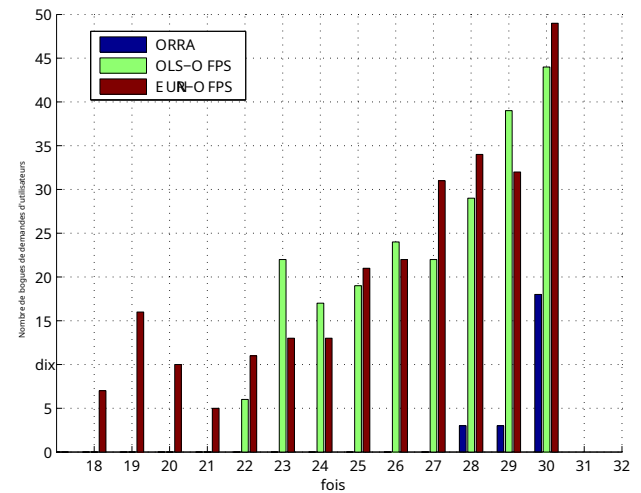


Figure 6 Répartition du nombre de bogues des requêtes des utilisateurs

il n'y a pas de numéro de bogue avant l'intervalle de temps 28 pour ORRA, tandis que OLS-OSPF et EAR-OSPF se produisent plus tôt. Les performances de l'algorithme EAR-OSPF sont les pires tandis que l'OLS-OSPF suit. Après l'intervalle de temps 28, le nombre de bogues d'ORRA est également bien inférieur à OLS-OSPF et EAR-OSPF. Cela suggère en outre que ORRA détient les meilleures performances de routage que les deux autres algorithmes.

Chiffre 7 trace la distribution de l'utilisation pour chaque liaison du réseau, où l'axe horizontal représente les intervalles de temps et l'axe vertical indique la taille numérique de l'utilisation de la liaison. La courbe décrit l'utilisation de chaque lien, et la ligne transversale est l'utilisation moyenne de trois algorithmes. Chiffre 7 nous indique que l'utilisation moyenne des liens dans l'ordre décroissant est ORRA, OLSA-OSPF et EAR-OSPF. L'utilisation moyenne d'ORRA est de 319,6486, ce qui est supérieur à ceux d'OLSA-OSPF et d'EAR-OSPF. De plus, nous calculons la variance de trois algorithmes

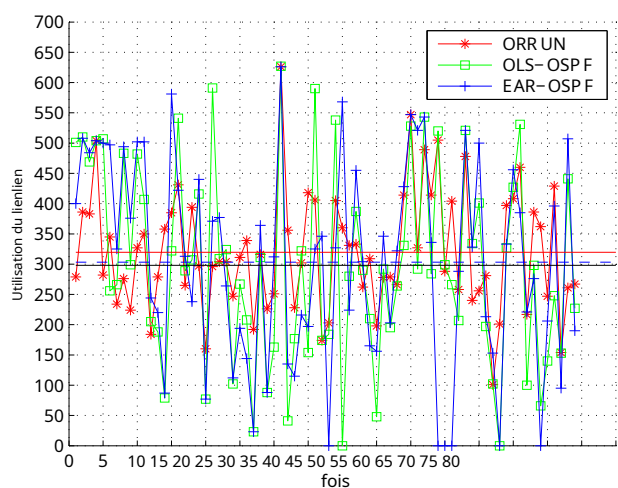


Figure 7 Répartition de l'utilisation des liens pour trois algorithmes

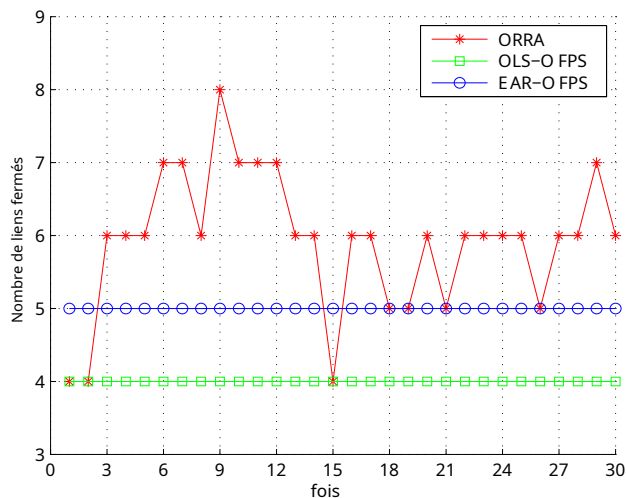


Figure 8 Répartition du nombre de liens fermés

pour présenter la fluctuation de l'utilisation des liens. Les résultats montrent que les variances de ORRA, OLSA-OSPF et EAR-OSPF sont de  $0.971 \times 10^4$ ,  $2.629 \times 10^4$  et  $2.884 \times 10^4$ , respectivement. Les variances d'OLSA-OSPF et d'EAR-OSPF sont presque deux fois supérieures à celles d'ORRA. Ceci explique que l'ORRA puisse effectivement utiliser les liens. Contrairement à OLSA-OSPF et EAR-OSPF, il n'y a pas d'utilisations de liaison trop faibles ou trop élevées pour ORRA. Ceci est très important pour fournir des communications robustes et économes en énergie.

Chiffre 8 indique la distribution du nombre de liens fermés pour cause de sommeil. ORRA peut définir dynamiquement le poids du lien dans chaque période de temps, ce qui affectera l'état de veille et actif du lien et aura donc un impact sur le chemin de routage. EAR-OSPF reste toujours inchangé après l'initialisation, ce qui a conduit le réseau à ne pas modifier dynamiquement la répartition du routage en fonction de l'état de

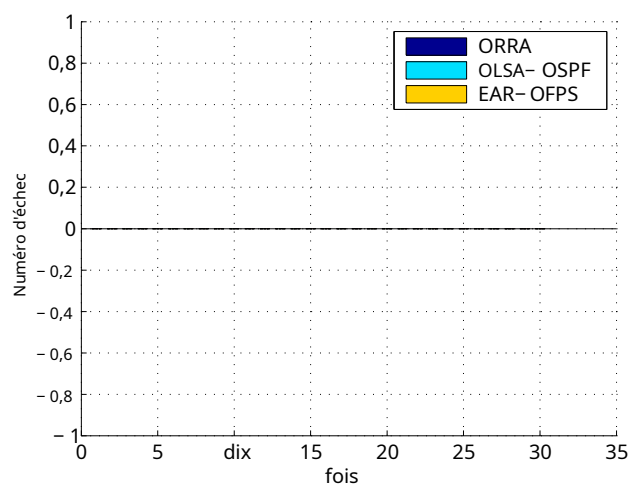


Figure 9 Répartition statistique du routage des pannes

demandes des utilisateurs et non pour réaliser l'équilibrage de charge. De ce fait, la robustesse du réseau d'EAR-OSPF est relativement faible. Comme on peut le voir sur la Fig. 8, ORRA peut modifier dynamiquement le nombre de liaisons dormantes afin de s'adapter aux variations du réseau. EAR-OSPF et OSL-OSPF n'ont pas une telle supériorité. Ceci explique qu'ORRA est efficace, non seulement capable d'améliorer l'efficacité énergétique du réseau, mais aussi capable de répondre à la demande de QoS des utilisateurs et d'améliorer la robustesse du réseau.

Chiffre 9 désigne la distribution du routage des pannes. De la Fig. 9, nous pouvons constater que la figure est vide. Cela montre que les performances de routage de trois algorithmes sont très bonnes. Cela indique en outre que ces algorithmes peuvent obtenir une robustesse relativement meilleure lorsqu'ils effectuent la mise en réseau économe en énergie. Plus important encore, à partir des simulations ci-dessus, nous pouvons voir qu'ORRA détient non seulement la robustesse pour le cloud computing, mais aussi l'efficacité énergétique des réseaux beaucoup plus élevée que les autres méthodes. Cela démontre qu'ORRA est faisable et prometteur pour mettre en œuvre le réseau économe en énergie pour le cloud computing.

## 5. Conclusion

Cet article propose un algorithme de routage robuste pour résoudre le problème de routage dans les réseaux économes en énergie pour le cloud computing. En décrivant le problème d'efficacité énergétique du réseau dans le processus optimal de maximisation du nombre de liens dormants dans le réseau, nous pouvons trouver l'ensemble maximal de liens dormants pour économiser l'énergie du réseau. Nous exploitons la technologie de réacheminement du trafic pour améliorer l'utilisation des liens, avec pour résultat que les liens autant que possible peuvent rester en état de veille. De plus, nous présentons une méthode de sommeil optimisée et une stratégie d'adaptation au poids pour améliorer



l'efficacité énergétique et la robustesse des réseaux de cloud computing. Par conséquent, notre algorithme peut éviter la congestion du trafic et augmenter l'efficacité énergétique du réseau. Les résultats de la simulation montrent que notre algorithme détient les meilleures performances que les méthodes précédentes.

**Remerciements** Ce travail a été soutenu en partie par la Fondation nationale des sciences naturelles de Chine (n° 61071124), le programme pour les excellents talents du nouveau siècle à l'université (n° NCET-11-0075), les fonds de recherche fondamentale pour les universités centrales (n° 61071124), N120804004, N130504003) et le Fonds de bourses d'État (201208210013). Les auteurs tiennent à remercier les relecteurs pour leurs précieux commentaires.

## Les références

- Pickavet, M., Vereecken, W. et Demeyer, S. et al. (2008). Besoins énergétiques mondiaux pour les TIC : la montée en puissance des réseaux. *Actes de l'ANTS'08*(p. 1–3).
- Jiang, D., Xu, Z., Chen, Z., et al. (2011). Estimation conjointe temps-fréquence parcimonieuse du trafic réseau à grande échelle. *Revue informatique*, 55(10), 3533–3547.
- Jiang, D., Xu, Z., Nie, L., et al. (2012). Une approche approximative du trafic de bout en bout dans les réseaux de communication. *Journal chinois d'électronique*, 21(4), 705–710.
- Bolla, R., Bruschi, R., & Davoli, F., et al. (2009). Optimisation des performances écoénergétique pour les équipements de réseau verts de nouvelle génération, In *Actes du PRESTO'09*(p. 49–54).
- Cianfrani, A., Eramo, V., & Listanti, M., et al. (2010). Un algorithme de routage économe en énergie pour un protocole OSPF vert, In *Actes de l'INFOCOM'10*(p. 1–5).
- Restrepo, J., Gruber, C., & Machoca, C. (2009). Routage sensible au profil énergétique, In *Actes de GreenComm'09*(p. 1–5).
- Chiaraviglio, L., Mellia, M. et Neri, F. (2009). Réduction de la consommation d'énergie dans les réseaux de base, en *Actes de la CPI'09*(p. 1–5).
- Chiaraviglio, L., Mellia, M. et Neri, F. (2009). Réseaux fédérateurs économes en énergie : une étude de cas, en *Actes de la CPI'09*(p. 1–5).
- Cianfrani, A., Eramo, V., Listanti, M., et al. (2009). Une stratégie de routage intégrée OSPF pour des économies d'énergie compatibles QoS dans les réseaux fédérateurs IP. *Transactions IEEE sur la gestion des réseaux et des services*, 9(3), 254–267.
- Lai, P., Yang, Q., & Wu, C., et al. (2011). Configuration de la topologie du réseau vers des réseaux IP économes en énergie, In *Actes de l'ICCSN'11*(p. 95–99).
- Zhang, D., Yang, Z., Raychoudhury, V., Chen, Z., & Lloret, J. (2013). Un protocole de routage économe en énergie utilisant les tendances de mouvement dans les réseaux ad hoc véhiculaires. *Le journal informatique*, 58(8), 938–946.
- Zhang, D., Zhang, D., Xiong, H., Hsu, C. et Vasilakos, AV (2014). BASA : Création de réseaux sociaux mobiles Ad-Hoc sur Android. *Revue IEEE*, 28(1), 4–9.
- Jiang, D., Zhao, Z., Xu, Z., Yao, C. et Xu, H. (2014). Comment reconstruire le trafic de bout en bout sur la base d'une analyse temps-fréquence et d'un réseau de neurones artificiels. *AEU-International Journal of Electronics and Communications*, 68(10), 915–925.
- Zhang, D., Chen, M., Guizani, M., Xiong, H. et Zhang, D. (2014). Prédiction de la mobilité dans le cloud des télécommunications à l'aide d'appels mobiles. *Communications sans fil IEEE*, 21(1), 26–32.
- Jiang, D., Xu, Z., Zhang, P. et Zhu, T. (2014). Une approche de détection d'anomalies basée sur le domaine pour transformer le trafic à l'échelle du réseau. *Journal des applications réseau et informatique*, 40(2), 292–306.
- Tizghadam, A., & Léon-Garcia, A. (2010). Ingénierie autonome du trafic pour la robustesse du réseau. *Journal IEEE sur des domaines sélectionnés des communications*, 28(1), 39–50.
- Bois, DR (1997). Un algorithme pour trouver une clique maximale dans un graphe. *Lettres de recherche opérationnelle*, 21(5), 211–217.
- Jaskiewicz, A. (2002). Sur les performances de la recherche locale génétique à objectifs multiples sur le problème du sac à dos 0/1 - Une expérience comparative. *Transactions IEEE sur le calcul évolutif*, 8(4), 402–412.



**Dingde Jiang** a reçu le Doctorat en communication et systèmes d'information de la School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, Chine, en 2009. Il est professeur associé au College of Information Science and Engineering, Northeastern University, Shenyang, Chine. Ses intérêts de recherche incluent les réseaux économes en énergie, les réseaux cognitifs, la mesure de réseau.

sécurité des réseaux, l'ingénierie du trafic Internet et l'analyse des performances. Le Dr Jiang est membre de l'IEEE et de l'IEICE.



**Zhengzheng Xua** obtenu un doctorat en sciences de gestion et en ingénierie au Collège des sciences et de l'ingénierie de l'information, Northeastern University, Shenyang, Chine. Elle travaille à l'École d'économie et de gestion, AnQing Normal University, AnQing 246011, Chine. Ses intérêts de recherche comprennent la gestion de la chaîne d'approvisionnement et de la logistique, l'analyse décisionnelle, la modélisation et l'optimisation.



**Jindi Liua** reçu MS et B.Sc. en génie des communications, Northeastern University. Elle est actuellement étudiante diplômée pour un doctorat en génie électrique et des communications à la Northeastern University. Ses intérêts de recherche incluent les réseaux d'efficacité énergétique et la détection d'anomalies.



**Wenhui Zhao** a reçu un B.Sc. en génie des communications, Université du Nord-Est à Qinhuangdao. Elle est actuellement étudiante diplômée d'une maîtrise en génie électrique et des communications à la Northeast University. Ses intérêts de recherche comprennent la mesure du réseau, l'ingénierie du trafic et la détection d'anomalies.