

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la
Propriété Intellectuelle
Bureau international



(43) Date de la publication internationale
12 décembre 2024 (12.12.2024)

WIPO | PCT

(10) Numéro de publication internationale
WO 2024/251669 A1

(51) Classification internationale des brevets :

H04L 69/04 (2022.01) H03M 7/30 (2006.01)
G06N 20/00 (2019.01) G06N 3/006 (2023.01)

(21) Numéro de la demande internationale :

PCT/EP2024/065212

(22) Date de dépôt international :

03 juin 2024 (03.06.2024)

(25) Langue de dépôt :

français

(26) Langue de publication :

français

(30) Données relatives à la priorité :

FR2305874 09 juin 2023 (09.06.2023) FR
FR2310288 27 septembre 2023 (27.09.2023) FR

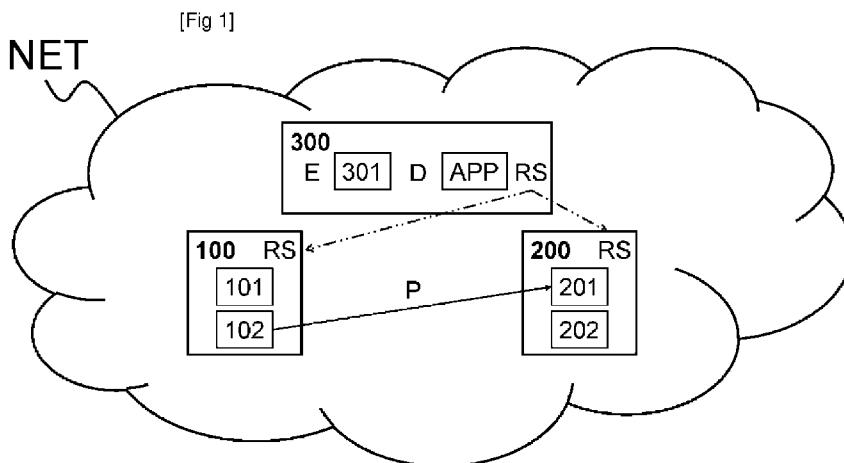
(71) Déposant : ORANGE [FR/FR] ; 111, quai du Président
Roosevelt, 92130 Issy-les-Moulineaux (FR).

(72) Inventeurs : LAMPIN, Quentin ; ORANGE - Intellectual
Property and Licensing, Orange Gardens - 44 Avenue de la
République - CS 50010, 92326 Chatillon (FR). DUMAY,
Marion ; ORANGE - Intellectual Property and Licensing,
Orange Gardens - 44 Avenue de la République - CS 50010,
92326 Chatillon (FR).

(81) États désignés (sauf indication contraire, pour tout titre de
protection nationale disponible) : AE, AG, AL, AM, AO,
AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA,
CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG,
KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY,
MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA,

(54) Title: TITLE: METHOD FOR OBTAINING A SET OF COMPRESSION RULES FOR A DATA PACKET TRANSMITTED
IN A NETWORK

(54) Titre : TITRE : PROCÉDÉ D'OBTENTION D'UN ENSEMBLE DE RÈGLES DE COMPRESSION D'UN PAQUET DE
DONNÉES TRANSMIS DANS UN RÉSEAU



(57) Abstract: The invention relates to a method for obtaining a set (RS) of compression rules able to compress a data packet (P) transmitted in a network (NET), the packet (P) comprising sections that take respective values expressed in the form of a bit string, the packet (P) being able to be compressed by applying compression rules that are able to delete bits of a value taken by a section of the packet (P), the method comprising, for multiple given compression rules, obtaining a count (D), in relation to a set (E) of data packets, referred to as the training set, of the number of bits deleted by the rule in question applied to packets of what is referred to as the training set, followed by obtaining the set (RS) of compression rules on the basis of the obtained counts (D).

(57) Abrégé : Procédé d'obtention d'un ensemble de règles de compression d'un paquet de données transmis dans un réseau L'invention se rapporte à un procédé d'obtention d'un ensemble (RS) de règles de compression aptes à compresser un paquet de données (P) transmis dans un réseau (NET), le paquet (P) comprenant des sections prenant des valeurs respectives exprimées sous la forme de suite de bits, le paquet (P) étant apte à être compressé par application de règles de compression aptes à supprimer des bits d'une valeur prise par

[Suite sur la page suivante]



NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,
RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH,
TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS,
ZA, ZM, ZW.

- (84) **États désignés** (*sauf indication contraire, pour tout titre de protection régionale disponible*) : ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), eurasien (AM, AZ, BY, KG, KZ, RU, TJ, TM), européen (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Publiée:

— avec rapport de recherche internationale (Art. 21(3))

une section du paquet (P), le procédé comprenant, pour plusieurs règles de compression données, une obtention d'un dénombrement (D), relativement à un ensemble (E) de paquets de données, ensemble dit d'apprentissage, du nombre de bits supprimés par la règle concernée appliquée à des paquets de l'ensemble dit d'apprentissage suivie par l'obtention de l'ensemble (RS) de règles de compression en fonction des dénombrements (D) obtenus.

Description

Titre : Procédé d'obtention d'un ensemble de règles de compression d'un paquet de données transmis dans un réseau

5 Domaine technique

Le domaine technique est celui de la compression et de la décompression de données.

Plus précisément, l'invention se rapporte à un procédé d'obtention d'un ensemble de règles de compression aptes à compresser un paquet de données transmis dans un réseau. Dans des modes de réalisation, l'invention se rapporte à un procédé de gestion de la compression et de la décompression d'un paquet de données transmis dans un réseau.

Les télécommunications numériques modernes fonctionnent sur le principe de l'échange de paquets de données entre émetteurs et récepteurs suivant des protocoles tels que le protocole IP (acronyme de l'anglais *Internet Protocol*). Un ensemble de paquets de données relatifs à une communication est envoyé par l'émetteur vers un récepteur. Les paquets contiennent l'adresse du récepteur pour pouvoir lui être adressé grâce à différents algorithmes de routage, et le contenu de la communication est reconstruit par le récepteur en regroupant les données contenues dans les différents paquets. Pour réaliser une communication complète, de nombreux protocoles doivent être utilisés qui remplissent chacun un rôle depuis l'envoi physique de bits, éléments atomiques constituant les données numériques, jusqu'à l'interprétation de données pour les applications informatiques. Les bits sont classiquement regroupés en octets, ensembles de huit bits. Les différents protocoles sont classiquement vus comme étant organisés en couches, un protocole d'une couche s'appuyant sur un protocole d'une couche inférieure. Le modèle OSI (de l'anglais *Open Systems Interconnection*) et le modèle Internet décrivent l'articulation des divers protocoles entre différentes couches.

Par exemple, dans une version simplifiée, le protocole HTTP (de l'anglais *Hypertext Transfer Protocol*) est un protocole de la couche Application, qui permet de transférer des données vers ou depuis un site web. Les données à transférer sont organisées en paquets de données. Le protocole HTTP utilise le protocole TCP (de l'anglais *Transmission Control Protocol*) qui appartient à la couche Transport, par exemple pour demander le transfert de paquets de données représentant le contenu d'un site web vers un utilisateur. Le protocole TCP, par exemple, fonctionne en trois phases : l'établissement d'une connexion, le transfert de données, et la fin de la connexion. Les protocoles de la couche Transport utiliseront un protocole de la couche Réseau, en général le protocole IP (*Internet Protocol*) dans une de ses versions IPv4 ou IPv6. Ce protocole ne s'intéresse qu'à la transmission de données organisées en paquets d'une adresse vers une autre, sans s'occuper de l'établissement de la connexion ou de l'ordre d'arrivée des paquets de données formant une transmission, ces considérations étant gérées par le protocole de la couche Transport supérieure. Le protocole IP utilisera un protocole de la couche Liaison, par exemple le protocole Ethernet qui va organiser l'envoi des données élémentaires constituant le paquet, ces données élémentaires étant des bits, transmis en utilisant la couche

Physique, que ce soit en utilisant une émission par fibre optique, ou par une communication sans fil suivant les cas.

D'autres protocoles de la couche Application vont utiliser d'autres protocoles des couches inférieures pour réaliser leurs fonctions. Par exemple, le protocole CoAP (de l'anglais *Constrained Application Protocol*) utilisé dans le domaine de l'Internet des objets s'appuie en général sur le protocole UDP (de l'anglais *User Datagram Protocol*) de la couche Transport.

Un point commun à l'ensemble des protocoles sur l'ensemble des couches est l'organisation des données échangées en paquets de données. Un paquet de données est organisé en deux parties : une première partie est l'en-tête (traduction de l'anglais *header*) du paquet qui regroupe les informations nécessaires au protocole qui a la charge de traiter le paquet ; une seconde partie est la charge (traduction de l'anglais *payload*) qui regroupe les données transportées dans le paquet lui-même. Au niveau de la couche Liaison, on parle de trame plutôt que de paquet, et, en plus de l'en-tête, il existe un pied de trame indiquant la fin de la trame.

Les en-têtes de paquets comprennent des champs. Un champ est un ensemble continu de bits d'une longueur fixe ou variable. Quand un champ est de longueur variable, des mécanismes permettent de savoir quelle est la longueur du champ effectivement transmis. Quand la valeur présente dans un champ de longueur fixe est de longueur variable, des bits de bourrage (traduction de l'anglais *padding*) vont compléter la valeur jusqu'à la longueur fixe du champ. Un champ dispose d'un identifiant qui indique la nature de la valeur qui est présente dans le champ. Les valeurs présentes dans le champ sont utilisées par le protocole concerné pour traiter le paquet de façon appropriée. Par exemple, dans l'en-tête d'un paquet IP, un champ dont l'identifiant est *Version* contiendra le numéro de version du protocole utilisé. La taille de ce champ est de quatre bits. D'autres champs concernent par exemple les adresses IP source et destination du paquet. Un en-tête est donc naturellement divisé en plusieurs sections, à savoir les champs de l'en-tête. La charge du paquet peut également être divisée en plusieurs sections, par exemple en définissant des sections d'une longueur constante exprimée en nombre de bits.

Un outil d'analyse de paquets tel que Wireshark ou tcpdump permet de capturer et analyser des paquets de données. L'analyse permet de séparer un paquet de données échangé sur un réseau entre son en-tête et la charge utile, et de lire les informations comprises dans l'en-tête et les données comprises dans la charge utile et d'identifier les diverses sections d'un paquet, à la fois dans son en-tête et dans sa charge. Nous rappelons que, à la suite de l'encapsulation des paquets de données, l'en-tête typique d'un paquet sera constitué de plusieurs en-têtes concaténés, un pour chaque protocole encapsulé. Par exemple, une trame Ethernet encapsulant un paquet IP commencera par l'en-tête IP, suivi de l'en-tête UDP. Les champs du paquet IP seront donc suivis dans la trame par les champs du paquet UDP.

L'ensemble de ces éléments sont bien connus dans l'état de la technique et ne seront pas décrits plus avant.

Etat de la technique

Il existe un besoin clair de compression des données échangées dans les télécommunications numériques pour économiser au mieux les ressources matérielles et en énergie consommées par les réseaux de télécommunications. Ce besoin explique la création de protocoles dédiés à la compression des paquets de données échangées.

Les paquets de données échangées sont constitués, comme on l'a vu, d'une part d'un en-tête contenant les informations nécessaires à la transmission du paquet et d'autre part d'une charge utile comprenant les données transmises par le paquet en tant que telles. La compression peut donc porter sur la charge utile, ou bien sur les en-têtes de paquets de données. Comme les valeurs de certains champs des en-têtes de paquets de données vont prendre des valeurs constantes ou variant peu, il est possible de procéder à des compressions très efficaces des paquets de données en cherchant uniquement à comprimer les valeurs des champs des en-têtes des paquets de données.

Un premier exemple de protocole comprimant les valeurs des champs d'en-têtes de paquets de données est le protocole ROHC (acronyme de l'anglais *Robust Header Compression*) qui est défini par l'organisme de standardisation IETF (pour *Internet Engineering Task Force*) dans le document RFC (*Request For Comments*, documents de spécification à la base d'Internet) numéro 3095. L'idée de base de ce protocole est que certaines valeurs des champs des en-têtes des paquets formant une même communication ne vont pas changer d'une communication sur l'autre. Par exemple, les valeurs des champs d'adresse source et destination vont rester les mêmes pendant une communication. Après un envoi d'un premier paquet contenant les valeurs de ces champs, ces valeurs seront effacées dans les en-têtes des paquets suivants afin de comprimer l'en-tête du paquet, et donc le paquet lui-même, et seront insérées à la réception puisqu'elles sont connues après l'envoi du premier paquet. L'effacement est effectué par un module de compression et l'insertion de la valeur par un module de décompression. D'autres types de compression peuvent être appliqués.

On obtient ainsi une compression très efficace des paquets de données même si la compression se limite aux en-têtes. Le protocole ROHC est utilisé par exemple dans les réseaux de télécommunications mobiles. Mais il nécessite que soit construit pour chaque communication un contexte indiquant quelles valeurs peuvent être comprimées, et de quelle manière. Les émetteurs et les récepteurs doivent donc mettre à jour régulièrement les contextes utilisés pour la compression des en-têtes. Cette opération de mise à jour est très coûteuse et d'autres mécanismes vont être préférés dans certains contextes d'utilisation.

Le contexte des réseaux de communication dédiés aux objets connectés impose des contraintes supplémentaires. On parle ici de réseaux étendus à basse consommation, également dénommés LPWANs, acronyme de l'anglais *Low-Power Wide Area Networks*. Dans un tel réseau, on retrouve des objets connectés qui doivent économiser au maximum leur consommation électrique, comme des compteurs déployés à des endroits n'ayant pas d'alimentation électrique et devant utiliser une batterie pendant plusieurs années. Ces objets communiquent sans fil avec une passerelle qui couvre une zone étendue. La mise à jour du contexte à

chaque communication impliquerait une consommation de bande passante élevée, si bien que le protocole ROHC de compression d'en-têtes n'est pas adapté dans ce cadre.

De plus, les données échangées par ces objets sont souvent de taille très réduite, et peuvent se limiter à signaler que l'objet est toujours actif, ou bien l'objet peut vouloir remonter une seule valeur, telle qu'une température pour un thermomètre, ou un niveau d'eau pour un compteur d'eau. La taille des en-têtes des paquets de données proportionnellement à la taille de la charge utile va donc être proportionnellement plus grande dans le contexte des LPWANs que dans d'autres contextes. Il existe donc un besoin de compression efficace des paquets de données qui, dans le contexte des LPWANs, va se porter sur la compression des valeurs des champs des en-têtes.

Le protocole SCHC (acronyme de l'anglais *Static Context Header Compression*), défini par l'IETF dans la RFC 8724, a été conçu pour être utilisé dans le contexte des LPWANs. Il se fonde sur un contexte statique. Pour un contexte donné, par exemple un type d'objet connecté déployé dans un réseau donné, des règles de compression vont être définies. Ces règles regroupent des opérations qui peuvent être appliquées aux valeurs des champs des en-têtes de paquets. Ces opérations peuvent être de trois types, toutes fondées sur l'idée que des valeurs de certains champs, ou une partie des valeurs, peuvent être effacées avant transmission puis reconstruites à la réception :

- Un premier type d'opération de compression est l'effacement : la valeur du champ est lue et sera effacée par le module de compression si la valeur est bien celle attendue. Le module de compression est par exemple situé dans l'objet connecté. La valeur effacée sera insérée à la réception par le module de décompression qui, lui, est situé dans la passerelle et qui dispose des mêmes règles que le module de compression. Ce type d'opération est adapté aux champs qui prennent une valeur constante, comme le champ *Version* des paquets IP. Si le champ prend une autre valeur que celle attendue, l'opération ne s'applique pas, la règle de compression est ignorée, et une autre règle sera éventuellement utilisée ou bien la valeur du champ pourra être transmise inchangée.
- Un deuxième type d'opération de compression utilise un dictionnaire : un dictionnaire est construit pour certaines valeurs du champ et si la valeur du champ se trouve dans le dictionnaire, le module de compression va remplacer la valeur du champ par son index dans le dictionnaire. Le module de décompression utilise le dictionnaire pour remplacer l'index par la valeur effacée. Là encore, une valeur de champ en-dehors du dictionnaire pourra être transmise inchangée si aucune règle ne s'applique.
- Un troisième type d'opération consiste en l'effacement du préfixe : la valeur du champ est une suite de bits, et un préfixe plus ou moins long peut être défini dans une opération de compression. Si la valeur a bien ce préfixe, le préfixe sera effacé à la compression et inséré à la décompression, ou bien la valeur sera transmise inchangée si le préfixe n'est pas reconnu par l'opération.

Pour des raisons de complétion, un quatrième type d'opération est défini qui consiste à ignorer la valeur du champ et à la transmettre telle quelle.

D'autres combinaisons sont possibles, ainsi que d'autres actions de compression/décompression, par exemple pour traiter les champs de longueur variable.

Un ensemble de règles de compression va alors être défini pour un contexte de déploiement donné. Par exemple, des règles de compression (et leurs règles symétriques de décompression) peuvent être définies pour une passerelle gérant un réseau LPWAN et pour l'ensemble des équipements connectés à ce réseau et à Internet grâce à la passerelle. L'ensemble de règles pourra alors traiter en compression et décompression l'ensemble des paquets de données qui s'échangent dans le réseau LPWAN, la compression se limitant aux valeurs des champs des en-têtes. Le protocole SCHC permet de déterminer quelle règle de compression s'applique lorsqu'un paquet de données circule dans le réseau.

Pour cela, chaque règle de compression présente dans l'ensemble est définie pour un format donné d'en-tête de paquet et va consister en un ensemble d'opérations à appliquer sur les valeurs des champs de l'en-tête d'un paquet qui respecte le format. Plusieurs règles peuvent couvrir un même format d'en-tête de paquet de données. Par exemple, des règles de compression vont être définies pour le format des paquets IP encapsulant un paquet UDP, et ces règles comprendront des opérations pour tous les champs d'un en-tête de paquet IP suivi par un en-tête de paquet UDP.

Une règle de compression fait partie d'un ensemble de règles et dispose d'un identifiant de règle. L'ensemble de règles va chercher à couvrir l'ensemble des formats possibles d'en-têtes de paquets de données qui peuvent être échangés dans un contexte donné.

Le protocole SCHC de compression/décompression est appliqué par un module de compression qui va traiter un paquet avant de l'adresser à un module de décompression. Les modules de compression et de décompression disposent du même ensemble de règles, défini pour un contexte donné. Le protocole va consister, lorsqu'un paquet de données doit être traité par le module de compression, à parcourir l'ensemble de règles dont dispose le module de compression afin de retenir une ou plusieurs règles. Si une règle ne correspond pas au format des champs de l'en-tête du paquet traité, la règle n'est pas retenue. Si le format des champs correspond, il faut examiner si les valeurs des champs correspondent à tous les opérateurs de correspondance prévus dans la règle. Si ce n'est pas le cas, la règle là encore n'est pas retenue. Si aucune règle de l'ensemble ne correspond, le paquet est expédié sans traitement et un indicateur de non-compression lui est préfixé. Dans tous les cas, la charge utile du paquet n'est pas traitée par le protocole SCHC. Dans le contexte des LPWANs, la compression du paquet limitée à la compression des valeurs des champs des en-têtes sera quand même efficace, car la taille des en-têtes est proportionnellement importante.

On voit ici l'intérêt de l'opération qui consiste à ignorer la valeur du champ et à la transmettre telle quelle. Cette opération permet de s'assurer qu'une règle pourra effectivement s'appliquer à tous les champs d'un en-tête donné : pour les champs faciles à comprimer, une opération de compression efficace sera définie, et pour les autres champs l'opération qui ne fait rien servira de complétion.

Finalement, plusieurs règles peuvent être retenues pour lesquelles le format des champs d'une part et les valeurs des champs d'autre part correspondent. Une règle est alors choisie, et une seule, par exemple la

première rencontre dans l'ensemble de règles, ou bien choisie aléatoirement, au choix de l'implémentation du protocole SCHC. Les opérations prévues dans la règle sont appliquées par le module de compression aux valeurs des champs de l'en-tête du paquet. Puis le paquet est transmis avec les valeurs de son en-tête comprimées, ainsi qu'avec l'identifiant de la règle qui a été appliquée. Le module de décompression reçoit le

5 paquet dont les valeurs de l'en-tête ont été comprimées ; utilise l'identifiant de la règle pour trouver les opérations qui ont été appliquées ; et applique les opérations symétriques de décompression. Par exemple, si, pour une valeur d'un champ, l'opération de compression a utilisé un dictionnaire, le module de décompression utilisera la valeur de l'index transmise pour le remplacer par la valeur prévue dans le dictionnaire.

- 10 Comme l'identifiant de la règle utilisée pour la compression doit être transmis, ainsi que certaines valeurs (index des dictionnaires), il va exister une limite au-delà de laquelle ajouter de nouvelles règles de compression dans un contexte donné ne sera plus intéressant car le gain éventuel qu'apporte la règle supplémentaire est dépassé par le coût de transmission induit par l'augmentation de la taille de l'identifiant de règle au fur et à mesure que de nouvelles règles sont ajoutées. Par exemple, au-delà de huit règles,
- 15 l'identifiant de règle transmis sera codé sur trois bits au lieu de deux. Si rajouter une neuvième règle ne fait gagner en moyenne qu'un seul bit, l'ajout de cette neuvième règle n'est pas intéressant. De même si une règle ajoutée implique l'utilisation de bits supplémentaires pour encoder les index de dictionnaires.

- Il est facile de voir que les opérations de compression et de décompression dans le protocole SCHC peuvent être sans perte, c'est-à-dire que la décompression de l'en-tête du paquet de données selon les opérations
- 20 prévues dans le protocole SCHC reconstruit exactement l'en-tête tel qu'il était avant compression. C'est le cas général même s'il peut exister des cadres d'utilisation où ce n'est pas le cas.

- Les ensembles de règles de compression sont donc définies, dans le protocole SCHC, pour un contexte donné, statique, et correspondent aux formats des paquets attendus dans les échanges entre émetteurs et
- 25 récepteurs. Ils sont définis par un expert qui sait que, dans un contexte de déploiement donné, tel ou tel protocole va être utilisé, selon une encapsulation donnée, et qui sait donc quels formats des en-têtes de paquets sont attendus. L'expert sait également que tel ou tel champ des paquets respectant tel ou tel format prendra une valeur constante, ou bien une valeur parmi un ensemble de taille limitée, ou bien une valeur qui aura souvent le même préfixe. Par exemple, l'expert sait que le champ *IP-Version* qui apparaîtra dans les en-têtes des paquets respectant plusieurs formats (IP seul, ou IP encapsulant des paquets UDP, ou autre...)
- 30 sera constante. Ou bien l'expert sait que les valeurs d'adresses IP présentes dans un champ donné appartiennent à une plage limitée et auront donc un préfixe commun. Cette connaissance permet à l'expert de choisir l'opération la plus appropriée pour un champ donné dans un format donné et de l'intégrer dans la règle. L'expert construit donc un ensemble de règles qui sera déployé, par exemple attaché à des modules de compression et de décompression embarqués dans un ensemble d'objets connectés et une passerelle
- 35 correspondante, pour mettre en œuvre le protocole SCHC.

Le protocole SCHC a été initialement conçu dans le contexte des réseaux LPWANs mais il pourrait être utilisé dans d'autres contextes, comme les réseaux domestiques, ou pour la compression d'en-têtes de trames

Ethernet. Les opérations de compression du protocole SCHC pourraient aussi être appliquées à des sections de la charge d'un paquet de données et pas seulement à l'en-tête.

Dans la situation actuelle, le déploiement du protocole SCHC est ralenti par le besoin de créer de nouvelles règles pour chaque contexte d'utilisation. Cette situation fait que des potentiels de compression facilement accessibles ne sont pas utilisés. Ceci a des impacts négatifs sur les consommations énergétiques et matérielles des réseaux LPWANs ou celles d'autres réseaux dans lesquels la compression des en-têtes selon le protocole SCHC pourrait être utilisée.

Une autre limite de la situation actuelle est que les règles de compression définies pour un contexte de déploiement donné ne vont plus aussi bien fonctionner quand ce contexte évolue. Par exemple, si un objet connecté redémarre, son adresse réseau peut changer ou bien les numéros de port qu'il utilise, et des règles de compression qui étaient efficaces peuvent se retrouver inefficaces. C'est un point qui introduit de l'inefficacité dans les compressions qui peuvent être obtenues.

De plus, il est souhaitable d'utiliser les opérations de compression définies par le protocole SCHC, qui apportent une compression très efficace, dans un maximum de contextes. Par exemple, ces opérations de compression devraient pouvoir s'appliquer à des sections de la charge du paquet, et pas seulement à des champs d'en-tête. Or, il n'existe pas à l'heure actuelle de solution technique permettant d'appliquer les principes de compression d'en-têtes définis dans le protocole SCHC à un contexte dynamique, même si celui-ci est faiblement dynamique. Par exemple, si, dans un contexte de déploiement, la compression SCHC s'applique à une valeur d'adresse IP, et que cette valeur vient à changer même si c'est à une fréquence très faible, la règle établie dans un contexte statique devient inopérante dès le premier changement d'adresse IP. Ou bien, si une section donnée d'une charge varie, même peu, les règles du protocole SCHC ne vont pas s'appliquer.

L'invention vient améliorer la situation.

Exposé de l'invention

Selon un premier aspect fonctionnel, l'invention a trait à un procédé d'obtention d'un ensemble de règles de compression d'un paquet de données transmis dans un réseau, le paquet comprenant des sections prenant des valeurs respectives exprimées sous la forme de suite de bits, le paquet étant apte à être compressé par application de règles de compression aptes à supprimer des bits d'une valeur prise par une section du paquet, le procédé comprenant, pour plusieurs règles de compression données, une obtention d'un dénombrement, relativement à un ensemble de paquets de données, ensemble dit d'apprentissage, du nombre de bits supprimés par la règle concernée appliquée à des paquets de l'ensemble dit d'apprentissage suivie par l'obtention de l'ensemble de règles de compression en fonction des dénombrements obtenus.

Grâce à l'invention, un ensemble de règles de compression pertinentes en terme de compression est créé automatiquement. Les règles de compression comprennent des opérations de compression qui s'appliquent à

des valeurs prises par des sections de paquets de données. Les règles de compression de l'ensemble obtenu peuvent être appliquées aux paquets de données transmis dans le réseau afin de réaliser une compression effective et efficace des paquets de données transmis. L'avantage de l'invention est que les règles de compression peuvent changer automatiquement quand le contexte de transmission change et rester efficaces. Cela permet de faciliter le déploiement de protocoles de compression et de les utiliser dans de nombreux contextes, y compris des contextes dynamiques.

Dans l'invention, des grandeurs dites dénombrements sont obtenues à partir d'un ensemble de paquets, ensemble dit d'apprentissage. Cet ensemble dit d'apprentissage est représentatif du trafic attendu, qui devra être comprimé par l'ensemble de règles de compression obtenu. Les dénombrements obtenus permettent de connaître précisément les quantités de bits qui peuvent être effacées par des règles de compression données dans cet ensemble représentatif et ainsi de guider la création de règles de compression. L'obtention de dénombrements exacts de bits pouvant être effacés par des règles de compression peut être long et complexe à réaliser pour des ensembles dits d'apprentissage de grande taille. Mais les ensembles dits d'apprentissage doivent être de grande taille pour pouvoir être vraiment représentatifs du trafic attendu. L'enjeu de l'invention est de trouver une représentation efficace d'un ensemble dit d'apprentissage de grande taille afin de pouvoir réaliser ces dénombrements exacts.

Selon un mode de réalisation, le procédé comprend de surcroît l'application d'au moins une opération de compression à une valeur de section dudit paquet de données, ladite opération étant comprise dans une règle de compression appartenant à l'ensemble de règles de compression obtenu, suivie par la transmission dudit paquet de données auquel ladite au moins une opération de compression a été appliquée et d'une donnée représentative de la règle de compression à laquelle appartient ladite opération.

Grâce à ce mode de réalisation, les règles de compression obtenues sont appliquées à des paquets de données et permettent de réaliser des compressions effectives. Les paquets de données sont comprimés par l'application d'opérations de compression qui effacent des portions de valeur, voire des valeurs complètes, prises par des sections du paquet de données. Pour reconstruire les portions de valeur effacées à la réception du paquet comprimé, celui-ci doit être transmis avec une donnée représentative de la règle de compression qui lui a été appliquée.

Selon un mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec le précédent, la section du paquet de données à la valeur de laquelle est appliquée une opération de compression est un champ d'un en-tête du paquet de données.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, la section du paquet de données à la valeur de laquelle est appliquée une opération de compression est une délimitation définie au préalable de la charge du paquet de données.

Ces deux modes de réalisation permettent d'appliquer l'invention à l'ensemble des valeurs prises par les sections du paquet de données, qu'il s'agisse de champs d'en-têtes ou bien de parties délimitées et définies au préalable de la charge du paquet. Dans ce dernier cas, le fait que la délimitation soit définie au préalable

permet de faciliter l'application du procédé. Une opération donnée, retrouvée grâce à la donnée représentative transmise, va par exemple s'appliquer aux 256 premiers bits de la charge, et le paquet pourra être rétabli à l'original en utilisant la connaissance de cette définition au préalable de la délimitation. Une autre délimitation définie au préalable peut par exemple consister à s'appliquer à la totalité de la charge du paquet, ou bien à la première ou seconde moitié, ou à toute autre proportion définie au préalable.

Les champs d'un en-tête sont eux-mêmes des sections définies au préalable puisque la structure de l'en-tête, et donc sa délimitation en plusieurs champs, est définie par le protocole, ou l'empilement de protocoles, auquel appartient le paquet de données transmis.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec le précédent, le paquet de données auquel ladite au moins une opération de compression a été appliquée est transmis conjointement avec la donnée représentative de la règle de compression à laquelle appartient ladite opération de compression.

Grâce à ce mode de réalisation, le paquet de données est transmis conjointement avec la donnée représentative de la règle de compression qui lui a été appliquée. Cette transmission conjointe permettra de faciliter l'opération de décompression ultérieure qui sera appliquée au paquet de données, en s'assurant que, lorsqu'un paquet de données est transmis, il est transmis avec l'information qui permet de connaître l'opération de compression qui lui a été appliquée.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le paquet de données transmis est formé en concaténant une donnée représentative d'une règle de compression, ladite donnée étant un identifiant de la règle de compression consistant en une suite de bits, avec le paquet de données auquel a été appliqué au moins une opération de compression appartenant à ladite règle de compression qui appartient à un ensemble de règles de compression obtenu par apprentissage.

Grâce à ce mode de réalisation, un format précis du paquet de données compressé est défini qui va permettre de faciliter la transmission de celui-ci. Le paquet de données transmis est un nouveau paquet de données formé en concaténant un identifiant de la règle de compression, à savoir une suite de bits, avec le paquet de données original auquel a été appliqué la règle de compression. Le paquet de données compressé est donc transmis conjointement avec un identifiant de la règle de compression qui permettra de retrouver celle-ci parmi l'ensemble de règles de compression obtenu par apprentissage.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'obtention de l'ensemble de règles de compression comprend la mise en œuvre d'un algorithme d'apprentissage par renforcement.

L'apprentissage par renforcement, au contraire de l'apprentissage supervisé, ne nécessite aucune intervention d'expert. L'effet technique obtenu est donc que les règles de compression peuvent être obtenues automatiquement afin de réaliser la compression des paquets de données transmis dans le réseau. Un

premier avantage est que les règles de compression peuvent être obtenues à bas coût et rapidement puisque cette obtention ne fait pas intervenir d'expert. Un autre avantage est que les règles de compression peuvent changer automatiquement quand le contexte de transmission change et rester efficaces.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'algorithme d'apprentissage par renforcement mis en œuvre dans l'obtention de l'ensemble de règles de compression utilise comme ensemble d'apprentissage l'ensemble dit d'apprentissage, formé de paquets de données collectés au préalable ; met en œuvre un environnement qui comprend l'ensemble dit d'apprentissage et l'ensemble de règles de compression ; et le changement d'état dans l'algorithme d'apprentissage par renforcement comprend l'ajout d'une nouvelle règle à l'ensemble de règles de compression.

Comme l'invention s'appuie dans ce mode de réalisation sur l'utilisation d'un algorithme d'apprentissage, elle nécessite d'utiliser un ensemble d'apprentissage qui doit être défini au préalable. Cet ensemble d'apprentissage est un ensemble de paquets de données pour lequel l'algorithme d'apprentissage va chercher à produire un ensemble de règles de compression qui le comprime au mieux. Cet ensemble d'apprentissage est l'ensemble dit d'apprentissage qui est utilisé pour obtenir les dénombrements du nombre de bits supprimés par une règle de compression qui serait appliquée à des paquets de l'ensemble dit d'apprentissage. Il est clair que l'apprentissage sera d'autant plus pertinent que l'ensemble d'apprentissage sera représentatif du trafic attendu dont on doit compresser les paquets de données. Si l'ensemble d'apprentissage est trop petit, en particulier, l'apprentissage ne pourra pas obtenir un ensemble de règles de compression efficaces dans tous les cas, car des exemples de trafic qui auraient pu être comprimés ne sont pas présents dans l'ensemble d'apprentissage.

L'apprentissage par renforcement consiste à faire évoluer l'ensemble de règles de compression qui est appris, dans un environnement comprenant l'ensemble d'apprentissage que l'ensemble de règles de compression doit compresser. Cette évolution se fait par l'ajout d'une nouvelle règle à l'ensemble de règles de compression. L'ensemble d'apprentissage, quant à lui, n'évolue pas durant la phase d'apprentissage.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'algorithme d'apprentissage par renforcement comprend une fonction de récompense qui est calculée selon la compression des paquets de l'ensemble d'apprentissage par les règles de compression appartenant à l'ensemble de règles de compression.

Il est avantageux que plusieurs paramètres de l'algorithme d'apprentissage par renforcement soient définis afin de pouvoir l'exécuter. Un apprentissage par renforcement consiste à faire évoluer un acteur, ici l'ensemble de règles de compression que l'on apprend, dans un environnement, ici la réunion de l'ensemble de règles et de l'ensemble d'apprentissage, et de guider les évolutions de l'acteur grâce à une fonction de récompense. La fonction de récompense est la compression réalisée par l'ensemble de règles de compression sur les paquets de l'ensemble d'apprentissage. Meilleure est la compression réalisée par l'ensemble de règles, meilleur est cet ensemble et donc l'apprentissage par renforcement aura tendance à produire un ensemble de règles qui produit une meilleure compression. Comme déjà vu, le changement d'état

est l'ajout d'une règle à l'ensemble de règles ; la fonction de récompense viendra sanctionner si cet ajout de règle est pertinent ou non et permet bien de faire évoluer l'ensemble de règles de compression vers une meilleure compression.

L'avantage de ce mode est donc de préciser comment exécuter un algorithme d'apprentissage par renforcement pour apprendre un ensemble de règles de compression et comment choisir des paramètres pertinents de l'apprentissage par renforcement pour obtenir une bonne compression des paquets de l'ensemble d'apprentissage par l'ensemble de règles résultat de l'apprentissage.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'apprentissage d'un ensemble de règles de compression est réalisé à partir d'un ensemble initial consistant en une règle unique ne réalisant aucune compression.

Dans ce mode de réalisation, le point de départ est un ensemble de règles formé d'une règle unique s'appliquant à tous les paquets de l'ensemble d'apprentissage collecté. Cet ensemble permet de s'assurer que dès le départ, l'ensemble de règles va bien s'appliquer à tout l'ensemble d'apprentissage même si la règle initiale ne réalise pas de compression sur les valeurs de sections des paquets de l'ensemble d'apprentissage. Dans la plupart des modes de réalisation de notre invention, l'ensemble de règles de compression évolue de manière à s'assurer que, pour tous les paquets de l'ensemble d'apprentissage, il existe dans l'ensemble de règles de compression une règle qui puisse s'appliquer, ce qui est bien le cas avec cette règle unique ne réalisant aucune compression.

L'avantage de ce mode de réalisation est de ne nécessiter l'intervention d'aucun expert pour définir l'ensemble initial de règles de compression qui est soumis à l'algorithme d'apprentissage.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'apprentissage d'un ensemble de règles de compression est réalisé à partir d'un ensemble initial comprenant une règle de compression fournie par un expert.

L'avantage de ce mode de réalisation est de permettre à un expert de s'assurer qu'une règle qui lui semble très utile est bien présente dans l'ensemble de règles de compression. En la plaçant dès le départ de l'apprentissage, on obtient ce résultat. Le reste de l'algorithme d'apprentissage pourra se faire par renforcement, donc sans intervention ultérieure de l'expert. On obtient donc ici un mode mixte où l'intervention de l'expert est limitée.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le procédé d'obtention comprend les étapes suivantes exécutées lors de l'obtention de l'ensemble de règles de compression :

- une étape de sélection d'une règle appartenant à l'ensemble de règles de compression en utilisant au moins un dénombrement parmi ceux obtenus ;
- une étape de création d'une nouvelle règle à partir de la règle sélectionnée en utilisant au moins un dénombrement parmi ceux obtenus ;

- l'ajout de la nouvelle règle à l'ensemble de règles de compression ;

et, de plus, lesdites étapes sont itérées, jusqu'à ce qu'un critère d'arrêt soit satisfait, en utilisant à l'itération suivante l'ensemble de règles modifié par l'ajout de la nouvelle règle.

Grâce à ce mode de réalisation, l'obtention d'un ensemble de règles de compression est guidé de façon efficace. En effet, l'obtention peut fonctionner en réalisant aléatoirement des changements d'état qui seront retenus ou non selon une fonction de récompense. Cette manière de procéder pourrait d'une part prendre beaucoup de temps et d'autre part conduire à des ensembles de règles de compression comprenant un nombre trop élevé de règles si les règles les plus efficaces ne sont pas créées en premier. Or, on a vu que si un trop grand nombre de règles de compression est créé, cela devient un obstacle à une compression efficace car l'index de la règle utilisée lors d'une compression doit être transmise avec le paquet comprimé. Un trop grand nombre de règles implique un index codé sur un grand nombre de bits qui rend inefficace la compression.

Il faut donc guider l'obtention des règles de compression. Pour cela, des dénombrements exacts des bits qui peuvent être effacés par des règles de compression sont obtenus qui vont d'abord indiquer quelle règle est sélectionnée pour créer une nouvelle règle puis comment créer une nouvelle règle à partir de la règle sélectionnée.

L'obtention fonctionne par itération jusqu'à ce qu'un critère d'arrêt soit satisfait, et l'itération consiste à appliquer le même algorithme à un environnement formé du nouvel ensemble de règles de compression (avec une nouvelle règle ajoutée) et de l'ensemble dit d'apprentissage. Pour s'assurer que l'obtention prend bien en compte l'ajout de la nouvelle règle, il peut être utile de calculer à l'itération suivante des dénombrements qui prennent bien en compte l'ajout de la nouvelle règle. Cela peut se faire en isolant dans l'ensemble dit d'apprentissage les sous-ensembles de paquets pour lesquels les différentes règles s'appliquent. Dans ce mode, les valeurs comprimées par la nouvelle règle ne seront pas prises en compte dans les calculs de dénombrements. Les dénombrements ne seront obtenus que pour les valeurs qui ne sont pas comprimées dans l'état courant de l'ensemble de règles de compression qui est en cours d'obtention.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, si la règle ajoutée s'applique à tous les paquets de données de l'ensemble dit d'apprentissage auxquels la règle sélectionnée s'applique, alors la règle sélectionnée est retirée de l'ensemble de règles de compression.

Il est avantageux que l'ajout de la nouvelle règle assure que l'ensemble de règles qui est en cours d'obtention va bien s'appliquer à l'ensemble des paquets de l'ensemble dit d'apprentissage. C'est pour cela que, dans ce mode de réalisation, si la règle ajoutée ne couvre pas l'ensemble des paquets que couvrirait la règle sélectionnée à partir de laquelle a été créée la règle ajoutée, alors la règle sélectionnée est conservée. Comme l'algorithme d'obtention peut partir, dans certains modes, d'un ensemble comprenant une règle initiale s'appliquant à tous les paquets, on est sûr alors de bien disposer à la fin d'un ensemble de règles s'appliquant à tous les paquets de l'ensemble dit d'apprentissage. Cependant, la suppression de la règle

sélectionnée si la règle ajoutée couvre les mêmes paquets permet de s'assurer que des règles inutiles ne sont pas conservées et donc de minimiser la taille de l'ensemble de règles de compression résultat de l'algorithme d'apprentissage.

5 Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le critère d'arrêt est satisfait quand la compression obtenue sur l'ensemble dit d'apprentissage par l'ensemble modifié de règles de compression est inférieure ou égale à la compression obtenue par l'ensemble de règles précédent.

10 Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le critère d'arrêt est satisfait quand l'ensemble de règles atteint une taille prédéterminée exprimée en nombre de règles ou en nombre d'opérations de compression incluses dans les règles.

Ces deux modes ont l'intérêt de proposer un critère d'arrêt facile à calculer. Dans le premier mode, la compression réalisée par l'ensemble de règles sur l'ensemble dit d'apprentissage est calculée et l'obtention s'arrête quand la compression ne progresse plus. Cela va arriver par exemple quand toutes les valeurs des sections des paquets de l'ensemble dit d'apprentissage sont comprimées. Dans ce cas, il n'est plus possible de rajouter une opération de compression. La compression réalisée par le nouvel ensemble peut être inférieure à la compression réalisée par l'ensemble précédent car il faut prendre en compte que les règles de compression doivent être identifiées, et que la taille de l'index de la règle ajoutée peut être plus grand que la compression réalisée par cette règle. Dans le deuxième mode, on utilise un critère de taille absolue de l'ensemble de règles de compression, soit en nombre de règles soit en nombre d'opérations. Ce mode permet de s'assurer que la taille de l'ensemble de règles obtenu reste toujours inférieure à une certaine taille, ce qui est avantageux dans un contexte où l'on veut économiser les ressources et où on veut s'assurer que les index des règles de compression pourront être encodés avec un nombre limité de bits (de deux à quatre par exemple). Ce mode de réalisation est particulièrement avantageux quand le procédé de compression est utilisé par des équipements aux capacités mémoire limitées, qui ne pourront mémoriser qu'un nombre limité de règles de compression de données émises ou reçues. Les deux modes peuvent être combinés, à savoir que l'apprentissage se poursuit tant que la compression réalisée par l'ensemble de règles progresse sauf si une taille maximale prédéfinie est atteinte.

30 Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'étape de sélection d'une règle appartenant à l'ensemble de règles de compression utilise une grandeur dite potentiel de compression pour une section donnée de paquets de l'ensemble dit d'apprentissage.

Dans certains modes de réalisation, il peut être utile, pour guider la sélection de règles de compression vers des sections de paquets non encore traitées par des règles de compression, d'utiliser des potentiels de compression plutôt que des dénombrements exacts de la quantité de bits qui peuvent être effacées par des règles de compression. Ces potentiels de compression peuvent être calculés plus efficacement que des dénombrements exacts et présentent une alternative pour guider l'algorithme d'obtention.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'un des potentiels de compression obtenus est dit potentiel de compression de champ et est calculé pour une section donnée de longueur fixe suivant la formule $N(L - H)$ dans laquelle N est le nombre de fois que la section apparaît dans l'ensemble dit d'apprentissage ; L est la longueur de la section exprimée en bits et H est une grandeur dite estimateur d'entropie de champ qui est calculée suivant la formule $H = -\sum_{i \in [1, K]} f_i \log_2(f_i)$ dans laquelle K est le nombre de valeurs différentes $V_{i \in [1, K]}$ que prend la section dans l'ensemble d'apprentissage et f_i est la fréquence d'apparition de la valeur V_i .

Dans de nombreux modes de réalisations, les sections pour les valeurs desquelles l'algorithme d'apprentissage va chercher à produire des opérations de compression appartenant à des règles sont des champs d'en-têtes. Ceci explique le choix des termes « potentiel de compression de champ » et « estimateur d'entropie de champ ». Cependant, le mode de réalisation peut s'appliquer aussi bien à des champs d'en-tête qu'à des sections délimitées dans la charge d'un paquet de données.

La formule de calcul dans ce mode de réalisation s'applique à une section dont la longueur L, exprimée en bits, est constante pour toutes les valeurs prises par la section dans l'ensemble d'apprentissage. On verra plus loin qu'un autre mode de réalisation peut s'appliquer à des sections de taille variable.

Grâce à ce mode de réalisation, le potentiel de compression de champ est calculé pour une section donnée, afin de détecter quelle est la section la plus intéressante à comprimer et guider ainsi l'algorithme d'obtention.

Pour cela, nous utilisons une grandeur qui estime l'entropie présente dans une section délimitée des charges des paquets de l'ensemble dit d'apprentissage ou dans un champ donné des en-têtes des paquets de l'ensemble dit d'apprentissage, que nous appelons section en général. Nous rappelons qu'une entropie (au sens de la théorie de l'information) se calcule à partir de probabilités. Ici, nous disposons d'un ensemble d'apprentissage, donc de valeurs observées. C'est pour cela que nous calculons un estimateur d'entropie, et non une entropie proprement dite, à partir des fréquences d'apparition f_i des valeurs V_i que prend la section considérée. Ces fréquences sont strictement positives et inférieures ou égales à 1 par définition. Les logarithmes des fréquences sont pris en base 2 pour s'appliquer à des valeurs exprimées en bits. Les logarithmes sont forcément négatifs ou nuls puisque les fréquences valent 1 au maximum. Les négations rendent l'estimateur d'entropie forcément positif.

L'estimateur d'entropie peut donc valoir au minimum zéro. C'est le cas quand la section donnée présente une seule valeur dans l'ensemble dit d'apprentissage. La fréquence d'apparition de cette valeur unique vaut donc 1 ; et le logarithme vaut 0.

Par ailleurs, l'estimateur d'entropie sera toujours inférieur à la longueur de la section exprimée en bits. Si une section délimitée, ou un champ, prend un grand nombre de valeurs différentes, et donc si son estimateur d'entropie augmente, alors la longueur de la section exprimée en bits qui est nécessaire pour encoder ce grand nombre de valeurs différentes va également croître et restera toujours supérieure à l'estimateur d'entropie.

La grandeur dite potentiel de compression de champ est donc toujours positive et sera maximale pour une section donnée quand son estimateur d'entropie est nul. Cela arrive quand la section prend une valeur unique dans l'ensemble d'apprentissage et on comprend facilement qu'une section qui prend une valeur unique va être facile à comprimer. Il suffit d'effacer cette valeur ! Le potentiel de compression de champ diminue au fur et à mesure que l'estimateur d'entropie augmente.

Si une section dont l'estimateur d'entropie est nul n'apparaît que peu de fois dans l'ensemble d'apprentissage, ou bien si la longueur de la section est faible, une autre section dont l'estimateur d'entropie est plus élevé mais qui apparaît plus souvent, ou dont la longueur est plus grande, pourra être plus intéressant à comprimer que la section à l'estimateur d'entropie nulle. C'est pour cela que le potentiel de compression du champ prend en compte le nombre de fois que la section apparaît ainsi que sa longueur.

Le potentiel de compression de champ permet donc de guider de façon efficace l'algorithme d'obtention en indiquant les sections de l'ensemble d'apprentissage dont la compression rapportera le plus selon la fonction de récompense qui est donnée par la compression réalisée, lesdites sections pouvant être des champs d'entêtes des paquets ou des sections délimitées de la charge des paquets.

Cependant, ce guidage est moins précis que celui apporté par des dénombrements exacts du nombre de bits qui peuvent être effacés par des opérations de compression spécifiques. Ces dénombrements exacts sont obtenus par un calcul différent de celui utilisé, dans certains modes de réalisation, pour obtenir les potentiels de compression.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le potentiel de compression de champ est calculé pour une section donnée de longueur variable suivant la formule $\sum_{i=1}^K N_i \cdot (L_i - H_i)$ dans laquelle K est le nombre de valeurs différentes $V_{i \in [1, K]}$ que prend la section dans l'ensemble dit d'apprentissage ; N_i est le nombre de fois que la section prend la valeur V_i dans l'ensemble dit d'apprentissage ; L_i est la longueur de la valeur V_i exprimée en bits et H_i est une grandeur dite estimateur d'entropie partielle d'une valeur, qui est calculée pour la valeur V_i suivant la formule $H_i = -f_i \log_2(f_i)$ dans laquelle f_i est la fréquence d'apparition de la valeur V_i dans l'ensemble dit d'apprentissage.

Grâce à ce mode de réalisation, le potentiel de compression de champ peut être calculé pour une section de longueur variable. Plusieurs éléments vont donc être pris en compte dans ce calcul. Une valeur qui apparaît fréquemment, mais qui est de faible longueur, pourra moins contribuer au potentiel de compression de champ qu'une valeur qui apparaît moins fréquemment mais dont la longueur est plus grande. Ce mode de réalisation permet donc de réaliser un calcul précis du potentiel de compression de champ même pour une section de longueur variable. On peut remarquer que la formule utilisée dans ce mode donne les mêmes résultats que la formule donnée précédemment dans le cas où la section a une longueur L constante pour toutes les valeurs prises par la section.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le potentiel de compression de champ est calculé suivant la formule $N \frac{L}{H+a}$ dans laquelle a est une constante strictement positive.

5 Ce mode de réalisation donne une variante pour obtenir un potentiel de compression de champ. Dans cette formule, le potentiel est bien maximal quand l'estimateur d'entropie est nul. La constante positive permet d'assurer la définition du potentiel de compression de champ dans ce cas. Cette variante est adaptée au cas où la section est de longueur fixe.

10 Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le potentiel de compression de champ est calculé suivant la formule $\sum_{i=1}^K N_i \frac{L_i}{H_i+a}$ dans laquelle a est une constante strictement positive.

Ce mode de réalisation reprend la variante précédente mais en l'adaptant au cas où la section est de longueur variable.

15 Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le potentiel de compression de champ est dit potentiel de compression corrigé et est calculé pour une section donnée en multipliant le potentiel de compression de champ par un facteur de correction $\alpha(1 - \frac{K-\beta}{N+\gamma})$ dans lequel N est le nombre de fois que la section apparaît dans l'ensemble dit d'apprentissage ; K est le nombre de valeurs différentes que prend la section dans l'ensemble dit d'apprentissage ; et α , β et γ sont des coefficients positifs choisis.

20 Dans notre expérience, le potentiel de compression de champ permet de guider l'algorithme d'obtention dans certains modes de réalisation, comme alternative à l'obtention de dénombrements exacts mais présente le défaut suivant : quand une section présente des valeurs différentes réparties uniformément, l'estimateur d'entropie augmente trop lentement au fur et à mesure que de nouvelles valeurs sont observées dans l'ensemble dit d'apprentissage. Pour peu qu'elle ait une grande longueur, une section qui présente des valeurs à chaque fois différentes dans l'ensemble dit d'apprentissage pourra avoir un meilleur potentiel de compression de champ qu'une section avec peu de valeurs mais une taille plus petite. Or, à l'usage, c'est bien cette section avec peu de valeurs pour laquelle on souhaite créer une ou plusieurs règles de compression qui s'appliqueront au peu de valeurs observées. Alors que la section très longue avec des valeurs observées toutes différentes sera peu intéressante car il est probable qu'à l'usage, les valeurs qui apparaîtront dans cette section seront encore de nouvelles valeurs, et donc des valeurs qui ne seront pas comprimées par les règles de compression apprises sur un ensemble d'apprentissage où elles n'apparaissent pas.

C'est pour corriger ce défaut que nous introduisons un facteur de correction qui permet d'obtenir un potentiel de compression corrigé. Les coefficients α , β et γ sont de préférence positifs pour assurer que le facteur de correction fasse évoluer le potentiel de compression dans le sens voulu. L'idée est qu'une section qui

présente des valeurs toutes différentes doit être défavorisée par rapport à une section présentant une observation semblant moins aléatoire.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les coefficients positifs α , β et γ sont choisis pour prendre des valeurs constantes. Selon un mode de réalisation particulier de ce dernier mode, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les coefficients positifs α , β et γ prennent les valeurs $\alpha = 1$, $\beta = 0$ et $\gamma = 0$.

Grâce à ce dernier mode, le facteur de correction prend la valeur $1 - \frac{K}{N}$ et vaut donc zéro quand toutes les valeurs de la section sont différentes et que l'on a $K = N$. Une telle observation dans l'ensemble dit d'apprentissage d'une section dont toutes les valeurs sont différentes indique fortement que les valeurs en dehors de l'ensemble dit d'apprentissage seront également différentes, et qu'une telle section ne pourra pas être comprimée efficacement. Le facteur de correction va alors annuler le potentiel de compression de champ pour cette section. Quand une section prend une seule valeur, le facteur de correction vaut $1 - \frac{1}{N}$ et le potentiel de correction corrigé est donc diminué par rapport à la valeur initiale du potentiel de correction de champ, mais est finalement très peu diminué.

D'autres valeurs constantes peuvent bien sûr être prises pour affiner la correction apportée par le facteur de correction.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les coefficients positifs α , β et γ prennent les valeurs $\alpha = 1$, $\beta = 1$ et $\gamma = 0$.

Grâce à ce dernier mode, le facteur de correction prend la valeur $1 - \frac{K-1}{N}$ et vaut donc exactement 1 quand la section prend une seule valeur, c'est-à-dire pour $K = 1$. A l'opposé, le facteur de correction ne s'annule pas quand $K = N$ ce qui peut être intéressant pour conserver un potentiel de compression de champ non nul même sur une section qui semble quand même être très aléatoire puisque toutes les valeurs prises par la section dans l'ensemble d'apprentissage sont différentes. Ainsi, le potentiel de compression de champ étant non nul, l'algorithme d'obtention pourra dans certains contextes ajouter une ou des règles de compression relatives à la section.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les coefficients positifs α , β et γ sont choisis en fonction de K et de N .

Ce mode de réalisation permet d'obtenir d'autres valeurs du facteur de correction pour affiner son choix en fonction des tailles des échantillons observés dans l'ensemble d'apprentissage. Par exemple, en prenant $\alpha = 1$, $\beta = 0,5K$ et $\gamma = N$, on obtient un facteur de correction qui vaut $1 - \frac{0,5K}{2N}$ et qui présente une influence différenciée selon la taille de l'échantillon des valeurs observées pour la section dont on calcule le potentiel de compression corrigé.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'étape d'obtention d'un dénombrement comprend, pour une section du paquet prenant des valeurs respectives exprimées sous la forme de suite de bits, le dénombrement du nombre de bits qui seraient effacés à la suite de l'application d'une règle d'effacement de préfixe sur les valeurs prises par la section dans l'ensemble dit d'apprentissage.

Grâce à ce mode de réalisation, on va réaliser un dénombrement des bits qui seraient effacés dans l'ensemble d'apprentissage si on lui appliquait une règle d'effacement de préfixe donnée. De cette manière, l'algorithme d'obtention d'un ensemble de règles de compression est guidé par des dénombrements exacts des suites de bits pouvant être effacés dans l'ensemble dit d'apprentissage par des règles de compression données.

Selon un autre mode de réalisation, qui pourra être mis en œuvre cumulativement avec le mode précédent, l'étape d'obtention d'un dénombrement comprend, pour une section du paquet prenant des valeurs respectives exprimées sous la forme de suite de bits, les étapes suivantes :

- une création d'un arbre binaire représentant l'ensemble des valeurs pouvant être prises par une suite de bits, la racine de l'arbre représentant une suite vide de bits, les nœuds de l'arbre valant 0 ou 1, et un nœud de l'arbre de profondeur n représentant une suite de bits de longueur n dont la valeur est celle de la suite de 0 et de 1 rencontrés depuis la racine de l'arbre pour rejoindre le dit nœud de profondeur n , une arête dudit arbre étant étiquetée par le nombre de valeurs prises par la section ayant comme préfixe la suite de bits représentée par le nœud auquel mène l'arête ;
- une multiplication des étiquettes des arêtes de l'arbre binaire créé par la profondeur du nœud auquel mène l'arête, ce qui réalise un dénombrement des bits qui seraient effacés dans l'ensemble dit d'apprentissage par l'application de règles d'effacement de préfixes des valeurs prises par la section exprimées sous forme de suite de bits ;

et l'étape de création d'une nouvelle règle comprend la création d'une règle d'effacement de préfixe pour les valeurs prises par une section exprimées sous forme de suite de bits, le préfixe effacé étant celui pour lequel le dénombrement des bits effacés est le plus grand.

Grâce à ce mode de réalisation, le dénombrement des bits pouvant être effacés par une règle d'effacement de préfixe si on l'applique à l'ensemble d'apprentissage est réalisé de façon efficace par l'utilisation d'une structure de données astucieuse. Ce mode de réalisation est directement attaché à la définition d'une règle de compression comprenant l'effacement d'un préfixe donné de section quand il prend une valeur donnée. Cet exemple peut être utilisé en combinaison ou de manière alternative aux autres exemples présentés jusqu'à présent. Dans ce mode, pour une section donnée, toutes les valeurs prises par la section dans l'ensemble d'apprentissage sont présentées sous forme de suite de bits puis arrangées dans un arbre binaire. La racine de l'arbre correspond au début des valeurs prises par la section. Selon que le premier bit des valeurs vaut 0 ou 1, l'arête correspondante va être étiquetée par le nombre de valeurs prises par la section dont le premier bit vaut 0 ou 1. Cet arrangement est réalisé pour l'ensemble des valeurs, puis les étiquettes des arêtes sont multipliées par leur profondeur. Ces chiffres correspondent à la quantité de bits qui peuvent

être effacés par une règle de compression qui efface le préfixe correspondant au chemin dans l'arbre. Cette multiplication correspond donc à un dénombrement exact. Le choix de la quantité la plus grande de bits potentiellement effacés dans l'ensemble d'apprentissage permet alors de construire une règle de compression d'effacement de préfixe. Cette règle est celle qui efface le plus grand nombre de bits parmi les règles d'effacement de préfixe pour cette section si elle est appliquée à l'ensemble d'apprentissage, et elle sera donc potentiellement efficace quand elle sera appliquée sur du trafic réel.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les règles de compression de l'ensemble de règles de compression obtenu et les opérations de compression comprises dans lesdites règles de compression sont des règles et des opérations de compression telles que définies dans le protocole SCHC.

Grâce à ce mode de réalisation, l'objectif de proposer une technique d'apprentissage automatique de règles de compression pour le protocole SCHC est bien tenu. Le problème technique de disposer d'un moyen de déploiement du protocole SCHC sans solliciter d'expert pour définir les règles de compression est bien résolu.

Selon un autre aspect fonctionnel, l'invention a trait à un procédé de gestion de la transmission d'un paquet de données dans un réseau, le paquet comprenant des sections prenant des valeurs respectives, caractérisé en ce qu'il comprend les étapes suivantes :

- l'obtention préalable d'un ensemble de règles de compression comprenant des opérations de compression qui peuvent s'appliquer aux valeurs prises par les sections dudit paquet de données ;
- l'application d'au moins une opération de compression à une valeur de section dudit paquet de données, ladite opération étant comprise dans une règle de compression appartenant audit ensemble de règles de compression obtenu ;
- la transmission dudit paquet de données et d'une donnée représentative de la règle de compression à laquelle appartient ladite opération ;
- la réception dudit paquet de données et de ladite donnée représentative ;
- l'application audit paquet de données d'au moins une opération de décompression symétrique de l'au moins une opération de compression appliquée à une valeur de section dudit paquet de données.

Grâce à cet aspect de l'invention, une transmission complète de paquet de données est réalisée, en transmettant le paquet de données compressé. L'obtention des règles de compression permet de rendre le procédé automatique, sans intervention d'expert. La transmission de la donnée représentative de la règle de compression qui a été appliquée au paquet transmis permet de retrouver cette règle et donc les opérations de compression comprises dans la règle, et de rétablir le paquet original en appliquant les opérations de décompression symétriques.

On a donc transmis un paquet compressé au lieu du paquet original, ce qui permet de très fortes économies en termes de ressources matérielles et énergétiques. Comme les règles de compression sont obtenues automatiquement, il n'y a pas besoin de faire intervenir un expert, ce qui permet de déployer très facilement la transmission efficace selon l'invention.

Selon un mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec le mode précédent, le procédé comprend de surcroît une étape préalable de collecte dans le réseau de paquets transmis pour former un ensemble dit d'apprentissage utilisé lors de l'étape d'obtention d'un ensemble de règles de compression et une étape de déploiement synchronisé dans les nœuds du réseau de l'ensemble de règles de compression obtenu.

Grâce à ce mode de réalisation, la transmission de paquets dans le réseau se fait en prenant en compte les paquets effectivement transmis.

L'obtention préalable de l'ensemble de règles de compression utilise un général un ensemble dit d'apprentissage. Celui-ci peut être fourni par un expert, ou bien généré automatiquement selon des contraintes prédéfinies. Mais le mode de réalisation préféré est celui où l'ensemble d'apprentissage est collecté parmi les paquets transmis dans le réseau. L'obtention se fait alors avec un ensemble dit d'apprentissage directement représentatif du trafic de paquets dans le réseau et les règles de compression obtenues seront donc bien plus adaptées. Une fois l'ensemble de règles de compression obtenu, il doit être déployé dans les nœuds du réseau de manière synchronisée afin qu'un nœud qui reçoit un paquet compressé, et la donnée représentative de la règle de compression utilisée, puisse retrouver cette règle et appliquer l'opération de décompression symétrique.

De plus, grâce à ce mode de réalisation, le procédé peut être mis en œuvre de façon permanente. La collecte de paquets de données servant à la formation de l'ensemble dit d'apprentissage est permanente et permet de capturer des changements de contexte dans la transmission en cours de paquets. Un premier ensemble de règles de compression peut être déjà déployé et, si le contexte change, par exemple si les adresses IP des nœuds du réseau changent, les règles de compression deviennent inefficaces. Mais les paquets collectés permettent de relancer l'obtention des règles de compression avec un nouvel ensemble d'apprentissage, et le déploiement d'un nouvel ensemble de règles de compression permet alors de s'adapter au nouveau contexte, et de réaliser de nouveau des compressions efficaces. Le déploiement doit être synchronisé afin de s'assurer que le nœud qui reçoit un paquet compressé connaît bien la règle de compression qui a été appliquée afin de pouvoir effectuer l'opération de décompression symétrique pour rétablir le paquet de données.

Selon un autre mode de réalisation, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les règles de compression de l'ensemble de règles de compression obtenu, les opérations de compression comprises dans lesdites règles de compression, et les opérations symétriques de décompression sont des règles et des opérations de compression et de décompression telles que définies dans le protocole SCHC.

Grâce à ce mode de réalisation, le protocole SCHC peut être déployé dans des contextes dynamiques. Les ensembles de règles de compression seront mis à jour au fur et à mesure de l'obtention utilisant des nouveaux ensembles dit d'apprentissage représentatifs du trafic entre les différents points d'un réseau, et des changements dans les en-têtes de paquets circulant effectivement. Le protocole SCHC est donc rendu apte à s'appliquer à des contextes dynamiques grâce à l'utilisation de l'apprentissage par renforcement pour créer les ensembles de règles de compression.

Selon un premier aspect matériel, l'invention a trait à une entité de gestion de l'obtention d'un ensemble de règles de compression aptes à compresser un paquet de données transmis dans un réseau, le paquet comprenant des sections prenant des valeurs respectives exprimées sous la forme de suite de bits, le paquet étant apte à être compressé par application de règles de compression aptes à supprimer des bits d'une valeur prise par une section du paquet, caractérisé en ce que l'entité de gestion comprend les modules suivants :

- un module d'obtention, pour plusieurs règles de compression données, d'un dénombrement, relativement à un ensemble de paquets de données, ensemble dit d'apprentissage, du nombre de bits supprimés par la règle concernée appliquée à des paquets de l'ensemble dit d'apprentissage ;
- un module d'obtention de l'ensemble de règles de compression en fonction des dénombrements obtenus.

Selon un autre aspect matériel, l'invention a trait à un paquet de données formé en concaténant une donnée représentative d'une règle de compression, ladite donnée étant un identifiant de la règle de compression consistant en une suite de bits, avec un paquet de données auquel a été appliqué au moins une opération de compression appartenant à ladite règle de compression qui appartient à un ensemble de règles de compression obtenu par le procédé.

Selon un autre aspect matériel, l'invention a trait à un équipement réseau comprenant une entité de gestion de l'obtention d'un ensemble de règles de compression selon l'invention.

Selon un autre aspect matériel, l'invention a trait à un système de gestion de la transmission d'un paquet de données dans un réseau, le paquet comprenant des sections prenant des valeurs respectives, caractérisé en ce que le système comprend les modules suivants :

- un module d'obtention préalable d'un ensemble de règles de compression comprenant des opérations de compression qui peuvent s'appliquer aux valeurs prises par les sections d'un paquet de données ;
- un module d'application d'au moins une opération de compression à une valeur de section d'un paquet de données, ladite opération étant comprise dans une règle de compression appartenant à un ensemble de règles de compression obtenu ;
- un module de transmission d'un paquet de données auquel a été appliqué une opération de compression et d'une donnée représentative d'une règle de compression à laquelle appartient ladite opération de compression ;
- un module de réception d'un paquet de données et d'une donnée représentative d'une règle de compression ;
- un module d'application à un paquet de données d'une opération de décompression symétrique d'une opération de compression appliquée à une valeur de section dudit paquet de données.

Selon un mode de réalisation, le système de gestion de la transmission comprend de surcroît :

- un module de collecte dans le réseau de paquets de données transmis pour former un ensemble dit d'apprentissage utilisé par un module d'obtention d'un ensemble de règles de compression ;
- un module de déploiement synchronisé dans les nœuds du réseau d'un ensemble de règles de compression obtenu.

5 Enfin, selon un autre aspect matériel, l'invention a trait à des supports de données sur lesquels sont enregistrés des programmes d'ordinateurs comprenant des séquences d'instructions pour la mise en œuvre des procédés définis ci-dessus.

Les supports de données peuvent être n'importe quelle entité ou dispositif capable de stocker les programmes. Par exemple, les supports peuvent comporter un moyen de stockage, tel qu'une ROM, par
10 exemple un CD ROM ou une ROM de circuit microélectronique, ou encore un moyen d'enregistrement magnétique tel qu'un un disque dur. D'autre part, les supports peuvent être des supports transmissibles tels qu'un signal électrique ou optique, qui peuvent être acheminés via un câble électrique ou optique, par radio ou par d'autres moyens. Les programmes selon l'invention peuvent être en particulier téléchargés sur un
15 réseau de type Internet. Alternativement, le support d'informations peut être un circuit intégré dans lequel le programme est incorporé, le circuit étant adapté pour exécuter ou pour être utilisé dans l'exécution du procédé en question.

Brève description des figures

L'invention sera mieux comprise à la lecture de la description qui suit, donnée à titre d'exemple, et faite en
20 référence aux dessins annexés sur lesquels :

[Fig 1] représente une entité d'obtention d'un ensemble de règles de compression ainsi qu'une entité de gestion de la compression d'un paquet de données transmis dans un réseau et une entité de gestion de la décompression d'un paquet de données.

[Fig 2] représente un exemple d'obtention de dénombrement des valeurs prises par une section en les
25 arrangeant selon un arbre binaire.

[Fig 3] représente un exemple de système de transmission de paquets de données selon l'invention, montrant le déploiement d'un tel système dans un réseau comprenant une passerelle et deux terminaux.

[Fig 4] représente un autre exemple de système de transmission de paquets de données selon l'invention, montrant le déploiement d'un tel système dans un réseau comprenant deux routeurs.

Description détaillée

La figure 1 représente une entité 300 d'obtention d'un ensemble RS de règles de compression ainsi qu'une entité de gestion de la compression 100 d'un paquet de données P transmis dans un réseau NET et une entité de gestion de la décompression 200 d'un paquet de données P.

- 5 L'entité 300 d'obtention d'un ensemble RS de règles de compression comprend un module 301 d'obtention d'un dénombrement D relativement aux valeurs prises par les sections de paquets de données appartenant à un ensemble E de paquets de données, ensemble dit d'apprentissage ainsi qu'un module APP d'obtention de l'ensemble RS de règles de compression en fonctions des dénombrements D obtenus.

10 Dans des exemples de réalisation, le module APP met en œuvre un algorithme d'apprentissage afin d'obtenir l'ensemble RS de règles de compression.

En plus de l'entité d'obtention 300, la figure 1 décrit deux entités de gestion de la compression 100 et de la décompression 200 d'un paquet P de données afin d'illustrer l'utilisation de l'ensemble RS de règles de compression obtenu par la mise en œuvre du procédé par l'entité d'obtention 300.

- 15 Les deux entités de gestion de la compression 100 et de la décompression 200 appartiennent à un même réseau NET qui est un réseau de communication utilisant des protocoles de transmission de paquets de données P. Dans l'exemple de réalisation décrit dans la figure 1, l'entité d'obtention 300 appartient au même réseau NET mais dans d'autres exemples de réalisation, l'entité d'obtention 300 peut appartenir à des réseaux différents ou même être isolée du réseau NET. L'ensemble RS de règles de compression obtenu par l'entité d'obtention 300 pourra être déployé dans les entités de gestion de la compression 100 et de la décompression 200 par tout moyen. Le paquet de données P comprend des sections prenant des valeurs respectives. Les deux entités de gestion de la compression 100 et de la décompression 200 disposent d'un même ensemble de règles de compression RS, cet ensemble comprenant des règles de compression qui comprennent elles-mêmes des opérations de compression qui s'appliquent à des valeurs prises par les sections d'un paquet de données P. L'ensemble RS de règles de compression a été obtenu par la mise en œuvre d'un algorithme d'obtention par l'entité 300.

L'entité de gestion de la compression 100 comprend un module 101 d'application d'au moins une opération de compression à une valeur de section d'un paquet de données P, ladite opération étant comprise dans une règle de compression appartenant à un ensemble de règles de compression RS, ledit ensemble RS étant obtenu par apprentissage.

- 30 L'entité de gestion de la compression 100 comprend un module 102 de transmission d'un paquet de données P auquel au moins une opération de compression a été appliquée et d'une donnée représentative de la règle de compression à laquelle appartient ladite opération.

- 35 L'entité de gestion de la décompression 200 comprend un module 201 de réception d'un paquet de données P auquel au moins une opération de compression a été appliquée à une valeur de section dudit paquet P, ladite opération étant comprise dans une règle de compression appartenant à un ensemble de règles de

compression RS, ledit ensemble RS étant obtenu par apprentissage, et d'une donnée représentative de la règle de compression à laquelle appartient ladite opération.

L'entité de gestion de la décompression 200 comprend un module 202 d'application à un paquet de données P d'au moins une opération de décompression symétrique d'une opération de compression appliquée à une valeur de section dudit paquet P.

La transmission du paquet de données P par le module 102, et la réception du même paquet P par le module 201, est représentée par une flèche reliant le module 102 au module 201, étiquetée par le paquet P. La transmission et la réception de la donnée représentative de la règle de compression appliquée au paquet P ne sont pas représentées dans la figure 1. Dans certains modes de réalisation, le paquet de données P et la donnée représentative sont transmis et reçus conjointement.

Comme l'entité de gestion de la compression 100 et de la décompression 200 disposent du même ensemble de règles de compression RS obtenu, l'entité de gestion de la décompression 200 pourra utiliser la donnée représentative de la règle de compression appliquée au paquet P pour retrouver celle-ci. Il est aisé ensuite de déterminer l'opération symétrique de décompression. Le module 202 d'application d'une opération de décompression symétrique permet alors de rétablir le paquet de données P tel qu'il était avant l'application d'une opération de compression.

Le paquet de données P a donc été transmis dans le réseau après application sur les valeurs prises par les sections du paquet P d'une ou plusieurs opérations de compression par l'entité de gestion de la compression 100. La transmission du paquet P une fois compressé permet de réaliser d'importantes économies de ressources énergétiques et matérielles. Le paquet P est ensuite rétabli par l'entité de gestion de la décompression 200 pour utilisation ultérieure. Comme l'ensemble de règles de compression RS est obtenu par apprentissage, le procédé ne demande pas d'intervention d'un expert, et il peut s'adapter à des contextes changeant en réitérant l'algorithme d'apprentissage pour disposer d'ensembles de règles de compression RS adaptés au contexte.

La figure 2, quant à elle, représente un exemple de dénombrement exact obtenu pour définir une règle de compression en utilisant ces dénombrements.

L'exemple présenté ici utilise un dénombrement des valeurs prises par une section donnée dans l'ensemble dit d'apprentissage. Ce dénombrement exact permet de guider très précisément la définition de règles de compression. Le mode de réalisation présenté ici permet de réaliser ces dénombrements exacts de manière efficace. Dans d'autres exemples de réalisation, il est possible d'utiliser des potentiels de compression, obtenus en utilisation une notion d'entropie, pour guider l'obtention de règles de compression. Ces grandeurs sont moins précises que les dénombrements exacts présentés ici mais peuvent apporter un guidage alternatif ou complémentaire aux dénombrements exacts.

Pour réaliser ce dénombrement, on suppose que les valeurs sont connues comme une suite de bits. Pour les dénombrer, dans notre exemple, on va les arranger selon un arbre binaire.

L'arbre de gauche de la figure 2 donne un exemple de cet arrangement. On s'intéresse ici à une section de longueur fixe. Toutes les valeurs prises par cette section dans l'ensemble dit d'apprentissage E sont des suites de 4 bits. La racine de l'arbre binaire, étiquetée S, correspond au début des suites de bits correspondant aux valeurs prises par la section dans l'ensemble dit d'apprentissage E. Les nœuds de l'arbre binaire sont étiquetés par 0 ou 1. Un chemin dans l'arbre depuis la racine jusqu'à un des nœuds correspond alors à une suite de bits de longueur maximale 4. Les arêtes de l'arbre sont étiquetées par le nombre de valeurs prises par la section dans l'ensemble dit d'apprentissage E dont le préfixe en terme de suite de bits est celui correspondant au nœud auquel mène l'arête.

Pour reprendre l'exemple de l'arbre de gauche de la figure 5, il représente 140 valeurs prises par une section donnée dans l'ensemble dit d'apprentissage. Si l'on regarde les deux premières arêtes issues de la racine de l'arbre S, celles-ci indiquent que, sur ces 140 valeurs, 126 commencent par le bit 1 et 14 commencent par le bit 0. Si l'on regarde ensuite les deux arêtes qui partent du premier nœud 1, celles-ci indiquent que, sur les 126 valeurs dont le premier bit est 1, le second bit vaudra 1 dans 94 cas et 0 dans 32 cas. Si on regarde les deux arêtes partant du premier nœud 0, par contre, on trouve une étiquette 6 menant vers le nœud 1 et une étiquette 8 menant vers le nœud 0. Ces chiffres indiquent que, sur les 14 valeurs dont le premier bit est 0, le second bit vaudra 1 dans 6 cas et vaudra 0 dans les 8 autres cas.

Si on récapitule ces arêtes de profondeur 2, on en déduit que, sur les 140 valeurs prises par la section dans l'ensemble dit d'apprentissage E :

- 94 ont un préfixe 11 ;
- 32 ont un préfixe 10 ;
- 6 ont un préfixe 01 ; et,
- 8 ont un préfixe 00.

L'arbre représente toutes les valeurs prises par la section, et en allant jusqu'au bout de la lecture de l'arbre en profondeur, et en partant du haut, on obtient les résultats suivants :

- la section prend 28 fois la valeur 1111 ;
- la section prend 28 fois la valeur 1100 ;
- la section prend 38 fois la valeur 1101 ;
- la section prend zéro fois la valeur 1100 ;
- la section prend 8 fois les valeurs 1011, 1010, 1001 et 1000 ;
- la section prend 1 fois les valeurs 0111, 0110 et 0101 ;
- la section prend 3 fois la valeur 0100 ;
- la section prend 8 fois la valeur 0011 ;
- et la section prend zéro fois les valeurs 0010, 0001 et 0000.

On vérifie que la somme des valeurs prises nous donne bien 140.

L'arbre de droite de la figure 5 montre ensuite le calcul des dénombrements du nombre de bits que des règles d'effacement de préfixe effaceraient dans l'ensemble dit d'apprentissage E. Le calcul, dans cet exemple, consiste à multiplier les étiquettes des arêtes par leur profondeur, soit dans l'exemple de la figure 3 par un nombre allant de 1 (pour les deux premières arêtes à gauche) à 4 (pour les 16 arêtes arrivant aux 16 nœuds les plus profonds de l'arbre). Le plus fort dénombrement (188) est obtenu par l'arête de profondeur 2 qui correspond aux 94 valeurs prises par la section qui ont le préfixe 11 dans l'ensemble dit d'apprentissage E. Une règle d'effacement du préfixe 11 appliquée aux 140 valeurs prises par cette section serait utilisée 94 fois pour effacer le préfixe 11, ce qui correspondrait à une compression de 188 bits. Ce chiffre est bien le dénombrement du nombre de bits qui serait effacé par la règle d'effacement du préfixe 11 pour cette section dans l'ensemble dit d'apprentissage E.

Ce calcul des dénombrements guide donc efficacement la création d'une règle de compression candidate. En l'occurrence, il s'agira d'une règle d'effacement des préfixes 11 pour les valeurs de la section considérée.

- 15 La figure 3, quant à elle, représente un système SYS complet de transmission de paquets de données dans un réseau NET qui utilise un ensemble RS de règles de compression obtenu par un procédé selon l'invention.

Le système SYS présente une architecture où l'on trouve plusieurs équipements réseau, à savoir une passerelle GW et deux terminaux DVC1 et DVC2. Le système SYS de transmission selon l'invention est formé de ces trois équipements GW, DV1 et DVC2.

- 20 Ces équipements réseau sont reliés par le réseau NET. Il peut s'agir par exemple d'un réseau local où la passerelle GW permet à des terminaux DVC1, DVC2 d'accéder à d'autres réseaux plus larges, tels que le réseau Internet. Les protocoles de communication entre les équipements réseaux GW, DVC1, DVC2 peuvent être de tous types du moment qu'ils se fondent sur la transmission de paquets de données P, P'. Le réseau NET peut donc être un réseau local domestique utilisant une liaison de communication WiFi, ou un réseau de déploiement d'objets connectés de type LPWAN utilisant une liaison de communication LoRa ou tout autre liaison adaptée à des objets connectés ou bien un réseau mobile où la passerelle GW est une station de base et les terminaux DVC1, DVC2 sont des terminaux mobiles.

- 25 La passerelle GW comprend une entité de gestion de la compression 100 ; une entité de gestion de la décompression 200 ; ainsi qu'une entité d'obtention 300 (non représentée sur la figure) qui comprend un module 301 d'obtention d'un dénombrement D (non représenté sur la figure), un module APP d'obtention d'un ensemble RS de règles de compression, un module de collecte COL, et un module de déploiement DEP.

Le terminal DVC1 comprend quant à lui une entité de gestion de la compression 100 et une entité de gestion de la décompression 200.

Le terminal DVC2, quant à lui, ne comprend qu'une entité de gestion de la compression 100 et pas d'entité de gestion de la décompression 200. Il peut avoir été décidé, lors de la construction du système SYS que le terminal DVC2 ne faisait quasi exclusivement qu'émettre des paquets P et n'en recevait presque pas. Pour économiser la taille mémoire nécessaire à la construction du terminal DVC2, il peut donc avoir été décidé de ne pas intégrer une entité de gestion de la décompression 200 dans le terminal DVC2. Il ne sera donc pas possible de transmettre au terminal DVC2 un paquet P compressé selon l'invention, car il ne pourrait pas le décompresser et donc le rétablir.

On voit que le système SYS de transmission comprend bien les modules suivants :

- au moins un module d'obtention APP d'un ensemble de règles de compression RS, présent dans la passerelle GW ;
- au moins un module 101 d'application d'une opération de compression à une valeur de section d'un paquet de données P, le module 101 n'étant pas représenté dans la figure 5 mais compris dans une entité de gestion de la compression 100 présente dans la passerelle GW ou les terminaux DVC1, DVC2 ;
- au moins un module 102 de transmission d'un paquet de données P, le module 102 étant de même compris dans une entité de gestion 100 ;
- au moins un module 201 de réception d'un paquet de données P, le module 201 n'étant pas représenté dans la figure 5 mais compris dans une entité de gestion de la décompression 200 présente dans la passerelle GW ou le terminal DVC1 ;
- au moins un module 202 d'application d'une opération de décompression, le module 202 étant de même compris dans une entité de gestion 200 ;
- au moins un module de collecte COL dans le réseau NET de paquets P, P' pour former un ensemble d'apprentissage E, présent dans la passerelle GW ;
- au moins un module de déploiement DEP synchronisé dans les nœuds du réseau d'un ensemble de règles de compression RS, présent dans la passerelle GW.

La passerelle GW utilise son entité de gestion de la compression 100 pour compresser puis transmettre au terminal DVC1 un paquet de données P. Le paquet P a été compressé en utilisant une règle de compression dont l'identifiant est I. Dans le mode de réalisation illustré ici, l'entité de gestion de la compression transmet de façon conjointe l'identifiant I de la règle de compression et le paquet P compressé, ce qui est illustré par la flèche étiquetée I+P entre l'entité de gestion de la compression 100 appartenant à la passerelle GW et l'entité de gestion de la décompression 200 appartenant au terminal DVC1.

Le terminal DVC2, quant à lui, adresse un paquet compressé P', qui a été compressé grâce à une règle dont l'identifiant est I'. L'entité de gestion de la compression 100 du terminal DVC2 adresse donc une concaténation I'+P' à l'entité de gestion de la décompression 200 de la passerelle GW.

Comme toutes les entités de gestion de la compression 100 et de la décompression 200 présentes dans le système SYS disposent du même ensemble de règles de compression RS, il est facile à l'équipement réseau qui le reçoit de rétablir un paquet compressé P, P' en lui appliquant les opérations de décompression

symétriques des opérations de compression qui ont été appliquées aux valeurs prises par les sections du paquet de données P, P'. Les identifiants I, I' transmis conjointement avec les paquets P, P' permettent de retrouver la règle et donc les opérations de compression qui ont été appliquées.

Dans l'architecture du système SYS présentée dans la figure 5, la passerelle GW dispose d'un module de collecte COL qui permet à la passerelle GW de rassembler les paquets de données P, P' qui transitent par la passerelle GW. Dans une architecture de type réseau local, comme celle présentée ici, la passerelle GW est idéalement placée pour voir passer l'ensemble des paquets P, P' transitant dans le réseau NET.

Les paquets P, P' collectés par le module COL vont permettre de créer un ensemble d'apprentissage E qui sera utilisé par le module d'obtention APP également présent dans la passerelle GW. Le module d'apprentissage APP apprend l'ensemble de règles de compression RS en utilisant des paquets P, P' réellement collectés dans le réseau NET par le module de collecte COL, ce qui est un point permettant d'améliorer fortement la pertinence de l'apprentissage et de l'ensemble de règles de compression RS. Le module de déploiement DEP va ensuite déployer de manière synchronisée l'ensemble de règles de compression RS dans la passerelle GW et les terminaux DVC1, DVC2 du système SYS.

Selon un mode de réalisation, les entités de gestion de la compression 100 et de la décompression 200 peuvent disposer de plusieurs ensembles de règles de compression. Ceci permet de faciliter le déploiement synchronisé de nouveaux ensembles de règles de compression RS en permettant d'utiliser de façon concurrente plusieurs ensembles de règles de compression RS le temps que tous les équipements présents dans le réseau NET aient bien reçu les nouveaux ensembles de règles de compression RS.

Le module de collecte COL va bien sûr transmettre au module d'obtention APP des paquets P, P' rétablis dans leur intégrité, et non pas des paquets compressés, car seuls des paquets rétablis permettent d'apprendre l'information sur les valeurs de section les plus intéressantes à compresser.

L'ensemble dit d'apprentissage E ne sera pas forcément construit avec la totalité des paquets P, P' collectés par le module COL. Des stratégies d'échantillonnage peuvent être utilisées pour construire des ensembles d'apprentissage les plus représentatifs possibles du trafic dans le réseau NET, selon les moments de transmission ou bien selon les actions des terminaux DVC1, DVC2 et de la passerelle GW.

Le système de transmission SYS remplit donc bien l'objectif technique attendu, à savoir l'application de règles de compression très efficaces aux paquets P, P' en utilisant un module d'obtention APP pour réaliser l'apprentissage automatique de l'ensemble de règles de compression RS. Le module de collecte COL pour alimenter le module d'obtention APP avec un ensemble dit d'apprentissage E représentatif du trafic présent dans le réseau NET et le module de déploiement DEP de façon synchronisée de l'ensemble de règles de compression RS obtenu permettent de s'assurer que cette compression très efficace des paquets P, P' est obtenue d'une part sans intervention humaine et d'autre part va s'adapter en cas de changement de contexte dans le réseau NET, par exemple en cas de changement d'adresses IP des équipements réseau GW, DVC1, DVC2 ou en cas de rajout de nouveaux équipements dans le réseau.

Le placement des modules de collecte COL, d'obtention APP et de déploiement DEP dans la passerelle GW permettent de disposer d'une architecture efficace où les fonctions liées à l'apprentissage sont centralisées dans la passerelle GW. Cette architecture est efficace car la passerelle GW voit transiter l'ensemble des paquets du réseau NET et a aussi une liaison directe avec les terminaux DVC1, DVC2 présents dans le réseau NET. Cependant, d'autres architectures sont possibles.

La figure 4, quant à elle, représente un système SYS complet de transmission de paquets de données dans un réseau NET selon l'invention, mais présentant une architecture différente de celle vue dans la figure 3.

Le système SYS comprend ici trois équipements réseaux, à savoir deux routeurs RTR1 et RTR2 et un serveur distinct SRV. Le réseau NET peut être ici par exemple un cœur de réseau IP et les deux routeurs RTR1, RTR2 peuvent former une branche d'un réseau à longue distance reliant deux points. La liaison de communication peut être alors sur une fibre optique à très grand débit, et le protocole sous-jacent peut être par exemple du Gigabit Ethernet souvent utilisé en cœur de réseau, ou tout autre protocole adapté.

Les deux routeurs RTR1, RTR2 comprennent chacun une entité de gestion de la compression 100 et de la décompression 200 qui permettent de s'échanger des paquets P, P', transmis conjointement avec des identifiants I, I' de règles de compression appartenant à l'ensemble RS.

On a vu dans la présentation de la figure 6 comment la transmission des paquets P, P' fonctionnait. On fait juste ici remarquer que le système SYS de transmission comprend un serveur SRV dédié à l'apprentissage de l'ensemble de règles de compression RS. Le routeur RTR1 comprend un module de collecte COL qui va adresser des paquets P1, P2, P3 au serveur SRV. Un module de collecte COL peut être placé dans un seul routeur RTR1 puisque celui-ci voit passer, en émission ou en réception, l'ensemble des paquets transitant sur le réseau NET. Mais on peut aussi avoir plusieurs modules de collecte COL, un dans chaque routeur RTR1, RTR2, selon la praticité de ce déploiement. Si le serveur SRV est colocalisé avec le routeur RTR1 dans un même centre de données ou une même salle réseau, il sera beaucoup plus efficace de ne déployer un module de collecte COL que dans le routeur RTR1 et pas dans le routeur RTR2 qui peut être très éloigné dans le cas d'une transmission à longue distance.

Le serveur SRV est dédié à l'apprentissage de l'ensemble de règles de compression RS. Il comprend une entité d'obtention 300 (non représentée sur la figure) qui comprend un module APP qui va réaliser l'apprentissage proprement dit selon l'invention de l'ensemble de règles de compression RS. Le module de déploiement DEP assure le déploiement synchronisé de l'ensemble de règles de compression RS dans les deux routeurs RTR1 et RTR2.

On voit que l'invention se prête à une utilisation très large, dans toutes sortes d'architectures de réseau. D'autres architectures du système de transmission SYS sont possibles pour s'adapter à d'autres architectures du réseau NET, par exemple une architecture en étoile, ou bien en anneau. De même, les protocoles qui

peuvent être optimisés grâce à la compression présentée ici sont très nombreux, à la fois en cœur de réseau ou pour des réseaux locaux ou pour des réseaux mobiles.

Signalons enfin ici que, dans le présent texte, le terme « module » peut correspondre aussi bien à un composant logiciel qu'à un composant matériel ou un ensemble de composants matériels et logiciels, un composant logiciel correspondant lui-même à un ou plusieurs programmes ou sous-programmes d'ordinateur ou de manière plus générale à tout élément d'un programme apte à mettre en œuvre une fonction ou un ensemble de fonctions telles que décrites pour les modules concernés. De la même manière, un composant matériel correspond à tout élément d'un ensemble matériel (ou hardware) apte à mettre en œuvre une fonction ou un ensemble de fonctions pour le module concerné (circuit intégré, carte à puce, carte à mémoire, etc.).

Revendications

1. Procédé d'obtention d'un ensemble (RS) de règles de compression aptes à compresser un paquet (P) de données transmis dans un réseau (NET), le paquet (P) comprenant des sections prenant des valeurs respectives exprimées sous la forme de suite de bits, le paquet (P) étant apte à être compressé par application de règles de compression aptes à supprimer des bits d'une valeur prise par une section du paquet (P), le procédé comprenant, pour plusieurs règles de compression données, une obtention d'un dénombrement (D), relativement à un ensemble (E) de paquets de données, ensemble dit d'apprentissage, du nombre de bits supprimés par la règle concernée appliquée à des paquets de l'ensemble dit d'apprentissage suivie par l'obtention de l'ensemble (RS) de règles de compression en fonction des dénombrements (D) obtenus.
2. Procédé d'obtention d'un ensemble (RS) de règles de compression selon la revendication 1, caractérisé en ce qu'il comprend les étapes suivantes exécutées lors de l'obtention de l'ensemble (RS) de règles de compression :
 - une étape de sélection d'une règle appartenant à l'ensemble (RS) de règles de compression en utilisant au moins un dénombrement (D) parmi ceux obtenus ;
 - une étape de création d'une nouvelle règle à partir de la règle sélectionnée en utilisant au moins un dénombrement (D) parmi ceux obtenus ;
 - l'ajout de la nouvelle règle à l'ensemble (RS) de règles de compression ;

et en ce que lesdites étapes sont itérées, jusqu'à ce qu'un critère d'arrêt soit satisfait, en utilisant à l'itération suivante l'ensemble (RS) de règles modifié par l'ajout de la nouvelle règle.

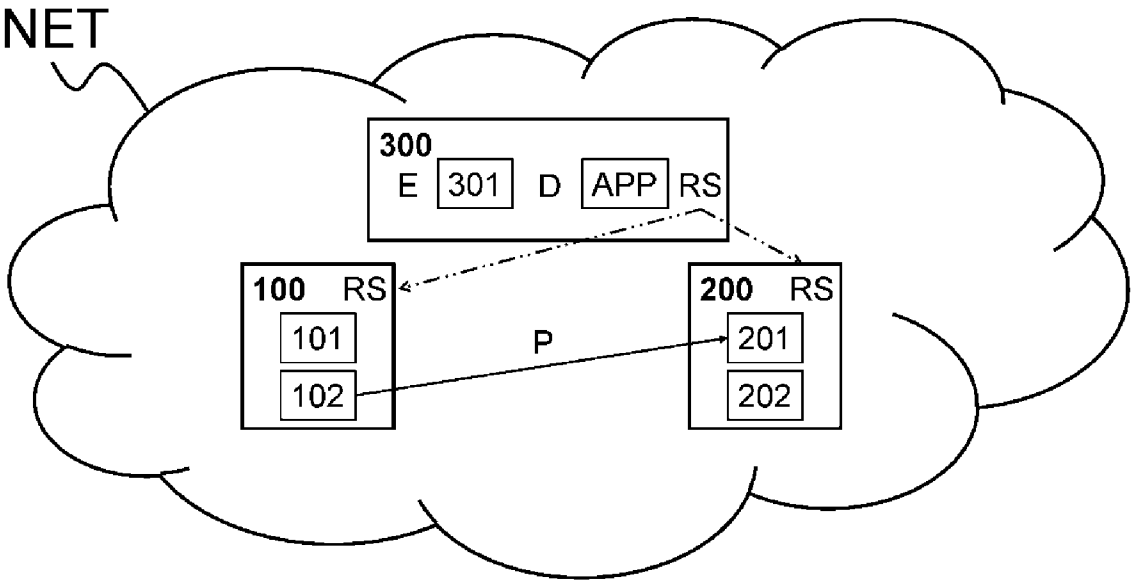
3. Procédé d'obtention d'un ensemble (RS) de règles de compression selon l'une des revendications 1 ou 2, caractérisé en ce que l'étape d'obtention d'un dénombrement (D) comprend, pour une section du paquet (P) prenant des valeurs respectives exprimées sous la forme de suite de bits, le dénombrement du nombre de bits qui seraient effacés à la suite de l'application d'une règle d'effacement de préfixe sur les valeurs prises par la section dans l'ensemble dit d'apprentissage (E).
4. Procédé d'obtention d'un ensemble (RS) de règles de compression selon la revendication 3, caractérisé en ce que l'étape d'obtention d'un dénombrement (D) comprend, pour une section du paquet (P) prenant des valeurs respectives exprimées sous la forme de suite de bits, les étapes suivantes :
 - une création d'un arbre binaire représentant l'ensemble des valeurs pouvant être prises par une suite de bits, la racine de l'arbre représentant une suite vide de bits, les nœuds de l'arbre valant 0 ou 1, et un nœud de l'arbre de profondeur n représentant une suite de bits de longueur n dont la valeur est celle de la suite de 0 et de 1 rencontrés depuis la racine de l'arbre pour rejoindre le dit nœud de profondeur n , une arête dudit arbre étant étiquetée par le nombre de valeurs prises par la section ayant comme préfixe la suite de bits représentée par le nœud auquel mène l'arête ;
 - une multiplication des étiquettes des arêtes de l'arbre binaire créé par la profondeur du nœud auquel mène l'arête, ce qui réalise un dénombrement des bits qui seraient effacés dans l'ensemble dit

d'apprentissage (E) par l'application de règles d'effacement de préfixes des valeurs prises par la section exprimées sous forme de suite de bits ;

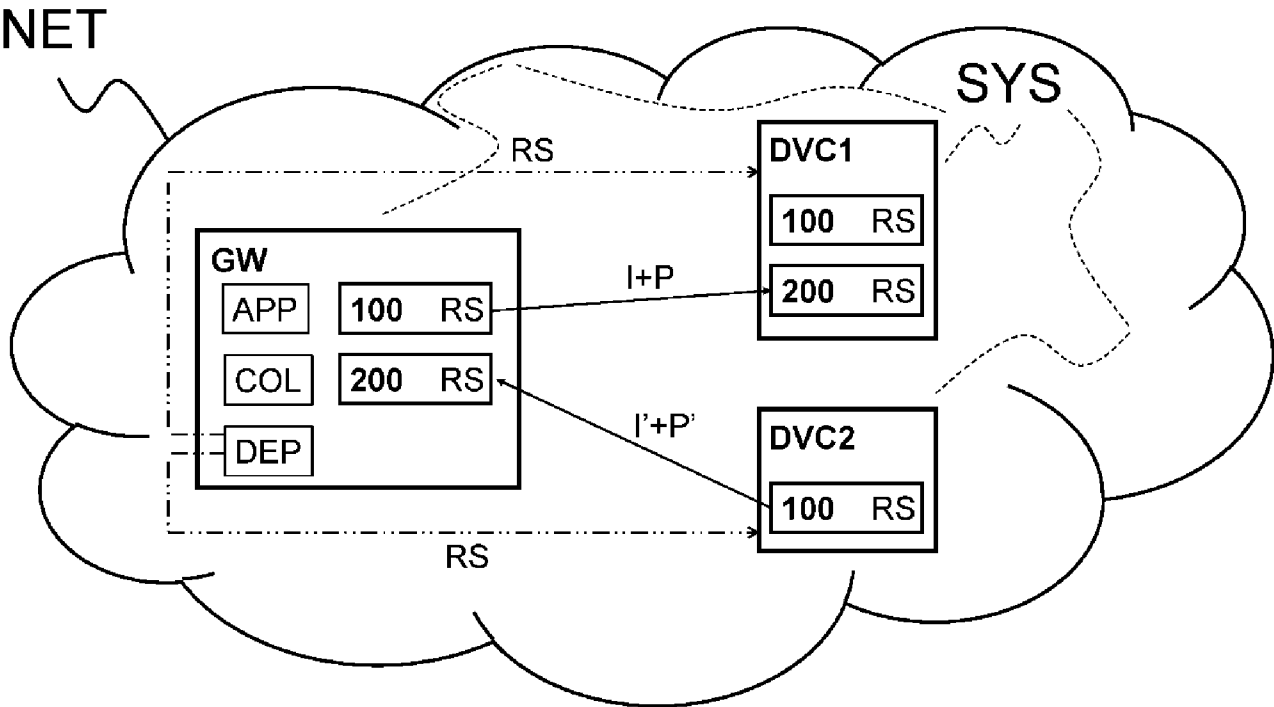
et caractérisé en ce que l'étape de création d'une nouvelle règle comprenne la création d'une règle d'effacement de préfixe pour les valeurs prises par une section exprimées sous forme de suite de bits, le préfixe effacé étant celui pour lequel le dénombrement des bits effacés est le plus grand.

5. Entité de gestion (300) de l'obtention d'un ensemble (RS) de règles de compression aptes à compresser un paquet (P) de données transmis dans un réseau (NET), le paquet (P) comprenant des sections prenant des valeurs respectives exprimées sous la forme de suite de bits, le paquet (P) étant apte à être compressé par application de règles de compression aptes à supprimer des bits d'une valeur prise par une section du paquet (P), caractérisé en ce que l'entité de gestion (300) comprend les modules suivants :
 - un module (301) d'obtention, pour plusieurs règles de compression données, d'un dénombrement (D), relativement à un ensemble (E) de paquets de données, ensemble dit d'apprentissage, du nombre de bits supprimés par la règle concernée appliquée à des paquets de l'ensemble dit d'apprentissage ;
 - un module (APP) d'obtention de l'ensemble (RS) de règles de compression en fonction des dénombrements (D) obtenus.
6. Entité de gestion (300) selon la revendication 5 comprenant de surcroît les modules suivants :
 - un module (COL) de collecte dans le réseau (NET) de paquets de données transmis pour former un ensemble d'apprentissage (E) utilisé pour obtenir l'ensemble (RS) de règles de compression ;
 - un module (DEP) de déploiement synchronisé dans les nœuds du réseau (NET) de l'ensemble (RS) de règles de compression
7. Equipement réseau (GW, DVC1, DVC2, RTR1, RTR2) comprenant une entité de gestion (300) de l'obtention d'un ensemble (RS) de règles de compression selon la revendication 5.
8. Programme d'ordinateur apte à être mis en œuvre par une entité de gestion (300), le programme comprenant des instructions de code qui, lorsqu'il est exécuté par un processeur, réalise les étapes du procédé d'obtention défini dans la revendication 1.
9. Support de données sur lequel est enregistré un programme d'ordinateur comprenant une séquence d'instructions de code pour la mise en œuvre du procédé d'obtention conforme à la revendication 1 lorsqu'il est chargé dans et exécuté par un processeur.

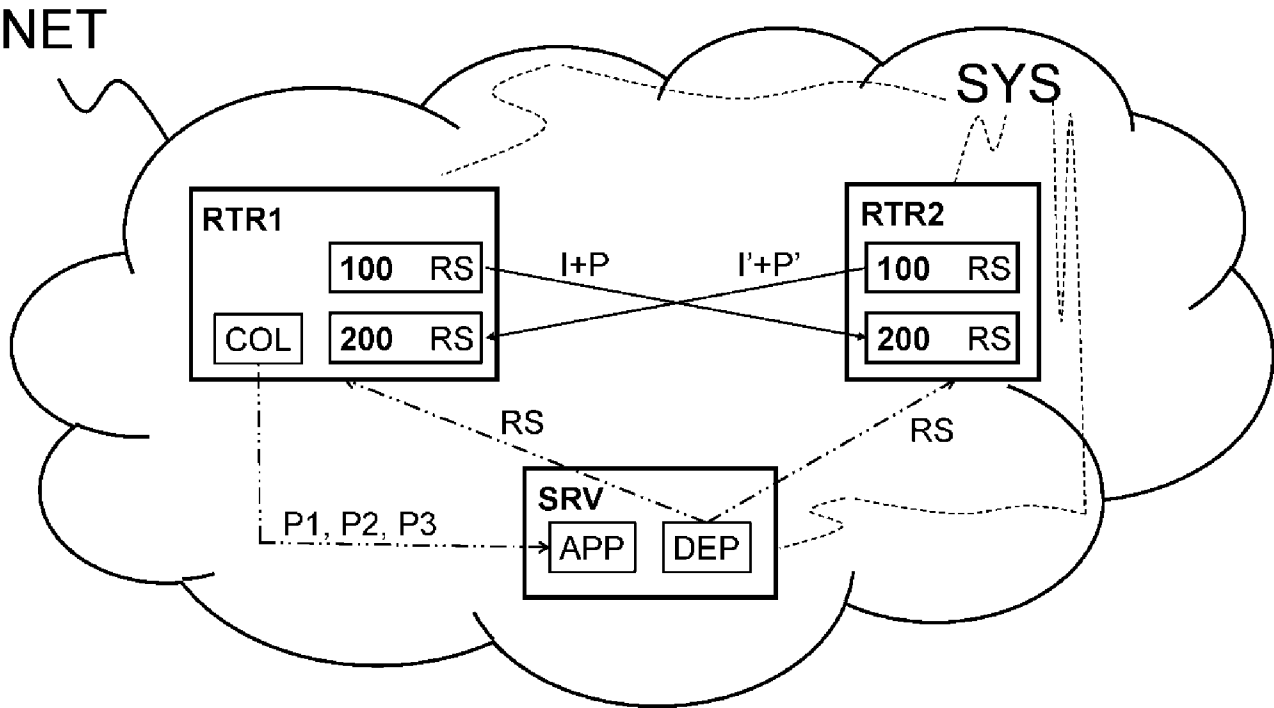
[Fig 1]



[Fig 3]



[Fig 4]



INTERNATIONAL SEARCH REPORT

International application No.

PCT/EP2024/065212

A. CLASSIFICATION OF SUBJECT MATTER H04L 69/04 (2022.01)i; G06N 20/00 (2019.01)i; H03M 7/30 (2006.01)i; G06N 3/006 (2023.01)i According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04L; G06N; H03M Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	TOMOSKOZI MATE ET AL. "Packet Header Compression: A Principle-Based Survey of Standards and Recent Research Studies" <i>IEEE COMMUNICATIONS SURVEYS & TUTORIALS, IEEE</i> , Vol. 24, No. 1, 20 January 2022 (2022-01-20), pages 698-740 DOI: 10.1109/COMST.2022.3144473 XP011901225	1-3,5-9
A	sections I.A, II.B.2)-6), II.C, II.F, III.H	4
A	US 2021110292 A1 (MINABURO ANA [FR] ET AL) 15 April 2021 (2021-04-15) paragraphs [0018] - [0131], [0147]; claims 1,4; figures	1-9
☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.		
* Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed “T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family		
Date of the actual completion of the international search 09 September 2024		Date of mailing of the international search report 19 September 2024
Name and mailing address of the ISA/EP European Patent Office p.b. 5818, Patentlaan 2, 2280 HV Rijswijk Netherlands (Kingdom of the) Telephone No. (+31-70)340-2040 Facsimile No. (+31-70)340-3016		Authorized officer Scappazzoni, E Telephone No.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/EP2024/065212

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2021110292	A1	15 April 2021	CN	110651457	A	03 January 2020
				EP	3603000	A1	05 February 2020
				ES	2864161	T3	13 October 2021
				FR	3064865	A1	05 October 2018
				US	2021110292	A1	15 April 2021
				WO	2018178242	A1	04 October 2018

A. CLASSEMENT DE L'OBJET DE LA DEMANDE INV. H04L69/04 G06N20/00 H03M7/30 G06N3/006 ADD.		
Selon la classification internationale des brevets (CIB) ou à la fois selon la classification nationale et la CIB		
B. DOMAINES SUR LESQUELS LA RECHERCHE A PORTE Documentation minimale consultée (système de classification suivi des symboles de classement) H04L G06N H03M		
Documentation consultée autre que la documentation minimale dans la mesure où ces documents relèvent des domaines sur lesquels a porté la recherche		
Base de données électronique consultée au cours de la recherche internationale (nom de la base de données, et si cela est réalisable, termes de recherche utilisés) EPO- Internal		
C. DOCUMENTS CONSIDERES COMME PERTINENTS		
Catégorie*	Identification des documents cités, avec, le cas échéant, l'indication des passages pertinents	no. des revendications visées
X	TOMOSKOZI MATE ET AL: "Packet Header Compression: A Principle-Based Survey of Standards and Recent Research Studies", IEEE COMMUNICATIONS SURVEYS & TUTORIALS, IEEE, vol. 24, no. 1, 20 janvier 2022 (2022-01-20), pages 698-740, XP011901225, DOI: 10.1109/COMST.2022.3144473	1 - 3, 5 - 9
A	Sections I.A, II.B.2) - 6), II.C, II.F, III.H	4
A	US 2021/110292 A1 (MINABURO ANA [FR] ET AL) 15 avril 2021 (2021-04-15) alinéas [0018] - [0131], [0147]; revendications 1,4; figures	1 - 9
☐ Voir la suite du cadre C pour la fin de la liste des documents ☒ Les documents de familles de brevets sont indiqués en annexe		
* Catégories spéciales de documents cités: "A" document définissant l'état général de la technique, non considéré comme particulièrement pertinent "E" document antérieur, mais publié à la date de dépôt international ou après cette date "L" document pouvant jeter un doute sur une revendication de priorité ou cité pour déterminer la date de publication d'une autre citation ou pour une raison spéciale (telle qu'indiquée) "O" document se référant à une divulgation orale, à un usage, à une exposition ou tous autres moyens "P" document publié avant la date de dépôt international, mais postérieurement à la date de priorité revendiquée "T" document ultérieur publié après la date de dépôt international ou la date de priorité et n'appartenant pas à l'état de la technique pertinent, mais cité pour comprendre le principe ou la théorie constituant la base de l'invention "X" document particulièrement pertinent:: l'invention revendiquée ne peut être considérée comme nouvelle ou comme impliquant une activité inventive par rapport au document considéré isolément "Y" document particulièrement pertinent:: l'invention revendiquée ne peut être considérée comme impliquant une activité inventive lorsque le document est associé à un ou plusieurs autres documents de même nature, cette combinaison étant évidente pour une personne du métier "&" document qui fait partie de la même famille de brevets		
Date à laquelle la recherche internationale a été effectivement achevée		Date d'expédition du présent rapport de recherche internationale
9 septembre 2024		19/09/2024
Nom et adresse postale de l'administration chargée de la recherche internationale Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Fonctionnaire autorisé Scappazzoni, E

RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Demande internationale n°

PCT/EP2024/065212

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
US 2021110292 A1	15-04-2021	CN 110651457 A	03-01-2020
		EP 3603000 A1	05-02-2020
		ES 2864161 T3	13-10-2021
		FR 3064865 A1	05-10-2018
		US 2021110292 A1	15-04-2021
		WO 2018178242 A1	04-10-2018
