(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
22 janvier 2026 (22.01.2026)

WIPO | PCT

(10) Numéro de publication internationale
WO 2026/017825 A1

(51) Classification internationale des brevets :
H04L 69/04 (2022.01) *H04L 69/22* (2022.01)
H03M 7/30 (2006.01)

(21) Numéro de la demande internationale :
PCT/EP2025/070564

(22) Date de dépôt international :
17 juillet 2025 (17.07.2025)

(25) Langue de dépôt : français

(26) Langue de publication : français

(30) Données relatives à la priorité :
FR2407944 19 juillet 2024 (19.07.2024) 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). **LARUE, Guillaume** ; ORANGE - Intellectual Property and Licensing, Orange Gardens - 44 Avenue de la République - CS 50010, 92326 Chatillon (FR). **DUFRENE, Louis Adrien** ; 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, 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, UY, 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,

(54) Title: MANAGEMENT OF THE TRANSMISSION AND RECEPTION OF A COMPRESSED DATA PACKET

(54) Titre : GESTION DE LA TRANSMISSION ET DE LA RECEPTION D'UN PAQUET DE DONNEES COMPRESSE

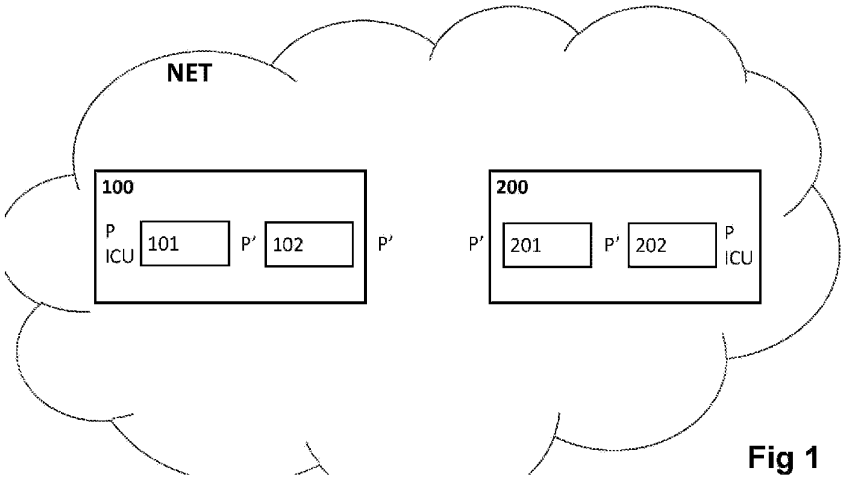


Fig 1

(57) Abstract: The invention relates to a method for managing the transmission of a data packet compressed by applying a compression rule belonging to a set of compression rules, referred to as the compressed packet (P), comprising the insertion into the compressed packet (P) of an identifier of the set of compression rules, referred to as the unique context identifier (ICU), followed by the transmission of the thus modified compressed packet (P').

(57) Abrégé : L'invention se rapporte à un procédé de gestion de la transmission d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé (P), comprenant l'insertion dans ledit paquet compressé (P) d'un identifiant dudit ensemble de règles de compression, dit identifiant de contexte unique (ICU), suivie par la transmission du paquet compressé ainsi modifié (P').

WO 2026/017825 A1

[Suite sur la page suivante]

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))*

— *en noir et blanc ; la demande internationale telle que déposée était en couleur ou en échelle de gris et est disponible sur PATENTSCOPE pour téléchargement.*

Description

GESTION DE LA TRANSMISSION ET DE LA RECEPTION D'UN PAQUET DE DONNEES COMPRESSE

Domaine technique

[0001] Le domaine technique est celui de la compression des données transmises dans un réseau de communication.

[0002] Plus précisément, l'invention se rapporte à un procédé de transmission d'un paquet de données compressé dans un réseau de communication. L'invention se rapporte également à un procédé de réception d'un paquet de données compressé.

[0003] L'informatique en réseau et 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 un message donné est envoyé par l'émetteur vers un récepteur, chaque paquet correspondant à une partie du message complet. Les paquets contiennent l'adresse du récepteur. Grâce à cette information, un paquet peut être adressé à son destinataire final grâce à différents algorithmes de routage. Le contenu global du message est reconstruit par le destinataire en regroupant les données contenues dans les différents paquets. Le message peut aussi être un média sonore ou visuel dans le cas des communications téléphoniques ou audiovisuelles qui fonctionnent aussi selon le principe de l'envoi de 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.

[0004] 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 utilisent 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 utilise un protocole de la couche Liaison, par exemple le protocole Ethernet qui organise 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.

[0005] D'autres protocoles de la couche Application utilisent 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.

[0006] 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. Dans le cas où les paquets d'un protocole donné contiennent un ou des paquets encapsulés d'un autre protocole, la charge d'un paquet du premier protocole peut contenir l'en-tête d'un autre protocole. 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.

[0007] 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 remplissage (traduction de l'anglais *padding bits*) vont compléter la valeur jusqu'à la longueur fixe du champ. On trouve également le terme de bits de bourrage (autre traduction de l'anglais *padding bits*). 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 contient 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.

[0008] 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, 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 est constitué de plusieurs en-têtes concaténés, un pour chaque protocole encapsulé. Par exemple, une trame Ethernet encapsulant un paquet IP commence par l'en-tête IP, suivi de l'en-tête UDP. Un datagramme UDP porté par une trame Ethernet présente alors successivement l'en-tête Ethernet suivie de l'en-tête IP puis de l'en-tête UDP.

[0009] Il existe un besoin clair de compression des données échangées en informatique et dans les télécommunications numériques pour économiser au mieux les ressources matérielles et en énergie consommées par les réseaux informatiques ou de télécommunications. Ce besoin explique la création de protocoles dédiés à la compression des paquets de données échangées. On connaît comme exemples de protocoles de compression les protocoles 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, ainsi que le protocole SCHC (acronyme de l'anglais *Static Context Header Compression*), défini par l'IETF dans la RFC 8724.

[0010] Ces protocoles fonctionnent grâce à des règles de compression qui s'appliquent à un paquet de données particulier. Une règle de compression indique d'une part les conditions à remplir par un paquet de données pour que la règle s'applique à ce paquet et d'autre part les compressions réalisées quand la règle s'applique à un paquet. Les conditions d'application d'une règle portent sur les valeurs prises par les différentes sections de ce paquet donné. Selon des règles de compression utilisées dans des protocoles de compression tels que ROHC ou SCHC, la condition d'application porte sur une valeur prise par une section d'un paquet et la compression réalisée consiste à effacer cette valeur. Le récepteur, connaissant la règle de compression appliquée, rétablit la valeur effacée à la réception du paquet. Selon d'autres règles, un index est transmis, cet index correspondant à une valeur prise par une section donnée. Lors de la réception, le récepteur rétablit la valeur qui a été remplacée par l'index en consultant un dictionnaire qui fait partie de la règle de compression appliquée. Selon d'autres

règles, la condition porte sur un préfixe de la valeur exprimée comme suite de bits et la compression consiste en l'effacement de ce préfixe. Le récepteur, là encore, rétablit le préfixe effacé quand il procède à la décompression du paquet. On obtient ainsi une compression très efficace des paquets de données même si la compression se limite aux en-têtes.

[0011] Dans tous les cas, le fonctionnement de protocoles de compression de paquets de données repose sur l'existence d'ensembles de règles de compression qui décrivent les opérations de compression à mener sur des valeurs de champs d'en-têtes ou plus généralement de sections des paquets de données. Ces opérations de compression sont déclenchées quand les paquets remplissent des conditions portant sur les valeurs prises par les sections des paquets. On dit d'une règle de compression qu'elle s'applique à un paquet quand toutes les opérations de compression présentes dans la règle peuvent s'appliquer aux valeurs prises par les champs du paquet en question. Lorsqu'un paquet de données est compressé par une règle donné, il est transmis en association avec un identifiant de la règle de compression appliquée. De cette manière, le récepteur peut retrouver la règle de compression appliquée au paquet, pouvant ainsi reconstruire le paquet original en réalisant les opérations inverses à la compression. Dans certains cas, l'identifiant de la règle de compression utilisée n'est pas transmis, et la règle de compression utilisée est déduite d'une autre information. Par exemple, si dans un ensemble de règles de compression, une règle réalise une compression beaucoup plus forte que les autres, et qu'un paquet reçu est d'une taille très faible, cette indication peut suffire à déduire que c'est cette règle très efficace qui a été appliquée, même sans transmission de l'identifiant de règle. Le récepteur et l'émetteur doivent dans tous les cas avoir une connaissance commune des règles de compression susceptibles d'être appliquées et de la règle effectivement appliquée pour compresser un paquet, afin de pouvoir le rétablir à la réception.

[0012] L'ensemble de ces éléments sont bien connus dans l'état de la technique et ne seront pas décrits plus avant.

État de la technique

[0013] Comme indiqué précédemment, un protocole de compression de données nécessite que l'ensemble de règles de compression susceptibles d'être appliquées soit connu à la fois de l'émetteur et du récepteur pour pouvoir assurer la compression puis la décompression des paquets transmis. Quand un émetteur applique une règle de compression à un paquet de données avant de le transmettre, il va attacher au paquet transmis l'identifiant de la règle de compression utilisée. Le récepteur, qui dispose du même ensemble de règles de compression que l'émetteur, retrouve la règle de compression appliquée dans l'ensemble de règles de compression grâce à son

identifiant et reconstitue le paquet original en appliquant les opérations inverses de celles qui forment la règle de compression appliquée.

[0014] Ces éléments sont bien établis dans les protocoles de compression, et notamment dans les protocoles ROHC et SCHC. Dans le protocole ROHC, la compression se fait dans le contexte des télécommunications mobiles entre une station de base et des terminaux mobiles, et les règles de compression, définies par le protocole ROHC, sont connues à la fois des émetteurs et des récepteurs. Dans le cas du protocole SCHC, le contexte d'utilisation est celui de l'Internet des objets (traduction de l'anglais *Internet of Things*, qui donne l'acronyme IoT) ou des réseaux étendus à basse consommation (traduction de l'anglais *Low Power Wide Area Networks*, qui donne l'acronyme LPWAN). Dans ce contexte, une passerelle va communiquer avec les objets appartenant au réseau créé par la passerelle et partager avec eux un même ensemble de règles de compression.

[0015] A ce jour, il n'est pas possible d'utiliser ces techniques de compression dans des réseaux plus larges, tels que le réseau Internet. En effet, dans des réseaux plus larges, des équipements tels que des routeurs servent de passerelles entre des réseaux distincts. Un paquet de données transite de réseau en réseau avant d'arriver à sa destination. Une communication se fait en passant de réseau en réseau et est dite alors multi-saut (traduction de l'anglais *multi-hop*). Un équipement donné, dans ce cadre, communique donc avec d'autres équipements dans plusieurs contextes distincts, un contexte correspondant à un réseau donné. De plus, les contextes changent avec le temps ; un équipement tel qu'un routeur peut être rattaché à un moment donné à un ensemble d'autres routeur et partager avec eux des ensembles de règles de compression, ensembles qui devront changer au fur et à mesure que d'autres routeurs se connectent ou se séparent du routeur initial. Les compressions qui seront utiles dans un contexte pourront ne plus être du tout adaptées dans un autre contexte. Un équipement donné devra donc pouvoir utiliser pour effectuer ses compressions de paquets des ensembles de règles de compression distincts qui correspondent chacun à un contexte donné.

[0016] Or, les identifiants de règles de compression qui sont en général transmis avec un paquet compressé sont uniques seulement dans un contexte donné et dans l'ensemble de règles de compression associé. Mais il n'y a aucune garantie qu'un même identifiant de règle de compression ne se retrouve pas dans plusieurs ensembles de règles de compression correspondant à plusieurs contextes distincts.

[0017] Il n'y a donc pas de possibilités à l'heure actuelle d'utiliser les règles de compression efficaces des protocoles de compression de données tels que ROHC ou SCHC dans un contexte où les équipements de réseau qui doivent appliquer les règles de compression appartiennent à plusieurs réseaux séparés, évolutifs, et où, donc, les équipements de

réseau doivent utiliser plusieurs ensembles de règles de compression correspondant chacun à des contextes distincts.

[0018] L'invention vient améliorer la situation.

Exposé de l'invention

[0019] Selon un premier aspect fonctionnel, l'invention a trait à un procédé de gestion de la transmission d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé, comprenant l'insertion dans ledit paquet compressé d'un identifiant dudit ensemble de règles de compression, dit identifiant de contexte unique, suivie par la transmission du paquet compressé ainsi modifié.

[0020] Grâce à l'invention, il devient possible d'utiliser des compressions très efficaces dans un environnement où les équipements de télécommunications appartiennent à plusieurs contextes différents et contribuent à des communications multi-saut, de réseau en réseau. Dans un contexte donné, un ensemble de règles de compression particulier est utilisé par un équipement donné pour compresser un paquet de données avant de l'adresser à un autre équipement qui fait partie du même contexte et a en mémoire l'ensemble de règles de compression particulier. Un équipement donné va donc manipuler plusieurs ensembles de règles de compression distincts, correspondants chacun à un contexte donné. Un identifiant de contexte unique (terme qui peut être traduit en anglais *unique context identifier*) peut être affecté à un ensemble de règles de compression donné, identifiant qui sera le même pour les différents équipements qui font partie du même contexte et utilisent ce même ensemble de règles de compression. Les méthodes permettant de créer et d'affecter cet identifiant de contexte unique sont en dehors de l'exposé de la présente invention. L'invention concerne le bon fonctionnement de la compression d'un paquet de données dans un environnement où plusieurs contextes différents coexistent et sont connus dans un même équipement. L'invention permet de faire fonctionner la compression dans un environnement multi-contextes et multi-saut en assurant qu'un identifiant de l'ensemble de règles utilisé soit transmis dans le paquet de données compressé. Cet identifiant est dit identifiant de contexte unique, mais il ne sera pas forcément unique pour tous les contextes possibles dans tous les réseaux. Il doit simplement être unique dans un voisinage proche d'un équipement de réseau donné, plus précisément dans son voisinage immédiat, et le voisinage de ce voisinage. De cette manière, il est possible d'éviter les confusions de contexte dans les communications multi-saut autour de l'équipement de réseau en question. Le récepteur peut ensuite utiliser cet identifiant de contexte pour identifier l'ensemble de règles de compression qui a été utilisé pour compresser le paquet de données. Il peut ensuite reconstruire le contenu original du paquet compressé en

utilisant l'ensemble de règles de compression ainsi identifié. L'invention résout donc bien le problème de l'utilisation des techniques de compression efficaces dans un environnement multi-contexte, pour des communications multi-saut.

[0021] Selon un mode de mise en œuvre particulier de l'invention, le paquet compressé comprend des bits de remplissage et l'identifiant de contexte unique est inséré dans les bits de remplissage du paquet compressé.

[0022] L'objectif de l'invention est de transmettre des paquets de données compressés afin de réaliser une économie maximale de transmission de données. Comme l'invention repose sur la transmission d'une information supplémentaire, à savoir l'identifiant de contexte unique, il faut s'assurer que cette transmission se fait de façon économique. Dans ce mode de réalisation, l'objectif d'économie maximale des données transmises est réalisé. Pour cela, l'invention utilise des bits de remplissage, qui sont d'habitude inutilisés, pour transmettre l'information supplémentaire de l'identifiant de contexte unique. De cette manière, la réalisation de l'invention ne nécessite pas d'utiliser davantage de bits transmis que ceux qui devaient l'être de toute façon. L'invention réalise également une optimisation des ressources réservées pour la transmission en utilisant des bits de remplissage qui sont transmis dans tous les cas et consomment donc des ressources indépendamment de la mise en œuvre ou non de ce mode de réalisation.

[0023] Selon un mode de mise en œuvre particulier de l'invention, qui pourra être mis en œuvre cumulativement avec le mode précédent, l'insertion de l'identifiant de contexte unique dans les bits de remplissage du paquet compressé se fait en partant du dernier bit du paquet compressé selon une permutation définie.

[0024] Ce mode de réalisation permet d'utiliser effectivement les bits de remplissage du paquet compressé. En effet, il n'est pas possible de connaître à l'avance le nombre de bits de remplissage du paquet compressé. Il n'est donc pas possible, pour un paquet compressé donné, de connaître le point de départ des bits de remplissage dans le sens standard de lecture, de gauche à droite, d'un paquet compressé. En revanche, les bits de remplissage se trouvent nécessairement, quand ils existent, au dernier bit du paquet compressé. Pour être sûr de partir d'un point de départ connu lors de l'insertion de l'identifiant de contexte unique, cette insertion se fait en partant du dernier bit du paquet compressé, selon une permutation définie. La permutation définie sur l'identifiant de contexte unique présenté comme une suite de bits permet d'affecter chaque bit qui forme l'identifiant de contexte unique à un bit de remplissage, en partant du dernier bit du paquet compressé, où se trouvent lesdits bits de remplissage. De cette manière, à la réception du paquet compressé, le point de départ de l'insertion de l'identifiant de contexte unique dans les bits de remplissage est bien connu par le récepteur. En utilisant une permutation, au sens mathématique du terme, chaque bit

de la suite de bits formant l'identifiant de contexte unique a bien une place réservée et distincte des autres dans les bits de remplissage du paquet compressé. Tous les bits de l'identifiant de contexte unique peuvent donc se retrouver insérés dans les bits de remplissage grâce à l'utilisation d'une permutation plutôt qu'en utilisant une fonction quelconque.

[0025] Selon un mode de mise en œuvre particulier de l'invention, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'insertion de l'identifiant de contexte unique dans les bits de remplissage du paquet compressé se fait en insérant l'identifiant de contexte unique dans les bits de remplissage en sens contraire au sens de lecture du paquet compressé.

[0026] Dans ce mode de réalisation, comme précédemment, l'insertion de l'identifiant de contexte unique dans les bits de remplissage se fait en partant du dernier bit du paquet compressé pour être sûr de bien insérer l'identifiant dans des bits de remplissage. Ici, le choix fait est d'insérer l'identifiant en l'insérant à partir du dernier bit du paquet compressé en sens inverse à celui du paquet compressé. Cette manière de procéder correspond à une permutation définie sur les bits formant l'identifiant de contexte unique, dans laquelle le premier bit de l'identifiant est affecté au dernier bit du paquet compressé, le deuxième bit de l'identifiant à l'avant-dernier, et ainsi de suite, en sens inverse de lecture du paquet compressé.

[0027] Selon un mode de mise en œuvre particulier de l'invention, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'insertion de l'identifiant de contexte unique dans les bits de remplissage du paquet compressé se fait en partant du dernier bit du paquet compressé et en insérant l'identifiant de contexte unique dans les bits de remplissage en partant de la fin de l'identifiant, dans le sens de lecture du paquet compressé.

[0028] Ce mode de réalisation propose de mettre en œuvre une autre permutation pour insérer l'identifiant de contexte unique dans les bits de remplissage. Ici, l'insertion se fait en partant du dernier bit du paquet compressé et en y insérant l'identifiant en partant de la fin de celui-ci. C'est-à-dire que le dernier bit de l'identifiant est inséré dans le dernier bit du paquet compressé, l'avant-dernier bit de l'identifiant inséré dans l'avant-dernier bit du paquet compressé et ainsi de suite.

[0029] Les deux modes de réalisation précédents insèrent l'identifiant de contexte unique dans les bits de remplissage de façon continue, un bit de l'identifiant après l'autre, sans faire de saut. Cette solution a l'avantage de la simplicité mais un mode de réalisation de l'invention peut utiliser n'importe quelle permutation, y compris en réalisant des sauts dans l'insertion, qui seront remplis à la fin de la permutation.

[0030] Selon un mode de réalisation, qui pourra être réalisé alternativement ou cumulativement avec les modes précédents, l'identifiant de contexte unique est répété

jusqu'à épuisement des bits de remplissage lorsque le nombre de bits de remplissage est supérieur à la taille de l'identifiant de contexte unique.

[0031] Grâce à ce mode de réalisation, l'ensemble des bits de remplissage est utilisé pour transmettre l'identifiant de contexte unique. En le répétant, la transmission est rendue plus robuste en cas d'erreur de transmission, tout en ne consommant pas davantage de bits dans la transmission, puisque l'insertion de l'identifiant de contexte unique est toujours limitée aux bits de remplissage qui devaient de toute façon être transmis.

[0032] Selon un mode de réalisation, qui pourra être réalisé alternativement ou cumulativement avec les modes précédents, l'identifiant de contexte unique est tronqué lors de l'insertion lorsque le nombre de bits de remplissage est inférieur à la taille de l'identifiant de contexte unique.

[0033] Grâce à ce mode de réalisation, l'insertion de l'identifiant de contexte unique ne se fait pas sur les bits du paquet de données qui ne sont pas des bits de remplissage. De cette manière, les informations présentes dans le paquet compressé sont bien préservées et ne sont pas effacées pour transmettre l'identifiant de contexte unique. Ce mode de réalisation prend en compte la taille variable des bits de remplissage. Il n'est pas possible de garantir que, pour tous les paquets compressés, il y aura suffisamment de bits de remplissage pour insérer la totalité de l'identifiant de contexte unique à leur place.

[0034] Selon un mode de mise en œuvre particulier de l'invention, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, l'insertion de l'identifiant de contexte unique comprend de surcroît une opération de combinaison des bits du paquet compressé autres que les bits de remplissage avec une séquence de bits construite en répétant l'identifiant de contexte unique.

[0035] Grâce à ce mode de réalisation, une information supplémentaire est utilisée pour faciliter la transmission de l'identifiant de contexte unique. En effet, le nombre de bits de remplissage du paquet compressé est variable et peut même être nul. Il n'y a donc pas de garantie qu'il sera toujours possible d'inscrire en totalité l'identifiant de contexte unique dans les bits de remplissage. En combinant les bits du paquet compressé en dehors des bits de remplissage avec une séquence de bits construite en répétant l'identifiant de contexte unique, une information supplémentaire est transmise qui permettra au récepteur de retrouver plus aisément l'identifiant de contexte unique inséré dans les bits de remplissage ou combiné avec les autres bits du paquet compressé. Dans ce mode, l'identifiant de contexte unique est combiné avec les bits autres que les bits de remplissage et non pas inséré à leur place. En effet, l'information présente dans les bits autres que les bits de remplissage ne doit pas être effacée. C'est donc une combinaison, qui permettra de retrouver à la fois l'information

initiale du paquet compressé ainsi que l'identifiant de contexte unique qui est réalisée, éventuellement en procédant à des hypothèses.

[0036] Les opérations d'insertion de l'identifiant de contexte unique dans les bits de remplissage d'une part et de combinaison des bits qui ne sont pas de remplissage avec une séquence construite en répétant l'identifiant de contexte unique d'autre part peuvent être réalisées l'une après l'autre, dans n'importe quel ordre, ou bien en parallèle. En effet, ces deux opérations travaillent sur des parties disjointes du paquet compressé. L'exécution des deux opérations de façon séquentielle est plus simple alors que l'exécution parallèle a l'avantage de gagner du temps d'exécution.

[0037] Selon un mode de réalisation, l'opération de combinaison utilise un ou exclusif entre les bits du paquet compressé et ceux de la séquence de bits construite.

[0038] Grâce à ce mode de réalisation, l'identifiant de contexte unique est combiné avec les bits du paquet compressé qui ne sont pas des bits de remplissage d'une façon qui permet de retrouver aisément les valeurs originales des bits du paquet compressé avant la combinaison. En effet, appliquer une nouvelle opération de ou exclusif avec l'identifiant de contexte unique permet de revenir à la valeur originale des bits du paquet compressé. De cette manière, il est possible de retrouver les informations originales en procédant par essais et erreurs.

[0039] Selon un deuxième aspect fonctionnel, l'invention a trait à un procédé de gestion de la réception d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé, comprenant la réception du paquet compressé suivie par la lecture dans le paquet compressé d'un identifiant dudit ensemble de règles de compression, dit identifiant de contexte unique, inséré au préalable dans le paquet compressé.

[0040] Grâce à cet aspect de l'invention, l'identifiant de contexte unique inséré dans le paquet compressé est reçu avec le paquet compressé puis lu. Cette information permet, dans un environnement multi-contexte et multi-saut, de retrouver l'ensemble de règles de compression qui a été utilisé pour compresser un paquet donné et de restaurer ainsi le paquet compressé dans son état original.

[0041] Selon un mode de mise en œuvre particulier relativement au deuxième aspect fonctionnel, le paquet compressé comprend des bits de remplissage, l'identifiant de contexte unique ayant été inséré au préalable dans les bits de remplissage du paquet compressé, et l'identifiant de contexte unique est lu en faisant une hypothèse portant sur le nombre de bits de remplissage du paquet compressé.

[0042] Grâce à ce mode de réalisation, l'identifiant de contexte unique inséré dans les bits de remplissage du paquet compressé peut être récupéré. Comme le nombre de bits de remplissage est variable pour chaque paquet, une manière de procéder est de réaliser des hypothèses sur le nombre de bits de remplissage présents dans le paquet

compressé qui a été reçu. Selon une telle hypothèse, il est possible de lire (ou pas) un identifiant de contexte inséré dans les bits de remplissage. De plus, pour une hypothèse donnée sur le nombre de bits de remplissage, il est possible que plusieurs identifiants de contexte soient possibles. Plusieurs hypothèses peuvent donc être faites relatives à l'identifiant de contexte pour une hypothèse donnée de longueur des bits de remplissage. Par exemple, pour une hypothèse de longueur nulle des bits de remplissage, l'identifiant de contexte n'a pas pu être inséré dans les bits de remplissage inexistants et tous les identifiants de contexte sont encore possibles. Différentes hypothèses sur la valeur possible de l'identifiant de l'ensemble de règles de compression utilisé pour compresser le paquet peuvent donc exister pour une longueur de bits de remplissage donnée.

[0043] Selon un mode de mise en œuvre particulier relativement au deuxième aspect fonctionnel, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le procédé est itératif, une hypothèse différente sur le nombre de bits de remplissage étant faite à chaque itération, et le procédé s'arrête dès que l'identifiant de contexte unique lu selon l'hypothèse permet de décompresser le paquet compressé.

[0044] Grâce à ce mode de réalisation, une manière possible de procéder à la lecture de l'identifiant de contexte unique est donnée. Une hypothèse sur la longueur des bits de remplissage permet de lire ou non un ou plusieurs identifiants de contexte unique pour cette longueur donnée. Cet identifiant de contexte lu peut alors être testé. Le test consiste à essayer cet identifiant pour essayer de décompresser le paquet compressé. En effet, l'identifiant de contexte lu permet d'identifier un ensemble de règles de compressions. Le paquet compressé porte également, en général, un identifiant de la règle utilisée pour compresser le paquet. Si l'ensemble de règles de compression correspondant à l'identifiant de contexte lu comprend bien une règle de compression correspondant à l'identifiant de règle lu sur le paquet compressé, et que cette règle de compression permet de décompresser de façon cohérente le paquet compressé, alors le test est réussi et le procédé peut s'arrêter. Une règle de décompression peut dans certains cas être déduite sans lecture d'un identifiant de règle, ce qui ne change pas le principe de l'invention. Le paquet compressé a donc, grâce au procédé, été transmis dans un environnement multi-contexte et multi-saut, sans rajouter de données au paquet compressé puisque l'identifiant de contexte unique est inséré dans les bits de remplissage, donc sans utiliser de bits supplémentaires.

[0045] Par malchance, l'identifiant de contexte unique lu selon l'hypothèse n'est pas le bon, mais a quand même permis de procéder à une décompression du paquet compressé. Cette décompression est donc erronée, mais n'a pas été détectée. Ce cas de figure devrait cependant être très rare et peut être accepté en comparaison de l'avantage

apporté par l'invention qui permet de réaliser des compressions très efficaces dans un environnement multi-contexte et multi-saut.

[0046] Selon un mode de mise en œuvre particulier relativement au deuxième aspect fonctionnel, qui pourra être mis en œuvre cumulativement avec le mode précédent, le procédé itératif commence en partant du nombre maximal possible pour la longueur des bits de remplissage et décroît ce nombre par un à chaque itération.

[0047] Grâce à ce mode de réalisation, une manière efficace de procéder à la sélection des différentes hypothèses est donnée. En partant d'une hypothèse de longueur maximale des bits de remplissage, un maximum de contraintes est donné sur l'identifiant de contexte unique qui peut être lu dans les bits de remplissage. Il est possible qu'aucun identifiant ne puisse être lu, pour des raisons d'incohérence, ce qui indiquerait tout simplement que l'hypothèse sur la longueur des bits de remplissage dans le paquet compressé n'est pas la bonne. En diminuant l'hypothèse sur la longueur des bits de remplissage un par un, il sera possible à un moment donné de lire un identifiant de contexte unique cohérent avec l'hypothèse sur la longueur des bits de remplissage. Si cet identifiant de contexte lu permet de décompresser le paquet compressé, il est selon toute probabilité l'identifiant correct. Le cas rare où cet identifiant ne serait pas le bon, mais permet quand même de décompresser le paquet compressé est accepté en regard des avantages apportés par l'invention.

[0048] Selon un mode de mise en œuvre particulier relativement au deuxième aspect fonctionnel, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, les bits du paquet compressé autres que les bits de remplissage ayant été combinés au préalable avec une séquence de bits construite en répétant l'identifiant de contexte unique, la décompression du paquet compressé comprend une combinaison inverse avec l'identifiant de contexte unique lu permettant de retrouver le paquet compressé tel qu'il était avant la combinaison préalable.

[0049] Grâce à ce mode de réalisation, le traitement des hypothèses sur les identifiants de contexte unique possibles est amélioré. En effet, dans ce mode de réalisation, pour décider si un identifiant de contexte unique issu d'une hypothèse est le bon, une opération supplémentaire est réalisée avant le test de décompression du paquet compressé. Cette opération est la combinaison inverse de celle qui a été réalisée entre l'identifiant de contexte unique et les bits autres que les bits de remplissage. Cette combinaison inverse permet de reconstruire la version originale du paquet compressé avant qu'il ait été combiné avec l'identifiant de contexte unique. Si le paquet original ainsi obtenu ne peut pas être décompressé en utilisant l'identifiant de contexte unique obtenu par hypothèse, c'est que cette hypothèse n'est pas bonne et il faut en essayer une autre. Cette opération supplémentaire permet donc de rendre plus discriminante les hypothèses successives en réalisant un test supplémentaire. Ce mode de réalisation

diminue donc le risque qu'une hypothèse retenue soit finalement erronée, et pour cela utilise une information supplémentaire qui a été donnée par la combinaison de l'identifiant de contexte unique avec les bits autres que les bits de remplissage du paquet compressé.

[0050] Selon un mode de de mise en œuvre particulier relativement au deuxième aspect fonctionnel, qui pourra être mis en œuvre alternativement ou cumulativement avec les modes précédents, le premier paquet décompressé obtenu en réalisant un essai de décompression avec l'identifiant de contexte unique lu est considéré comme étant le paquet original pour les utilisations ultérieures.

[0051] Grâce à ce mode de réalisation, une opération supplémentaire de décompression est évitée. Comme le procédé de réception du paquet compressé comprend des essais successifs de décompression du paquet compressé, et s'arrête lorsque la décompression réussit, le procédé selon l'invention comprend également au moins une opération de décompression et permet d'obtenir un paquet décompressé qui, selon toute probabilité, correspond bien au paquet original. Le paquet original est donc obtenu en évitant une opération supplémentaire de décompression qui serait faite avec l'identifiant de contexte unique lu et peut être utilisé ultérieurement. Le mode de réalisation économise donc une étape de calcul supplémentaire, donc économise du temps, de l'énergie et des ressources de calcul et de mémoire.

[0052] Selon un premier aspect matériel, l'invention a trait à une entité de gestion de la transmission d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé, comprenant les modules suivants :

- Un module d'insertion dans un paquet compressé d'un identifiant de l'ensemble de règles de compression, dit identifiant de contexte unique ; et,
- Un module de transmission d'un paquet compressé ainsi modifié par le module d'insertion.

[0053] Selon un mode de réalisation de cet aspect matériel, l'invention a trait à un équipement de télécommunications comprenant une entité de gestion de la transmission d'un paquet de données compressé selon l'invention.

[0054] Selon un mode de réalisation de cet aspect matériel, l'invention a trait à un terminal mobile comprenant une entité de gestion de la transmission d'un paquet de données compressé selon l'invention.

[0055] Selon un deuxième aspect matériel, l'invention a trait à une entité de gestion de la réception d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé, comprenant les modules suivants :

- Un module de réception d'un paquet compressé ; et,

- Un module de lecture dans un paquet compressé d'un identifiant de l'ensemble de règles de compression, dit identifiant de contexte unique, inséré au préalable dans le paquet compressé.

[0056] Selon un mode de réalisation de ce deuxième aspect matériel, l'invention a trait à un équipement de télécommunications comprenant une entité de gestion de la réception d'un paquet de données compressé selon l'invention.

[0057] Selon un mode de réalisation de ce deuxième aspect matériel, l'invention a trait à un terminal mobile comprenant une entité de gestion de la réception d'un paquet de données compressé selon l'invention.

[0058] Selon un autre aspect matériel, l'invention a trait à un programme d'ordinateur apte à être mis en œuvre par une entité de gestion de la transmission d'un paquet de données compressé, le programme comprenant des instructions de code qui, lorsqu'il est exécuté par un processeur, réalise les étapes du procédé de gestion de la transmission défini ci-dessus.

[0059] Selon un autre aspect matériel, l'invention a trait à un programme d'ordinateur apte à être mis en œuvre par une entité de gestion de la réception d'un paquet de données compressé, le programme comprenant des instructions de code qui, lorsqu'il est exécuté par un processeur, réalise les étapes du procédé de gestion de la réception défini ci-dessus.

[0060] 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 de gestion de la réception et de la transmission définis ci-dessus.

[0061] 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 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 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. Le programme selon l'invention peut utiliser tout type de technologie informatique en terme de langages de programmation compilés, interprétés, ou une combinaison des deux ainsi qu'en terme de systèmes d'exploitation.

Brève description des figures

[0062] L'invention sera mieux comprise à la lecture de la description qui suit, donnée à titre d'exemple, et faite en référence aux dessins annexées sur lesquels :

[0063] [Fig.1] représente une entité de gestion de la transmission d'un paquet de données compressé et une entité de gestion de la réception d'un paquet de données compressé selon l'invention.

[0064] [Fig.2] illustre un exemple d'insertion d'un identifiant de contexte unique dans les bits d'un paquet compressé selon un mode de réalisation de l'invention.

[0065] [Fig.3] illustre un exemple de succession d'étapes permettant la lecture d'un identifiant de contexte unique selon un mode de réalisation de l'invention.

[0066] [Fig.4] illustre un exemple de succession d'étapes mettant en œuvre les procédés de gestion de la transmission et de la réception d'un paquet de données compressé selon l'invention.

Description détaillée

[0067] La [Fig.1] représente un système informatique ou de télécommunications incluant une entité de gestion 100 de la transmission d'un paquet de données compressé P et une entité de gestion 200 de la réception d'un paquet de données compressé P' selon l'invention.

[0068] L'entité de gestion 100 de la transmission d'un paquet de données compressé P comprend les modules suivants :

- Un module 101 d'insertion dans un paquet de données compressé P d'un identifiant de contexte unique ICU ;
- Un module 102 de transmission du paquet de données compressé P', obtenu par l'insertion de l'identifiant ICU dans le paquet compressé P.

[0069] L'entité de gestion 200 de la réception d'un paquet de données compressé P' comprend les modules suivants :

- Un module 201 de réception du paquet de données compressé P' ;
- Un module 202 de lecture dans le paquet de données compressé P' de l'identifiant ICU qui avait été inséré au préalable dans le paquet compressé P. Cette insertion modifie le paquet de données compressé P en le paquet compressé P'.

[0070] Les deux entités de gestion 100 et 200 appartiennent à un réseau de communication NET. Le réseau de communication NET permet aux entités de gestion 100 et 200 de communiquer entre elles, et potentiellement avec d'autres entités, selon des protocoles de communication fondés sur l'émission et la réception de paquets de données. Le réseau NET peut par exemple être un réseau IP (acronyme pour l'anglais *Internet Protocol*). Le réseau NET peut aussi être un réseau Ethernet. D'autres protocoles de communication peuvent être déployés dans le réseau NET. Les technologies physiques

permettant la communication au sein du réseau NET peuvent être de tout type. Il peut s'agir d'un réseau filaire, ou non. Le réseau NET peut être par exemple le réseau Internet ou tout sous-réseau d'Internet. Le réseau NET peut également être un réseau privé, par exemple un réseau d'entreprise. Le réseau NET peut également être un réseau de télécommunications mobiles selon les technologies 3G et suivantes jusqu'à 6G. Le réseau NET peut être un réseau local ou à plus grande échelle. Le réseau NET peut par exemple utiliser une technologie WiFi, ou Bluetooth. Le réseau NET peut être un réseau de l'Internet des objets (traduction de l'anglais *Internet of Things*) et utiliser des protocoles adaptés à ce mode de communication tels que LoraWAN, Sigfox ou Zigbee ou autres technologies similaires.

[0071] Les deux entités de gestion 100 et 200 présentent l'architecture matérielle d'un ordinateur conventionnel. Elles comportent notamment un processeur, une mémoire vive de type RAM et une mémoire morte telle qu'une mémoire de type Flash, ROM, (non représentés sur la figure) ou peuvent disposer de telles ressources. Les entités de gestion 100 et 200 comportent également des dispositifs d'entrée-sortie, en particulier une ou des cartes réseau (non représentés sur la figure) qui leur permettent de communiquer via le réseau de communication NET. Elles peuvent également comporter, dans certains exemple de réalisation, des dispositifs tels que claviers et/ou écrans (non représentés sur la figure).

[0072] Les deux entités de gestion 100 et 200 peuvent appartenir à des équipements informatiques ou de télécommunications. Par exemple, elles peuvent être incluses dans des ordinateurs fixes ou mobiles, ou des serveurs informatiques. Elles peuvent également être incluses dans des routeurs, ou des passerelles entre réseaux, ou des terminaux, mobiles ou non. Elles peuvent également être incluses dans des objets connectés de tout type, tels que montres connectées, caméras connectées, véhicules connectés, compteurs d'eau ou d'énergie, ou autres.

[0073] Les deux entités de gestion 100 et 200 peuvent également être déployées dans une architecture en nuage (traduction de l'anglais *cloud architecture*). Les entités de gestion 100 et 200 peuvent s'exécuter sur des ressources distantes en calcul et mémoire pour exécuter les procédés objets de l'invention.

[0074] Les deux entités de gestion 100 et 200 peuvent se transmettre des paquets de données en utilisant les protocoles déployés dans le réseau de communication NET. En particulier, les communications au sein du réseau NET mettent en œuvre au moins un protocole de compression des paquets de données. Ce protocole peut être le protocole ROHC, ou le protocole SCHC, ou bien une évolution d'un de ces protocoles, ou bien un protocole qui utilise des principes ou techniques similaires à ceux mis en œuvre par ces protocoles.

[0075] En particulier, dans le réseau NET, les paquets de données échangés P, P' peuvent être compressés et décompressés par la mise en œuvre d'un protocole de compression. Un paquet de données P, P' est compressé par une entité en lui appliquant une règle de compression suivant le protocole. Le paquet de données P, P' compressé peut alors être transmis dans le réseau NET ; la compression du paquet fait économiser des ressources réseau, mémoire, de calcul de façon très significative. Lorsqu'il est reçu, le paquet compressé P, P' peut être décompressé par une autre entité en lui appliquant une règle de décompression suivant le protocole. La règle de décompression est, en général, l'application des opérations inverses des opérations qui ont été appliquées lors de la compression du paquet de données P, P'. La connaissance de la règle de compression qui a été appliquée pour compresser le paquet de données P, P' est donc, en général, nécessaire pour décompresser le paquet P, P'. On peut imaginer, dans certains modes de réalisation, un fonctionnement qui procède par essais-erreur, donc sans connaissance de la règle de compression utilisée. Mais le fonctionnement standard de la décompression repose sur cette connaissance partagée, entre l'entité qui réalise la compression d'une part et celle qui réalise la décompression d'autre part, de la règle de compression appliquée.

[0076] L'objectif de l'invention est de pouvoir mettre en œuvre un protocole de compression des paquets de données dans un environnement où il y a plusieurs contextes possibles de compression. Un contexte de compression peut être par exemple un sous-réseau particulier, pour laquelle certaines règles de compression sont plus adaptées que d'autres. La présence de différents contextes possibles de compression se traduit par la coexistence de plusieurs ensembles de règles de compression. La compression d'un paquet de données et la transmission d'un paquet compressé doit donc être adapté pour prendre en compte la coexistence de plusieurs contextes et de plusieurs ensembles de règles de compression associés aux différents contextes.

[0077] En particulier, des règles présentes dans des ensembles différents de règles de compression peuvent avoir le même identifiant de règle, qui sert à différencier une règle au sein d'un ensemble donné de règles de compression. Autrement dit, la connaissance seule de l'identifiant de règle de compression n'est pas suffisante pour retrouver la règle de compression utilisée pour compresser un paquet. En effet, un identifiant de règle donné peut correspondre à différentes règles, présentes dans différents ensembles de règles, qui se trouvent partager le même identifiant de règle mais être bien distinctes.

[0078] L'entité de gestion 100 réalise un procédé de transmission d'un paquet de données P compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression. Cet ensemble de règles de compression est identifié par un identifiant ICU, dit identifiant de contexte unique.

[0079] Dans des exemples de réalisation, le procédé selon l'invention comprend l'obtention préalable de l'identifiant ICU. Par exemple, des équipements de réseau ou des terminaux mobiles ou des équipements informatiques peuvent se voir attribuer différents ensembles de règles de compression correspondant aux différents contextes de compression que l'équipement ou le terminal doit gérer suivant un protocole de compression donné. Ces différents ensembles de règles de compression peuvent être obtenus par apprentissage automatique, en observant du trafic réel, puis distribués aux différents équipements et terminaux qui vont les appliquer. L'apprentissage automatique d'ensembles de règles de compression peut être centralisé ou distribué sur les différents équipements ou terminaux qui utiliseront les ensembles de règles de compression appris. Les ensembles de règles de compression correspondant à différents contextes peuvent aussi être définis par des experts et distribués dans les différents équipements ou terminaux qui vont les utiliser.

[0080] Dans des exemples de réalisation, un procédé d'obtention d'un identifiant pour les ensembles de règles de compression est mis en œuvre. Par exemple, le procédé peut comprendre l'extraction de valeurs cibles présentes dans les champs des règles de l'ensemble. Les valeurs cibles peuvent être extraites pour certains types de règles seulement. A partir de valeurs cibles, le procédé peut comprendre l'application de transformations qui vont produire l'identifiant de l'ensemble de règles de compression à partir des valeurs cibles extraites. Ces transformations peuvent comprendre l'application de fonctions telles que des fonctions de hachage ou d'autres fonctions d'encodage, ainsi que des fonctions permettant d'assurer une taille donnée à l'identifiant, telles que des opérations de troncature ou de poinçonnage.

[0081] L'identifiant d'ensemble de règles de compression ainsi obtenu peut être exprimé comme une suite de bits. Il peut être dénommé identifiant de contexte unique ICU car il permet d'identifier un contexte réseau pour lequel un ensemble de règles de compression donné est pertinent. Les équipements ou terminaux qui mettent en œuvre un procédé de compression et/ou de décompression peuvent utiliser un identifiant de contexte unique ICU pour mieux préciser dans quel contexte réseau ils vont appliquer des règles de compression et/ou de décompression. L'identifiant de contexte unique ICU permet donc à des équipements ou terminaux d'utiliser un protocole de compression dans un environnement multi-contexte ou multi-saut.

[0082] L'identifiant ICU ainsi obtenu dans certains exemples de réalisation peut être utilisé par l'entité de gestion 100.

[0083] Le module 101 de l'entité de gestion 100 réalise l'insertion de l'identifiant ICU dans un paquet de données compressé P, ayant été compressé par l'application d'une règle appartenant à un ensemble de règles de compression identifié par l'identifiant ICU. Le paquet P modifié par l'insertion de l'identifiant ICU est dénommé P' sur la [Fig.1].

Pour faciliter l'insertion de l'identifiant ICU dans le paquet compressé P, l'identifiant ICU peut être exprimé comme une suite de bits.

[0084] Selon un mode de réalisation, l'identifiant ICU est inséré dans les bits de remplissage du paquet compressé P. De cette manière, les données présentes dans le paquet compressé P ne sont pas effacées par l'insertion de l'identifiant ICU. Le fait d'utiliser les bits de remplissage du paquet compressé P pour insérer l'identifiant ICU contribue à l'objectif d'économie de ressources en bande passante du réseau NET, puisqu'aucune donnée supplémentaire n'est utilisée pour transmettre l'information supplémentaire de l'identifiant ICU.

[0085] Selon un mode de réalisation, l'identifiant ICU est inséré dans les bits de remplissage du paquet compressé P en partant du dernier bit du paquet P selon une permutation définie. Comme le nombre de bits de remplissage est variable pour chaque paquet, il est pratique de partir du dernier bit du paquet compressé pour réaliser l'insertion car les bits de remplissage se trouvent à la fin du paquet compressé. Pour être sûr que tous les bits du paquet compressé soient insérés, une permutation peut être définie afin d'affecter à chaque bit de remplissage un et un seul bit formant la suite de bits qui représente l'identifiant ICU. Cette permutation n'a pas à être continue.

[0086] Dans un exemple, l'insertion de l'identifiant de contexte unique ICU dans les bits de remplissage du paquet compressé P se fait en insérant l'identifiant de contexte unique ICU dans les bits de remplissage en sens contraire au sens de lecture du paquet compressé P. Ici, le premier bit de l'identifiant ICU est inséré dans le dernier bit du paquet compressé P, si celui-ci est un bit de remplissage, puis l'insertion de l'identifiant ICU se fait en sens inverse de la lecture du paquet compressé P.

[0087] Dans un autre exemple, l'insertion de l'identifiant de contexte unique ICU dans les bits de remplissage du paquet compressé P se fait en insérant l'identifiant de contexte unique ICU dans les bits de remplissage en partant de la fin de l'identifiant ICU, dans le sens de lecture du paquet compressé P. Ici, le dernier bit de l'identifiant ICU est inséré dans le dernier bit du paquet compressé P, si celui-ci est un bit de remplissage, puis l'insertion de l'identifiant ICU se fait en respectant le sens de lecture du paquet compressé P. C'est-à-dire que le dernier bit de l'identifiant ICU est inséré sur le dernier bit su paquet P, l'avant-dernier de l'identifiant ICU sur l'avant-dernier bit du paquet compressé P et ainsi de suite.

[0088] Il n'y a pas de garantie sur le nombre de bits de remplissage présents dans le paquet compressé P. Dans un exemple de réalisation, l'insertion de l'identifiant ICU dans les bits de remplissage du paquet compressé P s'arrête quand il n'y a plus de bits de remplissage disponibles pour insérer l'identifiant ICU. Autrement dit, dans ce cas, l'identifiant ICU est tronqué quand il n'y a pas assez de bits de remplissage pour

effectuer une insertion complète. Il peut y avoir zéro bit de remplissage dans le paquet compressé P et dans ce cas, aucun bit de l'identifiant ICU ne sera inséré.

[0089] En revanche, si le nombre de bits de remplissage présents dépasse la longueur de l'identifiant ICU, un exemple de réalisation possible est de poursuivre l'insertion de l'identifiant ICU. De cette manière, l'ensemble des bits de remplissage accessibles est utilisé. Cette redondance rend plus robuste la transmission de l'identifiant ICU.

[0090] Selon un exemple de réalisation, l'insertion de l'identifiant de contexte unique ICU comprend de surcroît une opération de combinaison des bits du paquet compressé P autres que les bits de remplissage avec une séquence de bits construite en répétant l'identifiant de contexte unique ICU. Dans cet exemple, une opération supplémentaire permet d'améliorer la transmission de l'identifiant ICU en dehors des bits de remplissage. Pour cela, l'identifiant ICU ne peut pas simplement être inséré dans les bits du paquet compressé P en dehors des bits de remplissage. En effet, une insertion directe effacerait l'information transmise dans le paquet compressé P. La solution donnée dans cet exemple de réalisation est de combiner l'identifiant ICU avec les bits autres que ceux de remplissage afin de préserver les deux informations. Une combinaison possible est d'utiliser une opération de ou exclusif qui permettra, comme on le verra, de retrouver aisément les bits du paquet compressé avant modification. La combinaison se fait avec une séquence construite en répétant l'identifiant de contexte unique ICU. Dans la quasi-totalité des cas, les bits autres que de remplissage du paquet compressé seront en plus grand nombre que ceux qui forment l'identifiant ICU, ce qui donne la possibilité de répéter celui-ci pour arriver au bout des bits autres que ceux de remplissage du paquet compressé P.

[0091] L'opération de combinaison avec les bits qui ne sont pas des bits de remplissage peut se faire de façon séquentielle avec l'opération d'insertion dans les bits de remplissage, et ce dans n'importe quel ordre, ou bien en parallèle. En effet, ces deux opérations manipulent des parties disjointes du paquet compressé P, à savoir les bits de remplissage d'une part, et les autres bits d'autre part.

[0092] Le résultat de l'insertion de l'identifiant ICU dans le paquet compressé P est un paquet P', modifié par rapport au paquet P, mais qui est également un paquet compressé car de même taille que le paquet P qui était lui-même compressé par l'application d'une règle de compression donnée. Le paquet compressé P' est produit par le module 101 de l'entité de gestion 100. Il est fourni au module 102 de l'entité de gestion 100.

[0093] Le module 102 de l'entité de gestion réalise la transmission du paquet compressé P' ainsi modifié par le module d'insertion 101.

[0094] Cette transmission du paquet compressé P' se fait en utilisant le réseau de communication NET selon des protocoles d'échanges de paquets de données mis en

œuvre dans le réseau NET. La transmission d'un paquet compressé P' économise des ressources en mémoire, énergétiques, en capacités de calcul et en bande passante par rapport au paquet de données original qui avait déjà été compressé pour donner le paquet P. Par rapport au paquet P, la transmission du paquet P' comporte une information supplémentaire relative à l'identifiant ICU qui a été inséré dans le paquet compressé P pour donner le paquet compressé P'.

[0095] Le module 201 de l'entité de gestion 200 réalise la réception du paquet compressé P'. Cette réception est la mise en œuvre des protocoles de communication déployés dans le réseau de communication NET qui ont été utilisés pour transmettre le paquet compressé P'.

[0096] Le module 202 de l'entité de gestion 2000 réalise la lecture dans le paquet compressé P' de l'identifiant ICU de l'ensemble de règles de compression dans lequel se trouve la règle qui a été appliquée pour réaliser la compression du paquet P. L'identifiant ICU a été inséré au préalable dans le paquet compressé P, ce qui a donné le paquet compressé P' qui a été transmis puis reçu ainsi que présenté ci-dessus.

[0097] Selon un exemple de réalisation, le paquet compressé P' comprend des bits de remplissage et l'identifiant ICU a été inséré au préalable dans les bits de remplissage du paquet P'. L'identifiant ICU est alors lu en faisant une hypothèse à propos du nombre de bits de remplissage présents dans le paquet P'. En effet, c'est une telle hypothèse qui permet de savoir quels bits du paquet compressé P' sont susceptibles de porter l'identifiant de contexte unique ICU.

[0098] Selon un exemple de réalisation, la lecture de l'identifiant ICU est faite grâce à un procédé itératif, une hypothèse différente sur le nombre de bits de remplissage étant faite à chaque itération, et le procédé s'arrête dès que l'identifiant de contexte unique lu selon l'hypothèse permet de décompresser le paquet compressé.

[0099] La lecture de l'identifiant ICU ne pourra pas en général être faite après une seule hypothèse portant sur le nombre de bits de remplissage dans le paquet compressé P'. Le procédé de lecture est donc itératif et va procéder en changeant d'hypothèse quant au nombre de bits de remplissage à chaque itération. Le procédé s'arrête quand l'hypothèse permet effectivement de lire un identifiant de contexte unique ICU qui permet ensuite de décompresser le paquet compressé P'. En effet, l'identifiant de contexte ICU permet de trouver un ensemble de règles de compression qui a pu être utilisé pour compresser le paquet P (et donc P', qui diffère de P par l'insertion de l'identifiant ICU). Un identifiant de règle peut alors être cherché dans le paquet compressé P', et si la règle correspondant à cet identifiant de règle dans l'ensemble de règles identifié par ICU permet de décompresser le paquet compressé P', alors il n'est pas nécessaire de chercher plus loin. Par malchance, l'identifiant ICU lu peut ne pas être le bon et mener à une décompression erronée du paquet compressé P'. Ce risque

est faible et peut être diminué en inscrivant suffisamment d'informations relatives à l'identifiant ICU dans le paquet compressé P'. Cet inconvénient est largement contrebalancé par la possibilité de mettre en œuvre des compressions très efficaces dans un contexte multi-saut.

[0100] Dans des modes de réalisation, le paquet obtenu quand la décompression du paquet compressé P' est réussie sera considéré comme le paquet original, qui avait été compressé au départ pour donner le paquet P. De cette manière, le procédé réalise la décompression complète du paquet compressé P et ne se contente pas de fournir l'identifiant de contexte unique ICU.

[0101] Selon un exemple de réalisation, le procédé itératif commence en partant du nombre maximal possible pour la longueur des bits de remplissage et décroît ce nombre par un à chaque itération. De cette manière, les hypothèses faites sur la longueur des bits de remplissage commencent par l'hypothèse la plus discriminante possible, celle qui comprend le plus de bits de remplissage possible. Le procédé procède en diminuant cette hypothèse de un à chaque itération afin d'examiner successivement toutes les hypothèses possibles sur la longueur des bits de remplissage du paquet P', et en prenant bien garde à examiner les hypothèses les plus discriminantes d'abord.

[0102] Dans un autre exemple, les bits du paquet compressé P autres que les bits de remplissage ont été combinés au préalable avec une séquence de bits construite en répétant l'identifiant de contexte unique ICU pour donner le paquet P'. Dans cet exemple, la décompression du paquet compressé P' comprend une combinaison inverse avec l'identifiant de contexte unique ICU lu permettant de retrouver le paquet compressé tel qu'il était avant la combinaison préalable.

[0103] Il est possible de fournir des informations sur l'identifiant ICU dans les bits de P' qui ne sont pas des bits de remplissage. Pour cela, une combinaison de l'identifiant ICU avec les bits qui ne sont pas des bits de remplissage peut être effectuée. Lorsqu'un identifiant ICU est lu dans les bits de remplissage du paquet P' selon une hypothèse, la combinaison inverse peut être appliquée aux bits de P' qui ne sont pas des bits de remplissage. C'est le résultat de cette combinaison inverse que le procédé va alors chercher à décompresser, puisque le résultat de cette combinaison inverse est le paquet compressé P tel qu'il était avant la combinaison initiale. Si la décompression réussit, c'est une indication supplémentaire que l'hypothèse faite est la bonne. La combinaison inverse permet donc de retrouver l'information supplémentaire encodée dans les bits du paquet P' qui ne sont pas les bits de remplissage.

[0104] La combinaison inverse, dans des exemples de réalisation, est une opération de ou exclusif. Quand la combinaison initiale a été faite avec un ou exclusif, la combinaison inverse utilise la même opération de ou exclusif grâce aux propriétés de cette

opération. Ceci rend l'application d'une combinaison d'une part et de la combinaison inverse d'autre part très facile et économe en ressources de calcul.

[0105] La [Fig.2], quant à elle, présente un exemple d'insertion d'un identifiant ICU de contexte unique dans un paquet compressé P selon un mode de réalisation de l'invention.

[0106] Dans l'exemple de la [Fig.2], le paquet compressé P est formé de trois octets. Les octets formant P sont représentés sur la figure comme séparés par des barres verticales. Le paquet compressé P a été compressé par l'application d'une règle R. Comme souvent, l'identifiant de la règle R a été ajouté au paquet compressé P dans son en-tête afin de faciliter la décompression ultérieure. L'identifiant de la règle R est donné par les quatre premiers octets du paquet compressé P, à savoir 0001. Cet identifiant est une partie intégrale du paquet compressé P.

[0107] Le paquet compressé P, exprimé en tant que suite de bits, ne tombait pas sur un nombre exact d'octets. Le paquet P a donc été complété avec quatre bits de remplissage, indiqué sur la figure par des lettres XXXX. Comme il est usuel, les bits de remplissage XXXX ont été ajoutés à la fin du paquet compressé P afin qu'il soit formé d'un nombre exact d'octets, pour simplifier sa transmission.

[0108] La règle de compression R a pour identifiant 0001. Cet identifiant est exprimé sur 4 bits. Il permet de retrouver la règle R dans un ensemble de règles de compression. Cet ensemble est lui-même identifié par un identifiant ICU qui vaut 110, qui est donc exprimé sur trois bits. Il est possible d'identifier huit ensembles de règles de compression avec trois bits, et seize règles de compression avec quatre bits. Autrement dit, dans l'exemple donné sur la [Fig.2], les équipements réseaux peuvent manipuler jusqu'à 8 ensembles de règles de compression distincts, donc 8 contextes réseau différents. Et chaque contexte ou ensemble de règles de compression peuvent avoir jusqu'à 16 règles différentes. D'autres exemples numériques sont bien sûr possibles selon les longueurs des identifiants de règles et d'ensembles de compression.

[0109] Il est bien clair que différentes ensembles de règles de compression peuvent comprendre des règles différentes mais qui ont le même identifiant 0001 que la règle R. En soi, la connaissance de l'identifiant 0001 inscrit dans le paquet compressé P ne permet pas de retrouver la règle R qui a servi à compresser le paquet P. L'identifiant de contexte unique ICU qui vaut 110 doit aussi être communiqué.

[0110] Pour cela, la mise en œuvre du procédé de transmission du paquet compressé P comprend une étape d'insertion de l'identifiant ICU dans le paquet P qui va être ainsi modifié en un paquet P'. Le paquet P' sera ensuite transmis, et un procédé de réception permettra de retrouver le paquet compressé P initial.

[0111] L'insertion de l'identifiant ICU dans le paquet compressé P comprend dans l'exemple de la [Fig.2] deux opérations distinctes. Ces deux opérations peuvent prendre place dans n'importe quel ordre ou être réalisées en parallèle.

[0112] En ce qui concerne la première opération, l'identifiant ICU est inséré dans les bits de remplissage XXXX du paquet P. Pour cela, un choix de permutation est fait ; dans notre exemple, le choix est d'écrire l'identifiant à partir du dernier bit du paquet compressé P, dans le sens inverse de lecture. Ceci est représenté par une flèche de droite à gauche dans la [Fig.2]. Les trois derniers bits du paquet compressé P' modifié sont donc les bits 011 qui correspondent à une lecture à l'envers de l'identifiant ICU 110.

[0113] Comme il y a quatre bits de remplissage et que l'identifiant ICU est formé de trois bits, le choix est fait de répéter l'identifiant ICU et, dans cet exemple, d'inscrire le premier bit de l'identifiant ICU, soit 1, sur le dernier bit de remplissage du paquet compressé P en partant de la fin. Les quatre derniers bits du paquet compressé modifié P', qui correspondent aux quatre bits de remplissage XXXX du paquet compressé P sont donc 1011.

[0114] S'il y avait eu moins de bits de remplissage que nécessaire pour inscrire l'identifiant ICU, celui-ci aurait pu être tronqué dans notre exemple de réalisation.

[0115] En ce qui concerne la deuxième opération, il s'agit de réaliser une combinaison des bits autres que ceux de remplissage du paquet compressé P avec une séquence construire en répétant l'identifiant ICU. Comme il y a 4 bits de remplissage et 20 bits autres dans le paquet compressé P, la séquence obtenue en répétant l'identifiant ICU est la suivante : | 11011011 | 01101101 | 1011. Nous avons utilisé des barres verticales pour indiquer les frontières des octets. Cette séquence est dénotée Rep(ICU) dans la [Fig.2].

[0116] L'opération de combinaison choisie est une opération de ou exclusif réalisée bit à bit entre les bits du paquet P autres que ceux de remplissage et la séquence construite en répétant l'identifiant ICU. Cette opération ou exclusif est dénotée XOR comme il est usuel dans la [Fig.2].

[0117] On obtient, après ces deux opérations, qui peuvent être réalisées dans n'importe quel ordre ou en parallèle, un paquet compressé P' modifié par rapport au paquet compressé P en insérant l'identifiant ICU dans les bits de remplissage XXXX du paquet P et en combinant une séquence construite en répétant l'identifiant ICU avec les bits du paquet P autres que ceux de remplissage. Finalement, en utilisant des barres verticales pour marquer les frontières d'octets, le paquet compressé P' obtenu en modifiant ainsi le paquet compressé P vaut : | 11001100 | 11100111 | 00101011 |.

[0118] La [Fig.3], quant à elle, présente un exemple de lecture d'un identifiant ICU de contexte unique dans un paquet compressé P' selon un mode de réalisation de l'invention.

[0119] Le paquet compressé P' reçu qui est montré dans la [Fig.3] est le même que celui qui a été obtenu dans l'exemple de la [Fig.2], soit : | 11001100 | 11100111 | 00101011 |.

[0120] Le procédé selon l'invention va chercher à lire l'identifiant ICU en procédant à des hypothèses sur la longueur initiale des bits de remplissage du paquet compressé P qui a été modifié pour donner le paquet P'. Dans l'exemple de la [Fig.2], la permutation choisie pour insérer l'identifiant ICU consistait à l'inscrire en partant du dernier bit du paquet compressé, en sens inverse de lecture, en le répétant s'il y avait trop de bits de remplissage comparé à la longueur 3 de l'identifiant ICU et en le tronquant si nécessaire.

[0121] Donc, la lecture montrée dans la [Fig.3] commence en prenant les huit derniers bits du paquet compressé P' et en les lisant à l'envers, ce qui donne la séquence 11010100. On prend dans cet exemple les huit derniers bits car il s'agit du nombre maximal de bits de remplissage qui pouvaient être présents dans le paquet compressé P, soit un octet entier.

[0122] La première hypothèse faite Hyp1 suppose que la longueur L des bits de remplissage était de 8, soit la longueur maximale. Dans ce cas, la longueur de l'identifiant ICU étant de trois bits, l'identifiant ICU devrait valoir 110. Si la longueur L des bits de remplissage était bien de huit, l'identifiant ICU aurait été répété jusqu'à couvrir les huit bits de remplissage. Or, la séquence 11010100 qui est lue dans les huit derniers bits du paquet compressé P' ne présente pas la répétition de l'identifiant ICU supposé valoir 110. La première hypothèse Hyp1 est donc impossible.

[0123] Il en est de même pour les hypothèses suivantes Hyp2, Hyp3, Hyp4 qui sont faites en supposant une longueur L des bits de remplissage décroissante à chaque fois de un, soit successivement L=7, L=6, L=5.

[0124] D'autres successions d'hypothèses sur la longueur L des bits de remplissage peuvent être choisies. Par exemple, il est possible de faire des sauts afin d'accélérer le procédé. Ou bien de partir d'une longueur plus faible afin de faire plusieurs hypothèses sur l'identifiant de contexte unique ICU.

[0125] L'hypothèse Hyp5 de longueur L=4 des bits de remplissage est cohérente. Elle donne un identifiant ICU qui vaut 110 répété uniquement sur un bit pour compléter les quatre bits de remplissage.

[0126] En utilisant cette hypothèse Hyp5, il est possible de séparer dans le paquet compressé P' les bits de remplissage (les quatre derniers, indiqués XXXX dans la [Fig.3]) des autres bits (les vingt premiers). Il est possible également de construire la séquence Rep(ICU) en répétant l'identifiant ICU jusqu'à obtenir vingt bits. La combinaison

inverse peut être alors appliquée à P' en appliquant l'opération de ou exclusif XOR. L'avantage d'utiliser l'opération ou exclusif XOR comme combinaison est qu'elle est utilisée à la fois en combinaison et combinaison inverse, ce qui simplifie les procédés.

[0127] On obtient ainsi une hypothèse de paquet compressé P qui a été modifié selon le procédé pour donner le paquet compressé P' transmis puis reçu à la place de P. L'identifiant ICU est connu dans l'hypothèse et vaut 110, et la règle R est identifiée dans l'hypothèse par les quatre premiers bits du paquet P, soit 0001. Ces deux identifiants conjoints 110 et 0001 permettent de retrouver la règle R qui a été utilisée pour donner le paquet compressé P.

[0128] Une opération non montrée dans la [Fig.3] est le test de cohérence final. La règle R identifiée permet-elle de décompresser correctement le paquet P ? Si c'est bien le cas, le procédé s'arrêtera. Le risque d'erreur est jugé suffisamment faible pour ne pas annuler l'avantage d'une compression très efficace. En particulier, la combinaison inverse réalisée avec la séquence Rep(ICU) donne une information indépendante de l'hypothèse sur la longueur L des bits de remplissage, donc une forte cohérence du résultat final.

[0129] En l'occurrence, dans l'exemple de la [Fig.3], le résultat obtenu par la lecture de l'identifiant ICU donne bien le paquet compressé P initial. La décompression du paquet P avec la règle R obtenue dans l'ensemble identifié par l'identifiant ICU qui vaut 110 en utilisant l'identifiant de règle 0001 sera donc réussie.

[0130] Le procédé de réception du paquet compressé permet donc aussi d'obtenir, par ce test final, le paquet décompressé pour tout usage ultérieur.

[0131] La [Fig.4], quant à elle, présente un exemple de succession d'étapes mettant en œuvre les procédés de gestion de la transmission et de la réception d'un paquet de données compressé selon l'invention.

[0132] L'entité de gestion 100 a la charge de la transmission du paquet compressé P dans un environnement multi-contextes. L'ensemble de règles de compression d'où a été extrait la règle qui a permis de compresser le paquet P est identifié par l'identifiant ICU.

[0133] Pour réaliser le procédé de transmission selon l'invention, le module 101 de l'entité de gestion 100 va d'abord insérer l'identifiant ICU dans le paquet P. Pour cela, dans l'exemple présenté ici, le module 101 réalise les deux étapes S1 et S2.

[0134] Dans l'étape S1, l'identifiant ICU est inséré dans les bits de remplissage du paquet compressé P.

[0135] Dans l'étape S2, une séquence construite en répétant l'identifiant ICU est combinée bit à bit avec les bits du paquet compressé P qui ne sont pas des bits de remplissage.

[0136] Ces deux étapes sont présentées, dans l'exemple de la [Fig.4], comme prenant place l'une après l'autre, mais l'étape S2 peut prendre place avant l'étape S1 ou bien être

réalisée en parallèle avec l'étape S1. En effet, ces deux étapes manipulent des parties disjointes du paquet compressé P et sont donc indépendantes l'une de l'autre.

[0137] Le paquet compressé P ainsi modifié est dénommé, dans l'ensemble des figures, comme étant le paquet compressé P'. Il est fourni après l'exécution des étapes S1 et S2 (dans n'importe quel ordre ou en parallèle) au module 102 de l'entité de gestion 100.

[0138] Le module 102 de l'entité de gestion 100 réalise la transmission proprement dite du paquet compressé P', qui correspond au paquet compressé P modifié par l'insertion de l'identifiant de contexte unique ICU. Cette transmission est dénommée sur la [Fig.4] comme étant l'étape S3.

[0139] Cette transmission se fait à travers le réseau de communication NET, en utilisant les protocoles de communication offerts par ce réseau. Le paquet compressé P' est reçu par l'entité de gestion 200 qui met en œuvre un procédé de réception selon l'invention.

[0140] Dans l'exemple présenté dans la [Fig.4], le module 201 de l'entité de gestion 200 procède à la réception du paquet compressé P', en réalisant l'étape S4. Il le fournit au module 202 de l'entité de gestion 200, qui va procéder à la lecture du paquet compressé P et de l'identifiant de contexte unique ICU.

[0141] Pour cela, le module 202 réalise dans l'exemple présenté ici, lors de l'étape S5, une hypothèse sur le nombre de bits de remplissage et va lire l'identifiant ICU et le paquet compressé P selon cette hypothèse.

[0142] Si l'hypothèse permet de réaliser la décompression DEC du paquet compressé P, le procédé s'arrête. Il a permis de recevoir et décompresser le paquet compressé P et d'obtenir l'identifiant de contexte unique ICU qui identifie l'ensemble de règles de compression utilisé pour obtenir le paquet compressé P. Le paquet initial, résultat de la décompression du paquet P, est également obtenu par effet de bord.

[0143] Dans des exemples de réalisation plus simples des procédés, seules les étapes d'insertion de l'identifiant ICU dans le paquet compressé P, de transmission puis de réception de celui-ci, et de lecture dans le paquet reçu de l'identifiant ICU sont réalisées.

[0144] 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

[Revendication 1] Procédé de gestion de la transmission d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé (P), comprenant l'insertion dans ledit paquet compressé (P) d'un identifiant dudit ensemble de règles de compression, dit identifiant de contexte unique (ICU), suivie par la transmission du paquet compressé ainsi modifié (P').

[Revendication 2] Procédé de gestion de la transmission d'un paquet compressé selon la revendication 1 caractérisé en ce que le paquet compressé (P) comprend des bits de remplissage et en ce que l'identifiant de contexte unique (ICU) est inséré dans les bits de remplissage du paquet compressé (P).

[Revendication 3] Procédé de gestion de la transmission d'un paquet compressé selon la revendication 2 caractérisé en ce que l'insertion de l'identifiant de contexte unique (ICU) dans les bits de remplissage du paquet compressé (P) se fait en partant du dernier bit du paquet compressé (P) selon une permutation définie.

[Revendication 4] Procédé de gestion de la transmission d'un paquet compressé selon la revendication 3 caractérisé en ce que l'insertion de l'identifiant de contexte unique (ICU) dans les bits de remplissage du paquet compressé (P) se fait en insérant l'identifiant de contexte unique (ICU) dans les bits de remplissage en sens contraire au sens de lecture du paquet compressé (P).

[Revendication 5] Procédé de gestion de la transmission d'un paquet compressé selon l'une des revendications 2 à 4 caractérisé en ce que l'insertion de l'identifiant de contexte unique (ICU) comprend de surcroît une opération de combinaison des bits du paquet compressé (P) autres que les bits de remplissage avec une séquence de bits construite en répétant l'identifiant de contexte unique (ICU).

[Revendication 6] Procédé de gestion de la réception d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé (P'), comprenant la réception du paquet compressé (P') suivie par la lecture dans le paquet compressé (P') d'un identifiant dudit ensemble de règles de compression, dit identifiant de contexte unique (ICU), inséré au préalable dans le paquet compressé (P').

[Revendication 7] Procédé de gestion de la réception d'un paquet compressé selon la revendication 6 caractérisé en ce que le paquet compressé (P') comprend des bits de remplissage, l'identifiant de contexte unique (ICU) ayant été inséré au préalable dans les bits de remplissage du paquet compressé (P'), et en ce que l'identifiant de contexte unique (ICU) est lu en faisant une hypothèse portant sur le nombre de bits de remplissage du paquet compressé (P').

[Revendication 8] Procédé de gestion de la réception d'un paquet compressé selon la revendication 7 caractérisé en ce que le procédé est itératif, une hypothèse différente sur le nombre de bits de remplissage étant faite à chaque itération, et en ce que le procédé s'arrête dès que l'identifiant de contexte unique lu (ICU) selon l'hypothèse permet de décompresser le paquet compressé (P').

[Revendication 9] Procédé de gestion de la réception d'un paquet compressé selon la revendication 8 caractérisé en ce que le procédé itératif commence en partant du nombre maximal possible pour la longueur (L) des bits de remplissage et décroît ce nombre par un à chaque itération.

[Revendication 10] Procédé de gestion de la réception d'un paquet compressé selon l'une des revendications 6 à 9 caractérisé en ce que, les bits du paquet compressé (P') autres que les bits de remplissage ayant été combinés au préalable avec une séquence de bits construite en répétant l'identifiant de contexte unique (ICU), la décompression du paquet compressé (P') comprend une combinaison inverse avec l'identifiant de contexte unique (ICU) lu permettant de retrouver le paquet compressé (P') tel qu'il était avant la combinaison préalable.

[Revendication 11] Entité de gestion (100) de la transmission d'un paquet de données compressé par l'application d'une règle de compression appartenant à un ensemble de règles de compression, dit paquet compressé (P), comprenant les modules suivants :

- Un module (101) d'insertion dans le paquet compressé (P) d'un identifiant de l'ensemble de règles de compression, dit identifiant de contexte unique (ICU) ; et,
- Un module (102) de transmission du paquet compressé (P') ainsi modifié par le module d'insertion (101).

[Revendication 12] Entité de gestion (200) de la réception d'un paquet de données compressé par l'application d'une règle de compression appartenant

à un ensemble de règles de compression, dit paquet compressé (P'), comprenant les modules suivants :

- Un module (201) de réception du paquet compressé (P') ; et,
- Un module (202) de lecture dans le paquet compressé (P') d'un identifiant de l'ensemble de règles de compression, dit identifiant de contexte unique (ICU), inséré au préalable dans le paquet compressé (P').

[Revendication 13] Terminal mobile comprenant une entité de gestion (100) de la transmission d'un paquet de données compressé selon la revendication 11.

[Revendication 14] Terminal mobile comprenant une entité de gestion (200) de la réception d'un paquet de données compressé selon la revendication 12.

[Revendication 15] Programme d'ordinateur apte à être mis en œuvre par une entité de gestion (100), le programme comprenant des instructions de code qui, lorsqu'il est exécuté par un processeur, réalise les étapes du procédé de gestion de la transmission d'un paquet de données compressé défini dans la revendication 1.

[Revendication 16] Programme d'ordinateur apte à être mis en œuvre par une entité de gestion (200), le programme comprenant des instructions de code qui, lorsqu'il est exécuté par un processeur, réalise les étapes du procédé de gestion de la réception d'un paquet de données compressé défini dans la revendication 6.

[Revendication 17] Support de données, sur lequel est enregistré un programme d'ordinateur comprenant une séquence d'instructions pour la mise en œuvre du procédé de gestion de la transmission d'un paquet de données compressé conforme à la revendication 1 lorsqu'il est chargé dans et exécuté par un processeur.

[Revendication 18] Support de données, sur lequel est enregistré un programme d'ordinateur comprenant une séquence d'instructions pour la mise en œuvre du procédé de gestion de la réception d'un paquet de données compressé conforme à la revendication 6 lorsqu'il est chargé dans et exécuté par un processeur.

[Fig. 1]

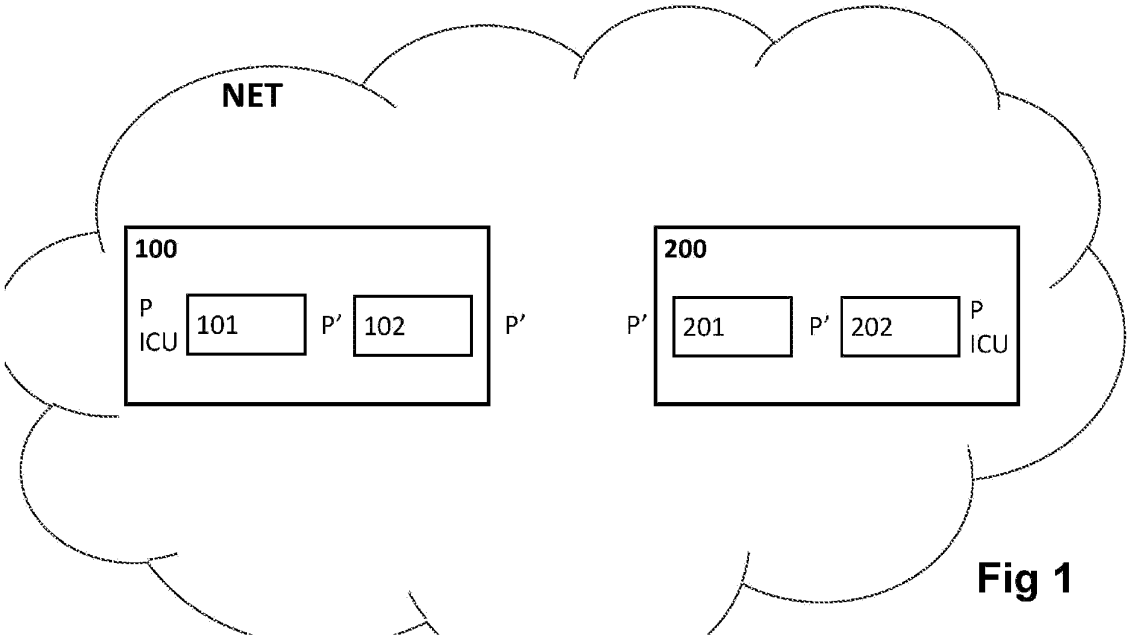


Fig 1

[Fig. 2]

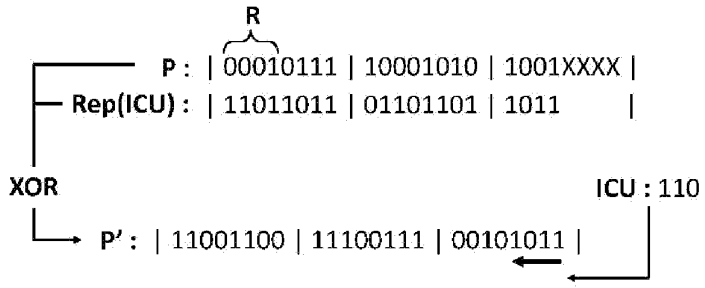


Fig 2

[Fig. 3]

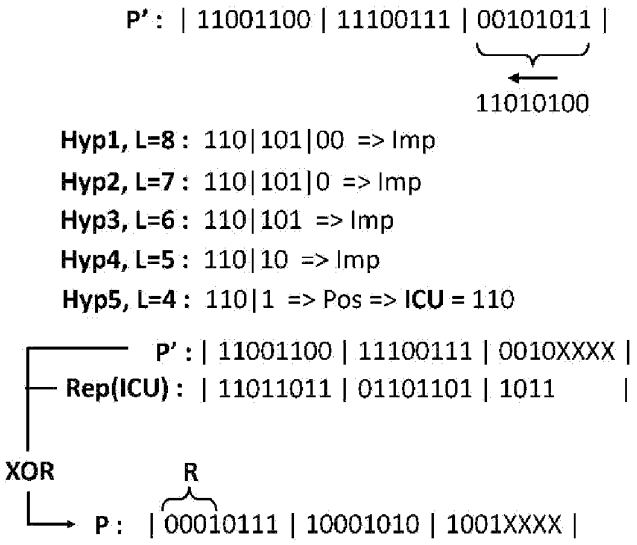


Fig 3

[Fig. 4]

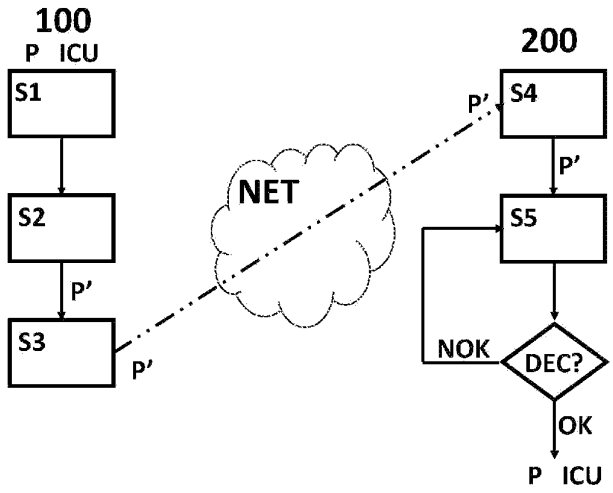


Fig 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/EP2025/070564

A. CLASSIFICATION OF SUBJECT MATTER

H04L 69/04(2022.01)i; ***H03M 7/30***(2006.01)i; *H04L 69/22*(2022.01)n

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; 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	US 2021266385 A1 (LU QIANXI [CN]) 26 August 2021 (2021-08-26)	1,6,11-18
A	the whole document	2-5,7-10
X	US 2020137628 A1 (KANDASAMY ARUNPRABHU [FR] ET AL) 30 April 2020 (2020-04-30)	1,6,11-18
A	the whole document	2-5,7-10
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 January 2022 (2022-01-20), pages 698-740 DOI: 10.1109/COMST.2022.3144473 XP011901225	1,6,11-18
A	the whole document	2-5,7-10

☐ 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	Date of mailing of the international search report
30 September 2025	**10 October 2025**
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 **Aura Marcos, F** Telephone No.

Form PCT/ISA/210 (second sheet) (January 2015)

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/EP2025/070564

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
US 2021266385 A1	26 August 2021	AU 2019443141 A1	27 May 2021
		BR 112021009524 A2	16 November 2021
		CA 3117837 A1	05 November 2020
		CN 112585925 A	30 March 2021
		CN 113133055 A	16 July 2021
		EP 3905627 A1	03 November 2021
		JP 2022515628 A	21 February 2022
		KR 20210113659 A	16 September 2021
		RU 2767321 C1	17 March 2022
		SG 11202105005X A	29 June 2021
		US 2021266385 A1	26 August 2021
		WO 2020220328 A1	05 November 2020
US 2020137628 A1	30 April 2020	CN 111092854 A	01 May 2020
		EP 3644573 A1	29 April 2020
		ES 2881255 T3	29 November 2021
		US 2020137628 A1	30 April 2020

RAPPORT DE RECHERCHE INTERNATIONALE

Demande internationale n°
PCT/EP2025/070564

A. CLASSEMENT DE L'OBJET DE LA DEMANDE
INV. H04L69/04 H03M7/30
ADD. H04L69/22

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 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	US 2021/266385 A1 (LU QIANXI [CN])	1,6,
	26 août 2021 (2021-08-26)	11-18
A	le document en entier	2-5,7-10

X	US 2020/137628 A1 (KANDASAMY ARUNPRABHU	1,6,
	[FR] ET AL) 30 avril 2020 (2020-04-30)	11-18
A	le document en entier	2-5,7-10

X	TOMOSKOZI MATE ET AL: "Packet Header	1,6,
	Compression: A Principle-Based Survey of	11-18
	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	
A	le document en entier	2-5,7-10

☐ 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'appartenenant 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
30 septembre 2025	10/10/2025

Nom et adresse postale de l'administration chargée de la recherche internationale	Fonctionnaire autorisé
Office Européen des Brevets, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Aura Marcos, F

Formulaire PCT/ISA/210 (deuxième feuille) (avril 2005)

RAPPORT DE RECHERCHE INTERNATIONALE

Renseignements relatifs aux membres de familles de brevets

Demande internationale n°
PCT/EP2025/070564

Document brevet cité au rapport de recherche	Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
US 2021266385 A1	26-08-2021	AU 2019443141 A1	27-05-2021
		BR 112021009524 A2	16-11-2021
		CA 3117837 A1	05-11-2020
		CN 112585925 A	30-03-2021
		CN 113133055 A	16-07-2021
		EP 3905627 A1	03-11-2021
		JP 2022515628 A	21-02-2022
		KR 20210113659 A	16-09-2021
		RU 2767321 C1	17-03-2022
		SG 11202105005X A	29-06-2021
		US 2021266385 A1	26-08-2021
		WO 2020220328 A1	05-11-2020
US 2020137628 A1	30-04-2020	CN 111092854 A	01-05-2020
		EP 3644573 A1	29-04-2020
		ES 2881255 T3	29-11-2021
		US 2020137628 A1	30-04-2020