

12 BREVET D'INVENTION B1

54 Procédé et dispositif de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état.

22 Date de dépôt : 17.03.23.

30 Priorité :

60 Références à d'autres documents nationaux
apparentés :

○ Demande(s) d'extension :

71 Demandeur(s) : PSA AUTOMOBILES SA Société
par actions simplifiée (SAS) —FR et INSTITUT
NATIONAL DES SCIENCES APPLIQUEES DE LYON
Établissement public à caractère scientifique, culturel
et professionnel — FR.

43 Date de mise à la disposition du public
de la demande : 20.09.24 Bulletin 24/38.

45 Date de la mise à disposition du public du
brevet d'invention : 07.02.25 Bulletin 25/06.

72 Inventeur(s) : FERNANDEZ BLANCO DAVID, Le
Mouël Frédéric et LIN TRISTA.

56 Liste des documents cités dans le rapport de
recherche :

Se reporter à la fin du présent fascicule

73 Titulaire(s) : STELLANTIS AUTO SAS Société par
actions simplifiée, INSTITUT NATIONAL DES
SCIENCES APPLIQUEES DE LYON Établissement
public à caractère scientifique, culturel et
professionnel.

74 Mandataire(s) : STELLANTIS AUTO SAS Société
par actions simplifiée.



Description

Titre de l'invention : Procédé et dispositif de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état

Domaine technique

[0001] La présente invention concerne l'exécution de tâches relatives à des fonctions sans état en périphérie d'un réseau de communication maillé comprenant des nœuds IoT. En particulier, la présente invention concerne le contrôle d'un tel réseau de communication maillé.

Arrière-plan technologique

[0002] Avec les nouvelles formes d'objets connectés, smartphones et ordinateurs, les applications mises en œuvre sur ces objets connectés ont progressivement accru leur demande de connectivité et de puissance de calcul. Cette augmentation des besoins en ressources des applications s'est accompagnée d'une augmentation constante des caractéristiques embarquées, de la consommation d'énergie et du prix des appareils.

[0003] Dans le but de maintenir les appareils abordables pour le public et d'exécuter des tâches relatives, par exemple, à des fonctions sans état dans un environnement optimal en termes de ressources, le « cloud computing » (« calcul dans le nuage », en français) a étendu ses services en ajoutant des nœuds à la périphérie des réseaux de communication maillés : Ceux-ci sont communément appelés Fog (en anglais, « brouillard » en français) ou Edge (en anglais, « bordure » en français).

[0004] Dans le « cloud computing », les nœuds périphériques supportent moins de ressources de calcul que les autres nœuds du « cloud ». Leur latence et leur charge réseau inférieures sont toutefois suffisantes pour répondre aux demandes des applications actuelles. Même si l'ajout de ressources en périphérie des réseaux de communication maillés résout les problèmes de demande de ressources actuelles, le « cloud computing » repose toujours sur des ressources informatiques qui devront évoluer dans le temps. Le développement du « cloud computing » est par conséquent fragile économiquement et écologiquement car il peut être fortement impacté par de futures pénuries en semi-conducteurs et/ou par des crises de ressources.

[0005] Pour améliorer les capacités de la couche périphérique du « cloud » sans déployer des quantités massives de ressources informatiques supplémentaires, il est nécessaire de tirer profit des ressources qui sont déjà déployées.

[0006] Les appareils IoT (de l'anglais « Internet of Things », en français « l'Internet des objets ») sont une source majeure de ressources informatiques sous-utilisée. Leur nombre augmente de manière exponentielle permettant ainsi de suivre l'évolution des

besoins du « cloud computing ». Les nœuds IoT sont généralement caractérisés par leur manque de ressources embarquées (calcul, énergie et stockage), leur disponibilité intermittente et leurs défauts. Ces nœuds subissent généralement des interruptions de réseau et des plantages inattendus qui les laissent injoignables pendant de longues périodes. Ces nœuds IoT peuvent également rencontrer des fautes byzantines en raison de caractéristiques bas de gamme ou d'attaques malveillantes.

- [0007] Pour utiliser des nœuds IoT pour le « cloud computing », il est donc nécessaire de mettre en œuvre un procédé de contrôle des infrastructures des réseaux de communication maillés pour dépasser les défauts des nœuds IoT et ainsi assurer la sécurité et la disponibilité des calculs de tâches à la périphérie du « cloud ».
- [0008] [Fig.1] illustre schématiquement d'un réseau de communication maillé 1, selon un exemple de réalisation particulier et non limitatif de la présente invention.
- [0009] Le réseau de communication maillé 1 correspond avantageusement à un réseau de type cellulaire, par exemple un réseau cellulaire de téléphonie mobile. Un tel réseau cellulaire est composé d'un ensemble de cellules, chaque cellule correspondant à une zone de couverture géographique d'une antenne de communication (aussi appelée station de base) permettant d'établir des communications radio entre utilisateurs (aussi appelés clients ou usagers, chaque utilisateur étant porteur d'un dispositif de communication mobile) et/ou entre les utilisateurs et le réseau de communication maillé 1. La taille d'une cellule varie et est par exemple comprise entre 1 km et quelques dizaines de kilomètres (par exemple 20 ou 30 kms).
- [0010] Un utilisateur correspond par exemple à une personne physique portant un dispositif de communication mobile de type téléphone intelligent (de l'anglais « smartphone ») ou une tablette. Selon une variante, un utilisateur correspond à un véhicule embarquant un dispositif de communication de type calculateur, par exemple une unité de contrôle télématique, dite TCU (de l'anglais « Telematic Control Unit »), ou un dispositif de communication mobile de type téléphone intelligent embarqué dans le véhicule et connecté à ce dernier via une liaison filaire (par exemple de type USB (de l'anglais « Universal Serial Bus » ou en français « Bus série universel »)) ou sans fil (par exemple de type Bluetooth® ou Wifi®).
- [0011] Le réseau de communication maillé 1 met en œuvre par exemple des communications selon la technologie LTE (de l'anglais « Long-Term Evolution » ou en français « Evolution à long terme »), LTE-Advanced (de l'anglais « Long-Term Evolution - Advanced » ou en français « Evolution à long terme avancée »), C-V2X (de l'anglais « Cellular - Vehicle to Everything » ou en français « Cellulaire – Véhicule vers tout ») qui s'appuie sur la 4G et bientôt la 5G, basées sur LTE.
- [0012] Le réseau de communication maillé 1 est destiné à exécuter des tâches relatives à des fonctions sans état. Ces tâches sont ordonnées par utilisateur au travers de requêtes

émises via le réseau de communication maillé 1. L'utilisateur reçoit alors des réponses à ses requêtes via le réseau de communication maillé 1.

- [0013] La topologie du réseau de communication maillé 1 comprend des nœuds appelés passerelle pour indiquer que ces nœuds correspondent à des passerelles (serveurs) du réseau de communication maillé 1, et des nœuds IoT pour indiquer que ces nœuds correspondent à des appareils IoT.
- [0014] Selon l'exemple de la [Fig.1], la topologie du réseau de communication maillé 1 a une dorsale entièrement connectée entre six passerelles P1 à P6 et un ensemble de sous-réseaux en étoile dans lesquels chaque passerelle P1 à P6 peut atteindre un groupe de nœuds IoT (représentés par des cercles non référencés). Les cercles non référencés en traits pleins représentent des nœuds IoT qui sont soit en sommeil soit éveillés. Les cercles gris clair représentent des nœuds IoT ou des passerelles en panne, Les cercles en pointillé représentent des nœuds IoT ou des passerelles en défaut. Les traits pleins représentent des liens de connexion viables tandis que les traits en pointillé représentent des liens de connexions en défaut.
- [0015] Les passerelles P1 à P6 gèrent un grand ensemble de nœuds IoT. Elles sont considérablement moins nombreuses que les nœuds IoT mais plus ingénieuses et plus disponibles que les nœuds IoT. Leur disponibilité est presque complète. Cependant, même si cela peut se produire beaucoup moins souvent que pour les nœuds IoT, les passerelles peuvent subir des coupures de réseau, des plantages inattendus ou des attaques malveillantes. Ces attaques ont un impact plus important dans le cas du réseau de communication maillé 1 car elles refusent alors généralement le service de la passerelle et des appareils IoT dans son sous-réseau.
- [0016] Ainsi, il est impératif de prévoir un procédé de contrôle du réseau de communication maillé 1 qui gère les défauts des nœuds IoT et des passerelles de manière à assurer une continuité de l'exécution des tâches relatives à des fonctions sans état dans le « cloud ».
- [0017] La définition d'un tel procédé passe par la définition d'un modèle consensuel pour contrôler le réseau de communication maillé 1.
- [0018] Un tel modèle consensuel doit (i) garantir la sécurité du système dans les deux défauts byzantins et non byzantins qui incluent le partitionnement du réseau, les retards, l'impossibilité de joindre un nœud et la perte de paquets, (ii) maintenir la disponibilité du service dès qu'au moins une passerelle et trois nœuds IoT sont disponibles, (iii) ne pas dépendre du moment pour synchroniser la topologie et l'exécution des tâches et mettre en œuvre un mécanisme communication intermédiaire asynchrone et (iv) être complété dès qu'une majorité est atteinte et pouvoir compenser la perte de performances causée par des nœuds plus lents.
- [0019] Les modèles de consensus (algorithmes de consensus) sont nés de la nécessité

d'apporter des solutions non triviales pour parvenir à un accord entre plusieurs machines de la manière la plus efficace et la plus fiable possible et renforcer la tolérance aux pannes des systèmes. Ces algorithmes, grâce à l'utilisation de plusieurs processus et mécanismes, garantissent une valeur de sortie unique entre un ensemble de valeurs possibles proposées par différentes machines (nœuds) sur un ensemble de machines (« cluster » en anglais).

- [0020] Paxos (A.Yousefpour. 2019. All one needs to know about fog computing and related edge computing paradigms: A complete survey. In J. of Syst. Architecture. Elsevier., D.Ongaro. 2014. In search of an understandable consensus algorithm. In 2014 USENIX Annual Technical Conference) est l'un de ces modèles de consensus. Paxos fonctionne en permettant à un groupe de nœuds de parvenir à un accord sur une valeur, en synchronisant ces valeurs et en les rendant apprenables pour tous les nœuds d'un groupe de nœuds. Pour ce faire, Paxos classe les nœuds en trois rôles (proposateur, accepteur et apprenant) ; il est possible qu'un nœud joue plus d'un rôle en même temps. Pour simplifier le comptage des votes et éviter la désynchronisation entre les nœuds lors de la réponse à différentes requêtes simultanées, ce modèle met en œuvre un processus en deux phases : (1) une phase de promesse et, une fois que la majorité des nœuds sont parvenus à un accord sur une valeur, (2) une phase de validation, au cours de laquelle le proposant détermine la réponse finale à une requête et propage le résultat à tous les nœuds du groupe de nœuds pour s'assurer que tout le réseau de communication maillé est synchronisé.
- [0021] Paxos a été largement critiqué en raison de son applicabilité complexe aux systèmes distribués réels en raison de l'hétérogénéité du système et des contraintes de nœud de groupe. De multiples propositions ont tenté d'améliorer cette adaptabilité et de réduire la complexité de Paxos. Cependant, ces propositions sont encore trop complexes pour les systèmes utilisant des nœuds IoT.
- [0022] Ayant pour objectifs d'atteindre des niveaux de performances élevées et de faciliter la compréhension et la mise en œuvre du modèle de consensus, ainsi que l'adaptabilité aux systèmes distribués réels, Raft (L.Lamport. 2019. Time, clocks, and the ordering of events in a distributed system. In Concurrency: the Works of Leslie Lamport) est alors apparu comme une alternative à Paxos pour la gestion de journaux (« log » en anglais) répliqués à travers tous les nœuds d'un groupe de nœuds. Même si Raft se concentre sur le stockage de réplication de données plutôt que sur l'exécution de tâches, la manière de parvenir à des accords sur la synchronisation et l'intégrité des données stockées repose sur les mêmes principes que Paxos.
- [0023] Pour simplifier la répartition des tâches (c'est-à-dire la coordination du système ou la maintenance de la topologie du réseau de communication maillé), Raft propose trois rôles exclusifs et différents des nœuds : (a) « meneur » (« leader » en anglais), le nœud

est alors chargé de coordonner toutes les actions et demandes de gestion du système et de garantir la cohérence et l'organisation, (b) « suiveur » (« follower » en anglais), le nœud écoute alors passivement et suit les ordres du meneur, et (c) « candidat » (« candidate » en anglais), le nœud intervient alors pendant un processus d'élection du meneur. En plus de cette centralisation du pouvoir par le meneur, et du moyen de résoudre les incohérences dans la synchronisation des tables de journaux (« log » en anglais), Raft se base sur des horloges logiques de Lamport (L.Lamport. 2019. Time, clocks, and the ordering of events in a distributed system. In *Concurrency: the Works of Leslie Lamport.*) en ajoutant un nombre de comptage (appelé « terme » (« term » en anglais) dans Raft) qui permet la détection des informations obsolètes et la restauration de la table de journaux (« log » en anglais) des nœuds. Cependant, malgré les simplifications et optimisations apportées par Raft, la complexité et le coût du processus de vote sur lequel repose ce modèle rend difficile son applicabilité aux réseaux de communication maillé comprenant des nœuds IoT qui sont hautement mobiles et défaillants.

- [0024] Pirogue (A.Munir et al. 2017. IFCIoT: Integrated Fog Cloud IoT: A novel architectural paradigm for the future Internet of Things. In *IEEE Consumer Electronics Magazine*) se présente comme une variante de Raft. Pirogue se concentre sur la réduction élevée de son empreinte énergétique tout en conservant une performance similaire à celle de Raft. Dans Pirogue, le processus de vote statique classique est remplacé par un processus de vote linéaire dynamique (S. Jajodia et al. 1990. Dynamic voting algorithms for maintaining the consistency of a replicated database. In *ACM Transactions on Database Systems.*). Ce changement remplace la majorité de quorum statique classique par un quorum dynamique, nécessitant moins de nœuds pour obtenir une majorité en cas de déconnexion d'une passerelle (serveur). Pirogue propose un nouveau rôle des nœuds : témoin (« witness » en anglais), le nœud est alors capable de participer au quorum de consensus en cas de besoin. Ce rôle est mis en œuvre par des appareils à faible consommation n'ayant pas la capacité de stocker l'ensemble des données du système, ce qui réduit l'empreinte énergétique du groupe de nœuds.
- [0025] Ces optimisations permettent à Pirogue d'atteindre des performances, une disponibilité et une tolérance aux pannes proches de celles d'un groupe à cinq nœuds modélisés par Raft avec des configurations à faible empreinte énergétique telles que des groupes à quatre nœuds ou trois nœuds et un témoin.
- [0026] Les modèles de consensus actuels ne peuvent pas fonctionner sur des topologies de réseau de communication maillé 1 comprenant des nœuds IoT hétérogènes du type de celles décrites en relation avec la [Fig.1] compte tenu de la forte hétérogénéité, des ressources restreintes des nœuds IoT, de la disponibilité intermittente des nœuds IoT, de la dynamique élevée des nœuds IoT et du manque de connexion directe entre les

nœuds IoT au sein du réseau de communication maillé 1.

[0027] En effet, les nœuds IoT et les passerelles sont très hétérogènes entre eux en termes de ressources qui dépendent non seulement de leurs caractéristiques matérielles qui peuvent être soit haut de gamme, soit bas de gamme, mais aussi du rapport cyclique et logiciel de ces nœuds et passerelles, puisque le but recherché est d'utiliser les ressources de réserve des nœuds IoT et des passerelles sans interrompre leurs fonctions d'origine.

[0028] Par ailleurs, les nœuds IoT sont fortement restreints en ressources. Même si les passerelles sont beaucoup moins restreintes en ressources, elles représentent néanmoins une petite partie de l'ensemble des nœuds du réseau de communication maillé 1. D'autre part, les appareils IoT se caractérisent, pour la plupart d'entre eux, par leur manque de ressources matérielles (de calcul et de stockage) et leurs contraintes de batterie, puisqu'ils sont généralement alimentés par des batteries. Les modèles de consensus actuels sont généralement déployés sur des nœuds de serveur (passerelles) capables de stocker une copie complète des journaux (« log » en anglais) des nœuds du réseau de communication, ce qui n'est pas le cas pour les nœuds IoT.

[0029] De plus, les nœuds IoT ont une disponibilité intermittente. Malgré les capacités de connectivité élevées des passerelles, les nœuds IoT subissent souvent des coupures de réseau et des plantages inattendus. Leur disponibilité est donc peu fiable, imprévisible et généralement intermittente.

[0030] De plus, les modèles de consensus reposent sur un grand nombre de messages de coordination et de synchronisation transmis sur une dorsale entièrement connectée, mais compte tenu de la taille des topologies de réseau de communication maillé comprenant des nœuds IoT, plusieurs dorsales sont présentes nécessitant que les messages passent par plusieurs nœuds intermédiaires avant d'atteindre leur destination. Ainsi, la réduction du nombre de messages est nécessaire pour améliorer les performances d'un modèle de consensus.

[0031] De plus, les nœuds IoT sont des nœuds hautement mobiles, à la fois physiquement et d'un point de vue réseau. Cela complique le maintien d'une synchronisation complète des tables de journaux (« log » en anglais) comme cela est requis par les modèles de consensus actuels.

[0032] Il est donc nécessaire de définir un modèle de consensus qui puisse être déployé sur un réseau de communication maillé 1 comprenant des passerelles et des nœuds IoT qui répondent aux différentes limitations des modèles de consensus actuels exposées ci-dessus.

[0033] **Résumé de la présente invention**

[0034] Un objet de la présente invention est de résoudre au moins l'un des problèmes de l'arrière-plan technologique décrit précédemment.

- [0035] Un objet de la présente invention est une plate-forme de service informatique collaborative ciblant l'exécution de fonctions sans état et utilisant des nœuds IoT ayant une disponibilité faible à moyenne d'une infrastructure d'un réseau de communication déjà déployée.
- [0036] La présente invention repose sur un modèle de consensus original qui peut s'apparenter aux modèles de consensus de l'état de la technique tels que Raft ou Pirogue. Mais le modèle de consensus de la présente invention étend ces modèles de consensus classiques en ajoutant :
- i. des rôles et une classification des nœuds : Le modèle de consensus divise les nœuds en deux couches en fonction de leurs capacités énergétiques et de calcul : une première couche, qui contient peu les passerelles susceptibles d'être gestionnaire des ressources du réseau de communication maillé, et une deuxième couche, qui contient de nombreux nœuds IoT. De plus, les nœuds IoT sont regroupés dynamiquement en sous-réseau, encore appelé unité de calcul UC, en fonction de leur degré de disponibilité et des leurs capacités de calcul, pour optimiser l'exécution des tâches relatives à des requêtes utilisateur. Ainsi, PCM opère sur un ensemble hétérogène de couches.
 - ii. un contrôle du réseau de communication maillé à deux niveaux : au niveau d'une passerelle alors gestionnaire des ressources qui est élue dynamiquement et au niveau de chaque sous-réseau en élisant un nœud orchestrateur de calcul OC pour chaque sous-réseau.
 - iii. des possibilité de mise à jour dynamique des sous-réseaux et de l'élection de nœud OC pour répondre à un environnement IoT mobile hautement dynamique qui entraîne la modification de chacun des sous-réseaux.
- [0037] Ce modèle de consensus a pour objectif de compenser les limitations des nœuds IoT (c'est-à-dire une forte hétérogénéité, des ressources restreintes, une disponibilité intermittente, une forte dynamique et un manque de connexion directe entre les nœuds au sein de la plateforme) pour garantir la tolérance aux pannes, la synchronisation des informations dans le réseau de communication maillé et la coordination de l'exécution des tâches.
- [0038] La présente invention permet l'exécution de tâches par une infrastructure d'un réseau de communication maillé faiblement connecté en optimisant l'élection d'une passerelle gestionnaire des ressources qui réduit le nombre total de messages échangés.
- [0039] Le contrôle à deux niveaux et le regroupement des nœuds de leur degré de disponibilité et de leur capacité de calcul permettent de compenser leurs erreurs et d'améliorer les performances globales de l'exécution de tâches par le réseau de communication maillé.
- [0040] La présente invention limite la consommation énergétique pour l'exécution de tâches

en périphérie du « cloud » car elle utilise les ressources d'appareils IoT au lieu des nœuds classiques en périphérie du « cloud ».

- [0041] La présente invention centralise la gestion des ressources d'un nombre important de nœuds IoT sans pour autant augmenter considérablement les messages échangés ni ralentir les performances du réseau de communication maillé.
- [0042] Etant donné que la présente invention améliore la tolérance aux pannes et la coordination des tâches, elle participe à réduire les ressources embarquées nécessaires aux véhicules, réduisant ainsi le prix du architectures embarquées de ces véhicules.
- [0043] La présente invention offre de nombreuses possibilités pour le développement de futures applications résilientes Vehicle-to-Anything (V2X) qui pourraient être commercialisées.
- [0044] Selon un premier aspect, la présente invention concerne un procédé de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état, ledit réseau de communication maillé comprenant des nœuds appelés passerelle pour indiquer que ces nœuds correspondent à des passerelles du réseau de communication maillé et des nœuds appelés nœuds IoT pour indiquer que ces nœuds correspondent à des appareils connectés à internet, ledit procédé comprenant les étapes suivantes :
 - obtention d'unités de calcul à partir des nœuds IoT, chaque unité de calcul comprend des nœuds IoT et est rattachée à une passerelle pour former un sous-réseau du réseau de communication maillé ;
 - élection d'une passerelle gestionnaire de ressources parmi les passerelles du réseau de communication maillé, ladite passerelle gestionnaire de ressources minimisant un nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches ;
 - élection, par la passerelle gestionnaire de ressources, d'un nœud orchestrateur de calcul pour chaque unité de calcul, un nœud orchestrateur de calcul d'une unité de calcul étant élu parmi les nœuds de ladite unité de calcul en fonction de performances et de disponibilité des nœuds de ladite unité de calcul ;
 - acceptation, par la passerelle gestionnaire de ressources, d'une requête d'exécution de tâches émise par un utilisateur ;
 - assignation, par la passerelle gestionnaire de ressources, de ladite requête à des nœuds IoT d'une unité de calcul pour l'exécution de tâches de ladite requête par les nœuds IoT de ladite unité de calcul ;
 - collecte, par un nœud orchestrateur de calcul de ladite unité de calcul, d'au moins une réponse intermédiaire à ladite requête, chaque réponse intermédiaire étant soit obtenue par le nœud orchestrateur de calcul soit émise par un nœud IoT de l'unité de calcul à destination du nœud orchestrateur de calcul suite à l'exécution de la tâche par

ledit nœud IoT de l'unité de calcul ;

- obtention, par le nœud orchestrateur de calcul, d'une réponse finale à partir d'un vote à la majorité basé sur les réponses intermédiaires collectées ;
- transmission de la réponse finale par le nœud orchestrateur de calcul à destination de la passerelle gestionnaire de ressources ; et
- transmission de la réponse finale par la passerelle gestionnaire de ressources à destination de l'utilisateur.

- [0045] Selon un exemple de réalisation particulier et non limitatif, si la passerelle gestionnaire de ressources ne reçoit pas de réponse finale relative à ladite requête avant qu'un délai se soit écoulé, ladite requête est assignée à une autre unité de calcul.
- [0046] Selon un exemple de réalisation particulier et non limitatif, le procédé comporte en outre une étape de vérification d'une présence de ladite requête dans une mémoire cache de la passerelle gestionnaire des ressources, et une étape d'envoi d'une réponse finale mémorisée dans la mémoire cache en relation avec ladite requête si ladite requête est mémorisée dans ladite mémoire cache.
- [0047] Selon un exemple de réalisation particulier et non limitatif, le contenu de la mémoire cache de la passerelle gestionnaire des ressources est synchronisé avec les contenus des mémoires cache des autres passerelles du réseau de communication maillé.
- [0048] Selon un exemple de réalisation particulier et non limitatif, la passerelle gestionnaire des ressources diffuse périodiquement aux passerelles du réseau de communication maillé un message comprenant une première table formée d'une union de plusieurs deuxièmes tables, chaque deuxième table représentant une disponibilité et des caractéristiques des nœuds IoT d'un sous-réseau du réseau de communication maillé, si une passerelle ne reçoit aucun message pendant un délai d'expiration d'élection, ladite passerelle déclenche une nouvelle election d'une passerelle gestionnaire de ressources.
- [0049] Selon un exemple de réalisation particulier et non limitatif, une nouvelle election d'une passerelle gestionnaire de ressources est déclenchée pour vérifier si une passerelle gestionnaire des ressources actuelle minimise le nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches.
- [0050] Selon un exemple de réalisation particulier et non limitatif, une nouvelle election d'une passerelle gestionnaire de ressources est déclenchée dès qu'une passerelle du réseau de communication maillé est déconnectée.
- [0051] Selon un exemple de réalisation particulier et non limitatif, le procédé comporte en outre une étape de mise à jour d'une topologie du réseau de communication maillé et de diffusion d'une topologie mise à jour à toutes les passerelles du réseau de communication maillé.
- [0052] Selon un deuxième aspect, la présente invention concerne un dispositif de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des

fonctions sans état, le dispositif comprenant une mémoire associée à un processeur configuré pour la mise en œuvre des étapes du procédé selon le premier aspect de la présente invention.

- [0053] Selon un troisième aspect, la présente invention concerne un véhicule comprenant un dispositif selon le me aspect.
- [0054] Selon un quatrième aspect, la présente invention concerne un programme d'ordinateur qui comporte des instructions adaptées pour l'exécution des étapes du procédé selon le premier aspect de la présente invention, ceci notamment lorsque le programme d'ordinateur est exécuté par au moins un processeur.
- [0055] Un tel programme d'ordinateur peut utiliser n'importe quel langage de programmation, et être sous la forme d'un code source, d'un code objet, ou d'un code intermédiaire entre un code source et un code objet, tel que dans une forme partiellement compilée, ou dans n'importe quelle autre forme souhaitable.
- [0056] Selon un cinquième aspect, la présente invention concerne un support d'enregistrement lisible par un ordinateur sur lequel est enregistré un programme d'ordinateur comprenant des instructions pour l'exécution des étapes du procédé selon le premier aspect de la présente invention.
- [0057] D'une part, le support d'enregistrement peut être n'importe quel entité ou dispositif capable de stocker le programme. Par exemple, le support peut comporter un moyen de stockage, tel qu'une mémoire ROM, un CD-ROM ou une mémoire ROM de type circuit microélectronique, ou encore un moyen d'enregistrement magnétique ou un disque dur.
- [0058] D'autre part, ce support d'enregistrement peut également être un support transmissible tel qu'un signal électrique ou optique, un tel signal pouvant être acheminé via un câble électrique ou optique, par radio classique ou hertzienne ou par faisceau laser autodirigé ou par d'autres moyens. Le programme d'ordinateur selon la présente invention peut être en particulier téléchargé sur un réseau de type Internet.
- [0059] Alternativement, le support d'enregistrement peut être un circuit intégré dans lequel le programme d'ordinateur est incorporé, le circuit intégré étant adapté pour exécuter ou pour être utilisé dans l'exécution du procédé en question.

Brève description des figures

- [0060] D'autres caractéristiques et avantages de la présente invention ressortiront de la description des exemples de réalisation particuliers et non limitatifs de la présente invention ci-après, en référence aux figures 1 à 8 annexées, sur lesquelles :
- [0061] [Fig.1] illustre schématiquement, selon un exemple de réalisation particulier et non limitatif de la présente invention ;
- [0062] [Fig.2] illustre un organigramme des différentes étapes du procédé 10 d'élection

d'une passerelle gestionnaire de ressources selon un exemple de réalisation particulier et non limitatif de la présente invention ;

[0063] [Fig.3] illustre un exemple de réalisation du procédé 10 de la [Fig.1] ;

[0064] [Fig.4] illustre un organigramme des différentes étapes du procédé 40 de contrôle du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention ;

[0065] [Fig.5] illustre un organigramme des différentes étapes du procédé 20 de mise à jour de la topologie du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention ;

[0066] [Fig.6] illustre un exemple de diffusion d'une table SFTT dans le réseau de communication maillé 1 selon un exemple particulier et non limitatif de la présente invention ;

[0067] [Fig.7] illustre un organigramme des différentes étapes du procédé 30 de formation des unités de calcul, selon un exemple de réalisation particulier et non limitatif de la présente invention ;

[0068] [Fig.8] illustre schématiquement un dispositif de contrôle du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention.

[0069] **Description des exemples de réalisation**

[0070] Des procédés et dispositifs vont maintenant être décrits dans ce qui va suivre en référence conjointement aux figures 1 à 8. Des mêmes éléments sont identifiés avec des mêmes signes de référence tout au long de la description qui va suivre.

[0071] La présente invention concerne un procédé 10 de contrôle d'un réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état.

[0072] La présente invention concerne également un procédé 20 de mise à jour de la topologie du réseau de communication maillé 1 pour orchestrer l'exécution tolérante aux pannes des tâches sur des nœuds IoT souvent défectueux ou indisponibles de ce réseau de communication maillé 1.

[0073] La présente invention concerne également un procédé 30 de formation d'unités de calcul par regroupement de nœuds IoT.

[0074] Selon la présente invention, chaque passerelle du réseau de communication maillé 1 mémorise à jour trois tables : une table de topologie globale (GTT de l'anglais « Global Topology Table»), une table de topologie des nœuds du sous-réseau rattaché à la passerelle (SFTT de l'anglais « Sub-network Follower Topology ») et une table de suivi des tâches (TTT de l'anglais Task Tracking Table).

[0075] La table SFTT représente la disponibilité et les caractéristiques des nœuds IoT d'un sous-réseau du réseau de communication maillé 1 comprenant plusieurs nœuds IoT. Ce

sous-réseau est rattaché à une passerelle. La table SFTT est mémorisée par chaque passerelle.

- [0076] La table GTT d'une passerelle est formée par l'union de tables SFTT de d'autres passerelles qui sont connues de ladite passerelle. Un index de table gtt_index est associé à chaque table GTT. Cet index de table est utilisé pour la synchronisation des tables GTT comme on le verra par la suite.
- [0077] La table TTT comprend les tâches à exécuter et leur état d'exécution. Un index de table ttt_index est associé à chaque table TTT. Cet index de table est utilisé pour la synchronisation des tables TTT comme on le verra par la suite.
- [0078] Un nœud IoT est soit dans un état « suiveur » (« follower » en anglais) soit dans un état orchestrateur de calcul (OC, « Calculus Orchestrator » en anglais).
- [0079] Un nœud suiveur est un nœud IoT passif c'est-à-dire qu'il n'émet aucune demande par lui-même. Un nœud suiveur exécute les demandes d'exécution de tâches qui lui sont assignées et envoie une réponse intermédiaire à un nœud OC de l'unité de calcul à laquelle appartient ce nœud suiveur.
- [0080] Un nœud orchestrateur de calcul, dit nœud OC, est un nœud IoT qui est élu pour chaque unité de calcul UC. Un nœud OC d'une unité de calcul UC exécute les demandes d'exécution de tâches qui lui sont assignées et obtient une réponse intermédiaire suite à l'exécution d'une tâche. Un nœud OC collecte également des réponses intermédiaires produites par des nœuds suiveurs de cette unité de calcul UC suite à l'exécution de cette tâche par les autres nœuds suiveurs de l'unité de calcul UC. Le nœud OC produit une réponse finale à partir d'un procédé de vote basé sur les réponses intermédiaires qu'il a obtenues comme on le verra par la suite.
- [0081] Une passerelle peut être soit dans un état de gestionnaire de ressources (RM de l'anglais « Ressources Manager »), soit dans une état « répartiteur » (en anglais « dispatcher »), soit dans une état candidat (en anglais « Candidate »).
- [0082] Une passerelle RM est adaptée pour la mise en œuvre du procédé 20 de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état. En particulier, une passerelle RM est adaptée pour gérer toutes les demandes des utilisateurs (si un utilisateur émet une requête vers une autre passerelle du réseau de communication maillé 1, cette requête est redirigée vers la passerelle RM). Elle est également adaptée pour assigner une tâche à exécuter à des nœuds suiveurs d'une unité de calcul UC, pour sélectionner dynamiquement les nœuds OC et pour retransmettre une réponse finale à un utilisateur émetteur d'une demande d'exécution de tâche.
- [0083] Une passerelle RM est également adaptée pour mettre en œuvre le procédé 20 de mise à jour de la topologie du réseau de communication maillé 1. En particulier, une passerelle RM est adaptée pour gérer la dynamique des nœuds IoT et de diffuser un état d'une topologie du réseau de communication maillé 1, tel qu'il est connu de la

passerelle RM, à destination des autres passerelles du réseau de communication maillé 1. Une passerelle RM synchronise également entre elles les tables GTT et TTT mémorisées par les passerelles du réseau de communication maillé 1.

- [0084] La passerelle RM est également adaptée pour mettre en œuvre un procédé 30 de formation des unités de calcul UC par regroupement des nœuds IoT.
- [0085] Par exemple, la passerelle RM peut déterminer un groupe de nœuds IoT et ajouter de nouveaux nœuds IoT à ce groupe sans consulter d'autres passerelles, n'ayant à propager les informations concernant cet ajout qu'une fois la décision prise. En particulier, une passerelle RM est adaptée pour choisir de manière optimale un nœud OC pour chaque unité de calcul UC.
- [0086] L'utilisation d'une passerelle RM centrale simplifie la gestion des table GTT, SFTT et TTT hautement dynamiques qui sont utilisées pour l'exécution des tâches.
- [0087] Même si des pannes des passerelles sont considérablement moins fréquentes que celles des nœuds IoT, une passerelle RM peut échouer ou se déconnecter du réseau de communication maillé 1, auquel cas une nouvelle passerelle RM est élue.
- [0088] Un répartiteur répond aux requêtes d'une passerelle RM.
- [0089] Selon la présente invention, le temps est divisé en périodes de longueur variable appelées ères. Les ères sont numérotées avec un nombre entier consécutif.
- [0090] Les différents répartiteurs peuvent observer des transitions entre des ères à des moments différents, et ils peuvent ne pas toujours être conscients qu'une ou plusieurs élections se sont produites alors qu'ils étaient hors réseau. Ainsi, les ères agissent comme des horloges logiques, permettant aux passerelles d'identifier les informations les plus récentes.
- [0091] Les ères et les index de table (gtt_index et ttt_index) sont utilisés pour la synchronisation des tables GTT et TTT comme on le verra en détails par la suite.
- [0092] Une passerelle dans l'état candidat, en bref passerelle candidat, est utilisée pour élire une nouvelle passerelle RM. Lors de la conversion d'une passerelle en passerelle candidat, la passerelle incrémente son ère d'une unité, initialise à 0 ses index des tables GTT et TTT, déclenche le procédé d'élection d'une passerelle RM et réinitialise un chronomètre des élections (délai d'élection).
- [0093] Compte tenu de la disponibilité et de la dynamique intermittentes des nœuds IoT, les nœuds du réseau de communication maillé 1 communiquent à l'aide d'appels de procédure à distance asynchrones (RPC de l'anglais Remote Procedure Call) via des dispositifs de messages asynchrones déployés sur les passerelles. Un dispositif de messages asynchrones déployé sur une passerelle permet de transmettre un message de la passerelle vers les nœuds IoT du sous-réseau rattaché à cette passerelle. Les messages sont alors livrés au maximum une fois aux destinataires afin d'améliorer les performances. Un message à destination d'un nœud IoT est toutefois réémis par une

passerelle dans le cas où une erreur s'est produite lors de la connexion du nœud IoT avec la passerelle. Les messages sont conservés jusqu'à ce qu'ils soient consommés ou effacés. Les passerelles peuvent nettoyer les piles de messages des nœuds IoT pour effacer les messages anciens non consommés qui sont arrivés sur la passerelle alors que ces nœuds IoT destinataires n'étaient pas disponibles.

- [0094] Différents messages sont définis. Chaque message comprend un type, des identifiants d'un nœud émetteur et d'un nœud récepteur, une ère de l'information portée par ce message, un index de table et une partie utile (« payload » en anglais).
- [0095] Selon un exemple de réalisation particulier et non limitatif, un message peut être de l'un des quatre types 1, 2 3 ou 4 suivants :
- [0096] configuration d'un nœud IoT suiveur destinée à être connue par un répartiteur ;
- [0097] table GTT destinée à être connue d'un destinataire ;
- [0098] table SFTT destinée à être connue d'un destinataire ; et
- [0099] table TTT destinée à être connue d'un destinataire.
- [0100] L'index de table d'un message peut être est égal à `gtt_index` pour les messages de type 1 à 3 et `ttt_index` pour les messages de type 4.
- [0101] La partie utile du message dépend du type de message. Si type 1 alors la partie comprend un cycle de service suiveur (« duty-cyle » en anglais). Le cycle de service suiveur est défini en millisecondes par trois valeurs <Actif, Occupé, Endormi> avec l'addition de ces trois valeurs inférieure à 120s). Si type 2 alors la partie utile comprend une table GTT. Si type 3 alors la partie utile comprend la table SFTT. Si type 4 alors la partie utile comprend la table TTT.
- [0102] Lors de la réception par un récepteur d'un nouveau message de type 1, 2, 3 ou 4, si les informations du message reçu sont pertinentes, c'est-à-dire si l'ère de ce message est supérieure à l'ère actuelle du récepteur, le récepteur actualise son ère et met à jour ses informations locales. Les informations du message reçu peuvent aussi être pertinentes si l'ère de ce message est égale à l'ère actuelle du récepteur et si l'index de ce message est supérieur à un index de table du récepteur (`gtt_index` ou `ttt_index`). Dans ce cas, le récepteur met à jour ses informations locales c'est-à-dire qu'il actualise l'index de table concerné qui devient alors égal à l'index porté par le message reçu et met à jour la table correspondante en fonction de la partie utile de ce message reçu. Si les informations du message reçu sont pertinentes, un journal d'historique des mises à jour est mis à jour en ajoutant une entrée pour toute nouvelle information du message qui ne serait pas déjà présente dans le journal et en supprimant les entrées du journal qui ne correspondraient pas aux informations portées par le message reçu. Un journal d'historique des mises à jour est mis à jour en ajoutant une entrée pour toute nouvelle information du message qui ne serait pas déjà présente dans le journal et en supprimant les entrées du journal qui ne correspondraient pas aux informations portées par le

message reçu.

- [0103] Si une entrée existante du journal est en conflit avec une nouvelle entrée, l'entrée existante et tout ce qui le suit sont supprimés.
- [0104] Si les informations du message reçues sont obsolètes, c'est-à-dire si l'ère de ce message est inférieure à l'ère actuelle du récepteur ou si l'ère de ce message est égale à l'ère actuelle du récepteur et si l'index de ce message est inférieur à un index de table du récepteur alors le message est ignoré et le récepteur émet sa table GTT (message de type 2) et sa table TTT (message de type 4) à l'émetteur du message reçu.
- [0105] [Fig.2] illustre un organigramme des différentes étapes du procédé de contrôle du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention.
- [0106] Dans une étape 110, des unités de calcul UC sont obtenues, chaque unité de calcul UC comprenant des nœuds IoT du réseau de communication maillé 1. Chaque unité de calcul UC est rattachée à une passerelle du réseau de communication maillé 1 et forme un sous-réseau du réseau de communication maillé.
- [0107] Dans une étape 120, une passerelle est élue passerelle RM parmi les passerelles du réseau de communication maillé 1, ladite passerelle RM minimisant un nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches.
- [0108] Dans une étape 130, un nœud OC est élu par la passerelle RM pour chaque unité de calcul UC, un nœud OC d'une unité de calcul étant élu parmi les nœuds de ladite unité de calcul en fonction de performances et de disponibilité des nœuds de ladite unité de calcul.
- [0109] Dans une étape 140, la passerelle RM accepte une requête d'exécution de tâches émise par un utilisateur.
- [0110] Chaque requête (appelée « Function Execution Request » (FER) en anglais, « Requête d'Exécution de Fonction » en français) contient une fonction sans état à exécuter par les nœuds IoT d'une unité de calcul UC. Une requête FER est composée de scripts de fonction à exécuter et de bibliothèques nécessaires à l'exécution de ces scripts.
- [0111] Dans une étape 150, la passerelle RM assigne la requête à des nœuds d'une unité de calcul UC pour l'exécution des tâches de cette requête par les nœuds de cette unité de calcul UC.
- [0112] Dans une étape 160, le nœud OC de ladite unité de calcul UC collecte au moins une réponse intermédiaire, chaque réponse intermédiaire étant soit obtenue par le nœud OC soit émise par un nœud suiveur de l'unité de calcul à destination du nœud OC suite à l'exécution de la tâche par ledit nœud suiveur de l'unité de calcul UC.
- [0113] Dans une étape 170, le nœud OC détermine une réponse finale à partir d'un vote à la

majorité basé sur des réponses intermédiaires collectées.

- [0114] Dans une étape 180, le nœud OC transmet la réponse finale à destination de la passerelle RM.
- [0115] Dans une étape 190, la passerelle RM transmet la réponse finale à destination de l'utilisateur.
- [0116] Selon une variante du procédé, dans une étape 135, la topologie du réseau de communications maillé 1 est mise à jour et diffusée à toutes les passerelles du réseau de communication maillé 1.
- [0117] Selon une variante, si la passerelle RM ne reçoit pas de réponse finale relative à une requête avant qu'un délai se soit écoulé, la tâche est assignée à une autre unité de calcul UC.
- [0118] Selon une variante, dans une étape 145, la passerelle RM vérifie la présence de chaque requête reçue par la passerelle RM dans sa mémoire cache locale.
- [0119] Si la requête a déjà été demandée et correspond à l'une des requêtes stockées dans une mémoire cache, dans une étape 146, la passerelle RM envoie la réponse finale mémorisée à l'utilisateur et rafraîchit la mémoire cache.
- [0120] Si aucune entrée de la mémoire cache ne correspond à la requête, la passerelle RM ajoute la requête dans sa table TTT de la passerelle en tant que nouvelle entrée, incrémente l'index de table TTT et émet un message asynchrone porteur de la requête à destination des nœuds d'une unité de calcul assignée à l'exécution de la requête.
- [0121] Cette variante donne la priorité aux unités de calcul formées des nœuds les moins défectueux à un moment donné.
- [0122] Lorsque le nœud OC de l'unité de calcul UC sélectionnée répond la réponse finale relative à une requête, la passerelle RM mémorise la réponse finale dans l'entrée de sa table TTT locale relative à la requête, envoie la réponse finale à l'utilisateur et synchronise sa table TTT avec celle des répartiteurs du réseau de communication maillé 1.
- [0123] [Fig.3] illustre un exemple d'exécution des tâches selon un exemple de réalisation particulier et non limitatif de la présente invention.
- [0124] Selon l'exemple de la [Fig.3], un utilisateur Cl1 émet une requête référencée f11 à une passerelle RM. La passerelle RM vérifie si f11 correspond à une entrée de l'une de ses mémoire cache.
- [0125] Supposons que f11 correspond à une requête référencée f2 mémorisée dans la mémoire cache référencée FIFO de la passerelle RM. La passerelle RM envoie alors à l'utilisateur Cl1 la réponse finale Rf2 mémorisée dans la mémoire cache FIFO en relation avec f2. L'ère et l'index de table TTT de la passerelle RM restent inchangés. Sur la [Fig.3], l'ère est égale à 1 (e=1) et l'index de la table TTT est égal à 1 (ttt_index=1).
- [0126] Supposons maintenant que f11 ne corresponde pas à une entrée de la mémoire cache

FIFO. La passerelle RM enregistre alors la nouvelle requête f11 dans la mémoire cache et incrémente de une unité l'index de table TTT `ttt_index` qui passe à 2.

- [0127] La passerelle RM sélectionne également une unité de calcul UC formée, selon l'exemple de la [Fig.3], d'un nœud OC et de deux nœuds suiveurs F1 et F2 et émet f11 au nœud OC et aux nœuds F1 et F2.
- [0128] Le nœud OC exécute la requête f11 et obtient une réponse intermédiaire référencée **a**.
- [0129] Le nœud F1 exécute la requête f11 et fournit une réponse intermédiaire **c** au nœud OC.
- [0130] Le nœud F2 exécute la requête f11 et fournit une réponse intermédiaire **a** au nœud OC.
- [0131] Le nœud OC obtient donc une majorité de vote et considère la réponse intermédiaire **a** comme étant la réponse finale (étape 170). Le nœud OC émet alors la réponse finale **a** à la passerelle RM (étape 180) qui la mémorise dans la mémoire FIFO en relation avec la nouvelle entrée correspondante à la requête f11. La passerelle RM émet alors la réponse finale **a** à destination de l'utilisateur CL1 (étape 190).
- [0132] On peut noter que la mémoire cache peut suivre le principe d'une pile FIFO (First In First Out » en anglais, premier entré premier sorti en français). Dans ce cas, lorsque le nombre maximal d'entrées de la mémoire cache est atteint, toute nouvelle entrée implique la suppression d'une entrée mémorisée dans la mémoire cache. C'est ce qui est illustrée sur la [Fig.3] : L'enregistrement de f11 implique la suppression d'une entrée correspondante à une requête f1.
- [0133] Selon un exemple de réalisation particulier et non limitatif, le contenu de la mémoire cache de la passerelle RM est synchronisée avec les contenus des mémoires cache des autres passerelles au même titre que les tables GTT c'est-à-dire que chaque fois qu'une nouvelle entrée est ajoutée ou supprimée à la mémoire cache d'une passerelle RM, l'index de la mémoire cache est incrémenté d'une unité et des messages asynchrones sont diffusés pour synchroniser les mémoires cache des autres répartiteurs du réseau de communication maillé 1 avec la mémoire cache de la passerelle RM.
- [0134] Selon un exemple de réalisation particulier et non limitatif, le déclenchement d'une élection d'une passerelle RM (étape 120) est basé sur un mécanisme de pulsation (« heartbeat » en anglais) défini comme suit.
- [0135] Lorsqu'une passerelle démarre, elle se positionne dans l'état de répartiteur et diffuse un message de type 2 informant les autres passerelles du réseau de communication maillé 1 du contenu de sa table GTT (des tables SFTT qu'elle connaît). La passerelle reste dans cet état répartiteur tant qu'elle ne reçoit pas un message lui indiquant qu'une passerelle RM est valide ou un message lui indiquant qu'elle doit basculer dans un état candidat.
- [0136] Selon un exemple de réalisation particulier et non limitatif, une passerelle RM diffuse

périodiquement un message de type 2 comprenant sa table GTT aux passerelles du réseau de communication maillé 1 afin de maintenir son autorité et de garder les tables de toutes les passerelles synchronisées. Cependant, si un répartiteur ne reçoit aucun message pendant un délai dit d'expiration d'élection, ce répartiteur suppose qu'aucune passerelle RM ni processus d'élection n'est en cours et déclenche une nouvelle election pour choisir une nouvelle passerelle RM (étape 120).

[0137] Selon un mode de réalisation particulier et non limitatif, une nouvelle election est déclenchée dès la déconnexion d'une passerelle du réseau de communication maillé 1.

[0138] Selon un exemple de réalisation particulier et non limitatif, une nouvelle election est déclenchée pour vérifier si une passerelle RM actuelle minimise le nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches. Dans ce cas, la passerelle RM déclenche une nouvelle election d'une passerelle RM (étape 120) en se considérant comme répartiteur. Si la passerelle RM n'est plus la passerelle candidate issue d'une election, la passerelle RM augmente l'ère et détermine son successeur, en propageant son choix à la fois à la nouvelle passerelle candidate et au reste des répartiteurs du réseau de communication maillé 1. Ce mécanisme de succession permet d'accélérer l'élection d'une nouvelle passerelle RM (étape 120) et de s'assurer au fil du temps de l'optimalité d'une passerelle RM courante.

[0139] Il peut aussi se produire le cas où un répartiteur reçoive un message d'une autre passerelle, RM ou répartiteur à tout moment avec une ère ou un index de table plus élevé que son ère (ou avec la même ère ou index de table provenant d'une passerelle RM différente) tout en ayant déjà considéré une passerelle RM active. Ce cas peut probablement être dû à un comportement malveillant ou à une segmentation du réseau de communication maillé 1.

[0140] Dans ce cas, selon un mode de réalisation particulier et non limitatif, le répartiteur alerte la passerelle RM active, qui fusionne sa table GTT avec la table GTT de l'autre passerelle RM, déclenche à nouveau le processus d'élection pour amener à une election d'une unique passerelle RM.

[0141] Selon un exemple de réalisation particulier et non limitatif, si une passerelle RM ou candidate est "obsolète", cette passerelle est supposée avoir temporairement perdu la connexion réseau. Cette passerelle se synchronise avec l'ère et les indexes de table de la passerelle RM et déclenche une nouvelle election de passerelle RM (étape 120), revenant à l'état de répartiteur. Ce même mécanisme fonctionne dans le sens inverse lors de la réception d'un numéro d'ère ancienne d'un autre répartiteur.

[0142] Selon un exemple de réalisation particulier et non limitatif, si un délai d'expiration d'élection de la passerelle RM s'écoule sans recevoir aucune requête émise par la passerelle RM, un répartiteur incrémente de une unité son ère et lance une nouvelle election d'une nouvelle passerelle RM.

- [0143] Si une passerelle candidate reçoit un message d'un nœud prétendant être une passerelle RM pour la même ère ou une ère supérieure, la passerelle candidate bascule à l'état de répartiteur.
- [0144] Selon un exemple de réalisation particulier et non limitatif, si le délai d'élection s'écoule sans majorité de répartiteurs, les passerelles candidates fusionnent leurs tables GTT, augmente l'ère et déclenche une nouvelle élection de passerelle RM (étape 120).
- [0145] Selon un exemple de réalisation particulier et non limitatif, l'élection d'une passerelle RM est mis en œuvre par un procédé 40 d'élection déterministe d'une passerelle RM basé sur une liste de tous les répartiteurs du réseau de communication maillé 1 ([Fig.4])
- [0146] Dans une étape 410, un répartiteur incrémente son ère de une unité.
- [0147] Dans une étape 420, une passerelle candidate est déterminée parmi les répartiteurs de la liste de répartiteurs.
- [0148] Pour cela, par exemple, une table GTT la plus récente dans le réseau de communication maillé 1 est considérée. Cette table GTT contient une liste de répartiteurs, chacun ayant une liste de nœuds connectés et leurs capacités de calcul. Cette liste exclut la passerelle RM.
- [0149] Initialement, deux variables `optimalDispatcher` et `optimalNumberOfMsgs` sont initialisées à la valeur NULL.
- [0150] Pour chaque répartiteur `dispLeadTmp` de la liste de répartiteurs, des étapes 430, 440 et 450 sont exécutées.
- [0151] Dans l'étape 430, un nombre de message `totalMessages` est initialisé à 0.
- [0152] Dans l'étape 440, pour chaque répartiteur `disp` de la liste de répartiteurs, un nombre `expectMsgs` de messages qu'il est prévu d'échanger entre le répartiteur `DispLeadTmp` et le répartiteur `disp` est déterminé en considérant que le répartiteur `DispLeadTmp` est élu passerelle RM. Le nombre total de messages `totalMsgs` est alors mis à jour par : $\text{totalMsgs} + \text{expectMsgs} * \text{distance}(\text{dispLeadTmp}, \text{disp})$ dans laquelle $\text{distance}(\text{DispLeadTmp}, \text{disp})$ indique une distance entre les répartiteurs `DispLeadTmp` et `disp`.
- [0153] Dans l'étape 450, si le nombre total de messages `totalMsgs` est inférieur à un nombre optimal de message `optimalNumberOfMsgs` alors le répartiteur `DispLeadTmp` est considéré comme une passerelle RM et une variable `optimalDispatcher` est mise à jour avec l'identifiant du répartiteur `DispLeadTmp`. Le nombre optimal de messages `optimalNumberOfMsgs` est aussi égal au nombre total de messages `totalMsgs`.
- [0154] Lorsque chaque répartiteur `DispLeadTmp` de la liste GTT a été considéré, la variable `optimalDispatcher` indique quel est le répartiteur qui a été élu passerelle candidate et la variable `optimalNumberOfMsgs` indique le nombre total de messages relatif à ce répartiteur.

- [0155] Lors de l'exécution de ce processus d'élection 40, certains répartiteurs peuvent ne pas répondre dans un délai inférieur à un délai d'expiration d'élection. Chacun de ces répartiteurs qui a dépassé ce délai d'expiration d'élection exécute à son tour le processus d'élection 40 et informe la passerelle candidate issue de la première élection que, à sa connaissance, cette passerelle candidate est la passerelle RM la plus optimale pour cette nouvelle ère. Si un répartiteur du réseau de communication maillé 1 reçoit l'un de ces messages, il tente d'atteindre cette passerelle candidate. Si il y arrive alors cette passerelle est considérée comme ayant voté pour que cette passerelle candidate soit la nouvelle passerelle RM. Si le répartiteur n'est pas en mesure d'attendre cette passerelle candidate, il déclenchera une élection dans le reste des répartiteurs qui n'ont pas déjà voté pour cette passerelle candidate. Ensuite, ce répartiteur passe à l'état de passerelle candidate jusqu'à ce que l'une des trois événements suivants se produise : (a) il remporte l'élection, (b) un autre répartiteur remporte l'élection ou est déjà la passerelle RM ou (c) une période d'élection s'écoule sans vainqueur.
- [0156] Dans l'évènement a) se produit, une passerelle candidate ne remporte une élection que si la majorité des répartiteurs disponibles dans le réseau de communication maillé 1 votent qu'il s'agit de la passerelle RM la plus optimale. Chaque répartiteur vote pour exactement une passerelle candidate à chaque élection, qui, compte tenu du comportement déterministe du processus d'élection, doit toujours être le même si ces répartiteurs sont synchronisés sur tout le réseau de communication maillé 1; cela ne conduit jamais à un vote par division. Cependant, pour garantir l'exactitude du processus, la règle de la majorité garantit qu'une seule passerelle candidate peut remporter l'élection pour une ère particulière.
- [0157] Dans l'évènement b), en attendant les votes, une passerelle candidate peut recevoir un message d'une autre passerelle candidate prétendant être la passerelle RM. Si l'ère de cette passerelle RM est au moins aussi grande que l'ère actuelle de la passerelle candidate, la passerelle candidate le reconnaît comme une passerelle RM légitime et revient à l'état répartiteur. Cependant, si l'ère de cette passerelle RM est inférieur à l'ère actuelle, le message est rejeté et la passerelle candidate reste dans l'état candidat.
- [0158] Dans l'évènement c), à la fin d'une période d'élection, aucune des passerelles candidates n'a suffisamment de voix pour gagner ou perdre l'élection. Cela peut se produire s'il y a plusieurs fractures du réseau de communication maillé 1 ne permettant pas aux répartiteurs d'avoir une table GTT à jour. Lorsque cela se produit, chaque passerelle candidate envoie sa table GTT au reste des passerelles candidates. Cela fusionne les tables GTT et le processus d'élection est réexécuter, augmentant ainsi l'ère. Cette fois, il est confirmé que tous les répartiteurs participants partagent la même table GTT, et donc la même passerelle candidate. Une fois qu'une passerelle candidate remporte l'élection, cette passerelle propage la décision au reste des répartiteurs.

- [0159] Selon un exemple de réalisation particulier et non limitatif, étant donné que l'élection d'une passerelle RM est calculée selon un procédé déterministe, les votes par division sont inhabituels. Cependant, si cela se produit, les nœuds divisés fusionnent leurs tables GTT et relancent une nouvelle élection d'une passerelle RM.
- [0160] [Fig.5] illustre un organigramme des différentes étapes du procédé 20 de mise à jour de la topologie du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention.
- [0161] Dans une étape 210, la passerelle à laquelle est attaché un sous-réseau détermine la disponibilité de chaque nœud de ce sous-réseau et met à jour sa table SFTT pour indiquer les disponibilités et caractéristiques de chaque nœud disponible du sous-réseau.
- [0162] Il est avantageux de déléguer le maintien de la disponibilité des nœuds d'un sous-réseau à la passerelle à laquelle est rattaché ce sous-réseau car la congestion du réseau de communication maillé 1 et le nombre global de messages échangés sur ce réseau sont réduits.
- [0163] Chaque passerelle suit alors les défauts des nœuds IoT de manière flexible puisque chaque nœud IoT a une disponibilité et une périodicité différentes.
- [0164] Selon un mode de réalisation particulier et non limitatif, chaque nœud IoT de chaque sous-réseau émet périodiquement un message pour indiquer qu'il est disponible. Si une passerelle rattachée à un sous-réseau ne reçoit pas ledit message ou un message en relation avec une exécution d'une tâche d'un nœud IoT dudit sous-réseau pendant une période donnée, ladite passerelle suppose que le nœud IoT s'est déconnecté et le purge de sa table SFTT. De même, si une passerelle rattachée à un sous-réseau reçoit un message pour la première fois en relation avec une exécution d'une tâche d'un nœud IoT dudit sous-réseau, ladite passerelle suppose que le nœud IoT est connecté et l'ajoute à sa table SFTT.
- [0165] Dans une étape 220, les tables SFTT des passerelles sont diffusées aux passerelles du réseau de communication maillé 1 pour que la nouvelle topologie d'un sous-réseau soit connue de l'ensemble des passerelles du réseau de communication maillé 1.
- [0166] [Fig.6] illustre un exemple de diffusion d'une table SFTT dans le réseau de communication maillé 1 selon un exemple particulier et non limitatif de la présente invention.
- [0167] Supposons dans cet exemple, que le réseau de communication maillé 1 comprenne une passerelle RM et quatre répartiteurs D2 à D5.
- [0168] Supposons également que la passerelle RM comprenne une table local GTT formée de l'union de sa table SFTT locale référencée SFTT1 et des tables SFTT de chacun des répartiteurs D2 à D5 référencées respectivement SFTT2, SFTT3, SFTT4 et SFTT5. L'ère courante de la passerelle RM est égale à 1 ($e=1$) et l'index de table GTT est égal

à 1 ($i=1$).

- [0169] Dans étape 221, la passerelle RM diffuse périodiquement un message de type 2 pour diffuser sa table GTT à l'ensemble des autres répartiteurs du réseau de communication maillé 1.
- [0170] Une fois ce message reçu par un répartiteur, dans étape 222, si les informations du message reçu sont pertinentes, ce répartiteur met à jour sa table GTT et les informations qu'ils mémorisent en fonction des informations pertinentes portées par ce message reçu.
- [0171] Par exemple, ce répartiteur met à jour l'ère ($e=1$) et l'index de table GTT ($i=1$) ainsi que sa table GTT en fonction de la table GTT portée par le message de type 2.
- [0172] Selon l'exemple de la [Fig.6], le répartiteur D2 stocke ainsi sa table SFTT locale référencée SFTT2, la table GTT référencée GTT1 portée par le message reçu. Chaque autre répartiteur D3 à D5 en fait de même.
- [0173] Dans une étape 223, chaque répartiteur émet un message de type 3 porteur de sa table SFTT à destination de la passerelle RM.
- [0174] Dans une étape 224, la passerelle RM compare la table SFTT de chaque message reçu avec des informations qu'elle connaît du sous-réseau correspondant à cette table SFTT (tables SFTT locales formant la table GTT locale).
- [0175] Si la passerelle RM ne trouve aucun changement (cas 1 sur la [Fig.6]), la passerelle RM ignore le contenu des messages reçus.
- [0176] Si la passerelle RM trouve un changement, dans étape 225, la passerelle RM fusionne les changements dans sa table GTT locale et augmente l'index de la table GTT. L'index n'augmente que d'une unité à chaque période, quel que soit le nombre de changements détectés.
- [0177] Selon l'exemple de la [Fig.6], les répartiteurs D3 et D4 ont chacun une table SFTT modifiée par l'apparition ou la disparition d'au moins un nœud IoT de leur sous-réseau. Ces tables SFTT modifiées sont référencées SFTT3' et SFTT4'. Les répartiteurs D3 et D4 émettent alors chacun un message de type 3 à destination de la passerelle RM qui détecte un changement des tables SFTT des répartiteurs D3 et D4. La passerelle RM fusionne alors les changements dans sa table GTT locale (maintenant référencée GTT2) et augmente l'index de la table GTT ($i=2$) (cas 2 sur la [Fig.6]). L'ère de la passerelle RM reste égale à 1 ($e=1$). L'index de table GTT n'augmente que d'une unité à chaque période, quel que soit le nombre de changements détectés.
- [0178] Dans une étape 226, la passerelle RM diffuse la nouvelle table GTT à l'ensemble des répartiteurs du réseau de communication maillé 1.
- [0179] Selon l'exemple de la [Fig.6], les répartiteurs D2 à D5 sont alors informés de la nouvelle table GTT (GTT2) de la passerelle RM lors d'une prochaine diffusion du message de type 2 (étape 221).

- [0180] Puisque ce processus repose sur la comparaison de tables SFTT, il est avantageux de garder la comparaison aussi légère que possible. Cependant, les tailles de ces tables sont différentes. La table SFTT a une liste d'une douzaine de nœuds, et la table GTT a une liste de dizaines de tables SFTT par exemple.
- [0181] Selon une variante du processus de mise à jour de la topologie du réseau de communication maillé 1, la passerelle RM compare le couple <ère, index de table > du message reçu à celui stocké localement, ce qui permet d'identifier quelle est la table la plus récente parmi des tables SFTT et la table GTT locale. Le processus se poursuit ensuite par la comparaison des tables SFTT entre elles une fois que les tables SFTT contenue dans le message reçu est plus récent que la table GTT locale (a une ère plus élevée ou une même ère mais un index de table plus élevé que celui de la table SFTT de la table GTT).
- [0182] [Fig.7] illustre un organigramme des différentes étapes du procédé 30 de formation des unités de calcul par regroupement de nœuds IoT, selon un exemple de réalisation particulier et non limitatif de la présente invention ;
- [0183] La performance de chaque unité de calcul UC dépend fortement de la rapidité de la réception d'au moins la moitié des votes des nœuds de l'unité de calcul UC afin que le nœud OC puisse déterminer une réponse finale à une requête.
- [0184] En conséquence, le procédé de formation d'unité de calcul (étape 110) tente de minimiser l'écart entre les périodes de disponibilité des nœuds.
- [0185] Dans une étape 310, la passerelle RM classe les nœuds IoT en fonction de leur degré de disponibilité et de leurs capacités de calcul : un nœud IoT est dit performant lorsque son degré de disponibilité et ses capacités de calcul sont élevées et dit non performant dans le cas contraire.
- [0186] De plus, compte tenu des capacités limitées des appareils IoT, et afin de perdre le moins de périodes de calcul possible, le procédé de formation d'unité de calcul examine également la distribution des unités de calcul courante pour assigner les nœuds à une unité de calcul de manière à minimiser les changements de cette distribution.
- [0187] Dans une étape 320, un nombre d'unité de calcul N est calculée en fonction d'un nombre de nœuds M par unité de calcul.
- [0188] Selon un mode de réalisation particulier et non limitatif, N est égal à 5.
- [0189] Ce mode de réalisation est avantageux car il permet d'obtenir des performances de calcul optimales selon des tests réalisés par l'inventeur.
- [0190] Dans une étape 330, chaque unité de calcul UC est formée par regroupement de N nœuds IoT du réseau de communication maillé 1, au moins un desdits nœuds étant performant et au moins un desdits nœuds étant non performant.
- [0191] Selon un mode de réalisation particulier et non limitatif, les N nœuds les plus rapides

sont par exemple distribués parmi les unités de calcul UC.

- [0192] Selon un mode de réalisation particulier et non limitatif, chaque unité de calcul comprend un nœud OC.
- [0193] Selon un mode de réalisation particulier et non limitatif, lorsque une unité de calcul comprend 5 nœuds, l'un des nœuds d'une unité de calcul est considéré comme étant un nœud orchestrateur de calcul, deux autres nœuds sont des nœuds performants et deux autres nœuds sont des nœuds non performants.
- [0194] Dans une étape 340, chaque unité de calcul UC forme un sous-réseau du réseau de communication maillé 1 qui est rattaché à une passerelle du réseau de communication maillé 1. La passerelle mémorise une table SFTT représentant la disponibilité et les caractéristiques des nœuds IoT du sous-réseau. Les nœuds IoT de cette unité de calcul UC sont informés de leur rattachement à cette unité de calcul UC.
- [0195] Ce procédé de formation d'unités de calcul permet d'augmenter le nombre de périodes de calcul et d'homogénéiser les capacités de l'infrastructure du réseau de communication maillé 1 tout en cherchant maintenir une distribution antérieure des nœuds.
- [0196] Ce procédé de formation d'unité de calcul peut consommer des ressources de manière importante et ajouter une surcharge au procédé de contrôle du réseau de communication maillé 1. Ainsi, si les nœuds IoT ont une forte intermittence de leur disponibilité, ce processus de formation d'unité de calcul ne peut être exécuté à chaque fois qu'une déconnexion de nœud IoT se produit sans pénaliser fortement les performances du procédé de contrôle du réseau de communication maillé 1.
- [0197] Selon un exemple de réalisation particulier et non limitatif, le procédé de formation d'unité de calcul, ne se produit que si l'une des conditions suivantes est vérifiée :
 - a. une unité de calcul ne contient pas de nœud OC ou de nœud IoT performant ;
 - b. le nombre de nœuds IoT du réseau de communication maillé 1 est inférieur à $3.N$
 - c. (c) lorsqu'une période prédéterminée est écoulée pour réoptimiser la distribution des nœuds IoT.
- [0198] La passerelle RM contrôle les nœuds IoT pour s'assurer que ces règles sont respectées.
- [0199] Selon une variante, pour garder l'infrastructure aussi performante que possible, en palliant les pertes de nœuds, la passerelle RM donne la priorité à l'exécution des tâches vers les unités de calcul UC avec tout leur quorum de nœuds performants actifs.
- [0200] Selon une variante, la passerelle RM informe le nœud OC d'une unité de calcul UC chaque fois qu'un changement se produit dans cette unité de calcul UC concernant le nombre de nœuds de cette unité de calcul.
- [0201] Cette variante optimise les performances du processus de contrôle du réseau de communication maillé 1.

- [0202] Selon une variante, lorsqu'une déconnexion d'un nœud OC d'une unité de calcul UC se produit, la passerelle RM doit choisir un autre nœud OC pour cette unité de calcul UC parmi les nœuds de cette unité de calcul UC pour que cette unité de calcul UC soit opérationnelle jusqu'à ce que le processus de formation d'unité de calcul soit à nouveau exécuté.
- [0203] De cette façon, chaque unité de calcul peut tolérer une déconnexion de nœud hors du quorum des nœuds performants et jusqu'à trois déconnexions si deux d'entre elles sont des nœuds non performants.
- [0204] Une fois qu'un nœud IoT d'une unité de calcul UC reçoit un message porteur d'une requête d'exécution de tâches, ce nœud IoT exécute ces tâches et renvoie son résultat au nœud OC de cette unité de calcul UC. Le nœud OC, qui connaît l'état de l'unité de calcul UC, vérifie alors s'il est en mesure d'obtenir un vote majoritaire.
- [0205] Selon un exemple de réalisation particulier et non limitatif, le procédé de vote utilisé par le nœud OC est une variante du processus de vote connu sous le nom de DLV [S. Jajodia et al. 1990. Dynamic voting algorithms for maintaining the consistency of a replicated database. In ACM Transactions on Database Systems]. DLV définit dynamiquement un seuil de majorité de votes en fonction du nombre de nœuds connectés, ce qui permet d'améliorer la flexibilité et les performances du processus de vote. En cas d'égalité, le nœud OC privilégie sa réponse intermédiaire à une FER ou, si sa réponse intermédiaire ne fait pas partie des réponses intermédiaires ex aequo, la première réponse intermédiaire qui atteint la quantité ex aequo est retenue en tant que réponse finale. L'utilisation du processus de vote basé sur DLV nécessite au moins $n + 2$ serveurs par unité de calcul UC pour pouvoir tolérer n pannes au lieu des $2n + 1$ requis par Raft. Cette propriété permet aux unités de calcul UC de fonctionner avec moins de membres avec un niveau acceptable de tolérance aux pannes, ce qui est critique dans un environnement défaillant et hautement dynamique.
- [0206] [Fig.8] illustre schématiquement un dispositif de contrôle du réseau de communication maillé 1 pour l'exécution de tâches relatives à des fonctions sans état, selon un exemple de réalisation particulier et non limitatif de la présente invention.
- [0207] Le dispositif 23 est par exemple configuré pour la mise en œuvre d'au moins une des étapes d'au moins un procédé 10, 20, 30 ou 40. Les éléments du dispositif 23, individuellement ou en combinaison, peuvent être intégrés dans un unique circuit intégré, dans plusieurs circuits intégrés, et/ou dans des composants discrets. Le dispositif 23 peut être réalisé sous la forme de circuits électroniques ou de modules logiciels (ou informatiques) ou encore d'une combinaison de circuits électroniques et de modules logiciels.
- [0208] Selon différents modes de réalisation particuliers, le dispositif 23 est couplé en communication avec d'autres dispositifs ou systèmes similaires, par exemple les cal-

culateurs 20, 21 et/ou 22 et/ou avec des dispositifs de communication, par exemple une TCU (de l'anglais « Telematic Control Unit » ou en français « Unité de Contrôle Télématique »), par exemple par l'intermédiaire d'un bus de communication ou au travers de ports d'entrée / sortie dédiés.

- [0209] Le dispositif 23 comprend un (ou plusieurs) processeur(s) 230 configurés pour exécuter des instructions pour la réalisation d'au moins une partie des étapes d'au moins un procédé 10, 20, 30 ou 40 et/ou pour l'exécution des instructions du ou des logiciels embarqués dans le dispositif 23. Le processeur 230 peut inclure de la mémoire intégrée, une interface d'entrée/sortie, et différents circuits connus de l'homme du métier. Le dispositif 23 comprend en outre ou est en association avec au moins une mémoire 231 correspondant par exemple à une mémoire volatile et/ou non volatile et/ou comprend un dispositif de stockage mémoire qui peut comprendre de la mémoire volatile et/ou non volatile, telle que EEPROM, ROM, PROM, RAM, DRAM, SRAM, flash, disque magnétique ou optique.
- [0210] Le code informatique du ou des logiciels embarqués comprenant les instructions à charger et exécuter par le processeur est par exemple stocké sur la mémoire 231.
- [0211] Selon un mode de réalisation particulier et non limitatif, le dispositif 23 comprend un bloc 232 d'éléments d'interface pour communiquer avec des dispositifs externes, par exemple un dispositif de calcul distant. Les éléments d'interface du bloc 232 comprennent une ou plusieurs des interfaces suivantes :
- i. interface radiofréquence RF, par exemple de type Bluetooth® ou Wi-Fi®, LTE (de l'anglais « Long-Term Evolution » ou en français « Evolution à long terme »), LTE-Advanced (ou en français LTE-avancé) ;
 - ii. interface USB (de l'anglais « Universal Serial Bus » ou « Bus Universel en Série » en français) ;
 - iii. interface HDMI (de l'anglais « High Definition Multimedia Interface », ou « Interface Multimedia Haute Definition » en français) ;
 - iv. interface LIN (de l'anglais « Local Interconnect Network », ou en français « Réseau interconnecté local »).
- [0212] Selon un autre mode de réalisation particulier, le dispositif 23 comprend une interface de communication 233 qui permet d'établir une communication avec d'autres dispositifs, tels que les calculateurs 20, 21 et/ou 22 embarqués du véhicule via un canal de communication 234. L'interface de communication 233 correspond par exemple à un transmetteur configuré pour transmettre et recevoir des informations et/ou des données via le canal de communication 234. L'interface de communication 233 correspond par exemple à un réseau filaire de type CAN (de l'anglais « Controller Area Network » ou en français « Réseau de contrôleurs »), CAN FD (de l'anglais « Controller Area Network Flexible Data-Rate » ou en français « Réseau de

contrôleurs à débit de données flexible »), FlexRay (standardisé par la norme ISO 17458) ou Ethernet (standardisé par la norme ISO/IEC 802-3).

- [0213] Bien entendu, la présente invention ne se limite pas aux exemples de réalisation décrits ci-avant mais s'étend à un procédé de de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état qui inclurait des étapes secondaires sans pour cela sortir de la portée de la présente invention. Il en serait de même d'un dispositif configuré pour la mise en œuvre d'un tel procédé.
- [0214] La présente invention concerne également un véhicule, par exemple automobile ou plus généralement un véhicule autonome à moteur terrestre, comprenant le dispositif de la [Fig.8].

Revendications

[Revendication 1]

Procédé de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état, ledit réseau de communication maillé comprenant des nœuds appelés passerelle pour indiquer que ces nœuds correspondent à des passerelles du réseau de communication maillé et des nœuds appelés nœuds IoT pour indiquer que ces nœuds correspondent à des appareils connectés à internet, ledit procédé comprenant les étapes suivantes :

- obtention (110) d'unités de calcul à partir des nœuds IoT, chaque unité de calcul comprend des nœuds IoT et est rattachée à une passerelle pour former un sous-réseau du réseau de communication maillé ;
- élection (120) d'une passerelle gestionnaire de ressources (RM) parmi les passerelles du réseau de communication maillé, ladite passerelle gestionnaire de ressources (RM) minimisant un nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches ;
- élection (130), par la passerelle gestionnaire de ressources (RM), d'un nœud orchestrateur de calcul pour chaque unité de calcul, un nœud orchestrateur de calcul d'une unité de calcul étant élu parmi les nœuds de ladite unité de calcul en fonction de performances et de disponibilité des nœuds de ladite unité de calcul ;
- acceptation (140), par la passerelle gestionnaire de ressources (RM), d'une requête d'exécution de tâches émise par un utilisateur ;
- assignation (150), par la passerelle gestionnaire de ressources (RM), de ladite requête à des nœuds IoT d'une unité de calcul pour l'exécution de tâches de ladite requête par les nœuds IoT de ladite unité de calcul ;
- collecte (160), par un nœud orchestrateur de calcul (OC) de ladite unité de calcul, d'au moins une réponse intermédiaire à ladite requête, chaque réponse intermédiaire étant soit obtenue par le nœud orchestrateur de calcul soit émise par un nœud IoT de l'unité de calcul à destination du nœud orchestrateur de calcul (OC) suite à l'exécution de la tâche par ledit nœud IoT de l'unité de calcul ;
- obtention (170), par le nœud orchestrateur de calcul, d'une réponse finale à partir d'un vote à la majorité basé sur les réponses intermédiaires collectées ;
- transmission (180) de la réponse finale par le nœud orchestrateur de calcul (OC) à destination de la passerelle gestionnaire de ressources

(RM); et

- transmission (190) de la réponse finale par la passerelle gestionnaire de ressources (RM) à destination de l'utilisateur.

[Revendication 2]

Procédé selon la revendication 1, dans lequel si la passerelle gestionnaire de ressources (RM) ne reçoit pas de réponse finale relative à ladite requête avant qu'un délai se soit écoulé, ladite requête est assignée à une autre unité de calcul.

[Revendication 3]

Procédé selon l'une des revendications précédentes, qui comporte en outre une étape (145) de vérification d'une présence de ladite requête dans une mémoire cache de la passerelle gestionnaire des ressources, et une étape (146) d'envoi d'une réponse finale mémorisée dans la mémoire cache en relation avec ladite requête si ladite requête est mémorisée dans ladite mémoire cache.

[Revendication 4]

Procédé selon la revendication 3, dans laquelle le contenu de la mémoire cache de la passerelle gestionnaire des ressources est synchronisé avec les contenus des mémoires cache des autres passerelles du réseau de communication maillé.

[Revendication 5]

Procédé selon l'une des revendications précédentes, dans lequel la passerelle gestionnaire des ressources diffuse périodiquement aux passerelles du réseau de communication maillé un message comprenant une première table (GTT) formée d'une union de plusieurs deuxièmes tables (SFTT), chaque deuxième table représentant une disponibilité et des caractéristiques des nœuds IoT d'un sous-réseau du réseau de communication maillé, si une passerelle ne reçoit aucun message pendant un délai d'expiration d'élection, ladite passerelle déclenche une nouvelle election (120) d'une passerelle gestionnaire de ressources (RM).

[Revendication 6]

Procédé selon l'une des revendications précédentes, dans lequel une nouvelle election (120) d'une passerelle gestionnaire de ressources (RM) est déclenchée pour vérifier si une passerelle gestionnaire des ressources actuelle minimise le nombre de messages échangés dans le réseau de communication maillé pour l'exécution de tâches.

[Revendication 7]

Procédé selon l'une des revendications précédentes, dans lequel une nouvelle election (120) d'une passerelle gestionnaire de ressources (RM) est déclenchée dès qu'une passerelle du réseau de communication maillé est déconnectée.

[Revendication 8]

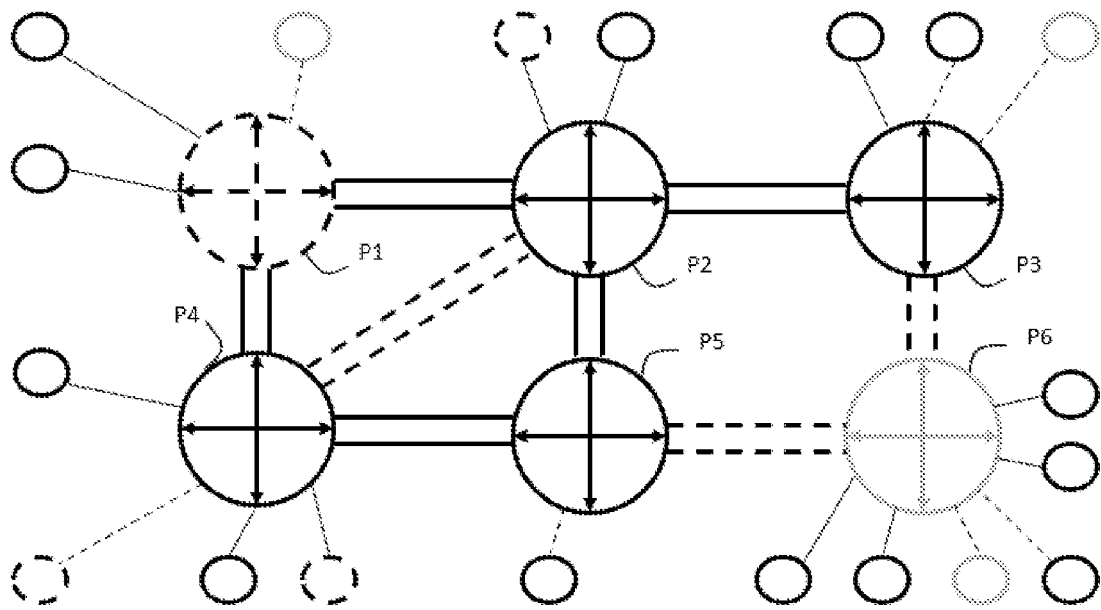
Procédé selon la revendication 1, qui comporte en outre une étape (135) de mise à jour d'une topologie du réseau de communication maillé et de diffusion d'une topologie mise à jour à toutes les passerelles du réseau

de communication maillé.

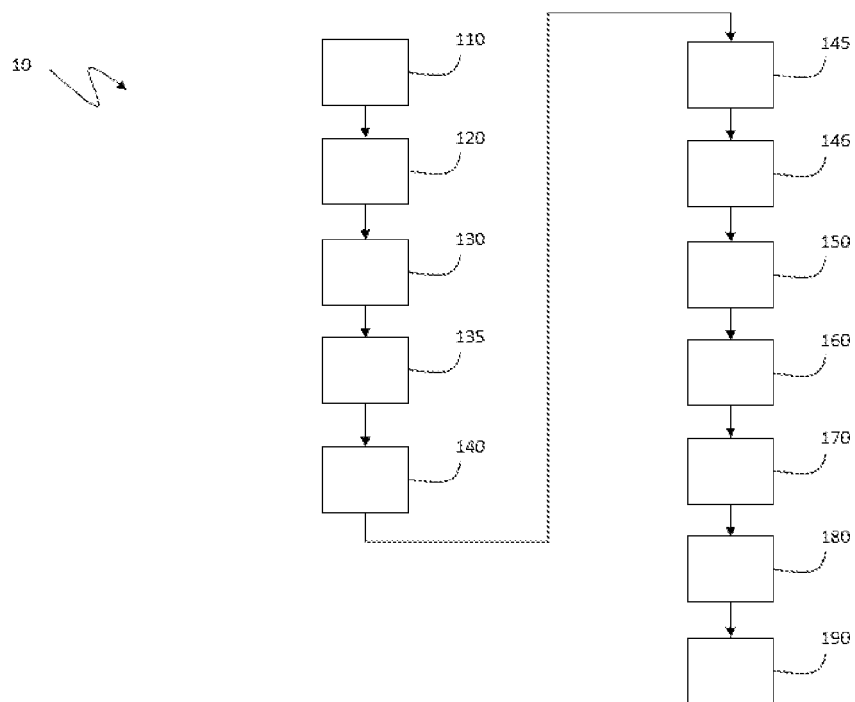
[Revendication 9] Dispositif de contrôle d'un réseau de communication maillé pour l'exécution de tâches relatives à des fonctions sans état comprenant une mémoire associée à au moins un processeur configuré pour la mise en œuvre des étapes du procédé selon l'une des revendications 1 à 8.

[Revendication 10] Véhicule comprenant un dispositif de la revendication 9.

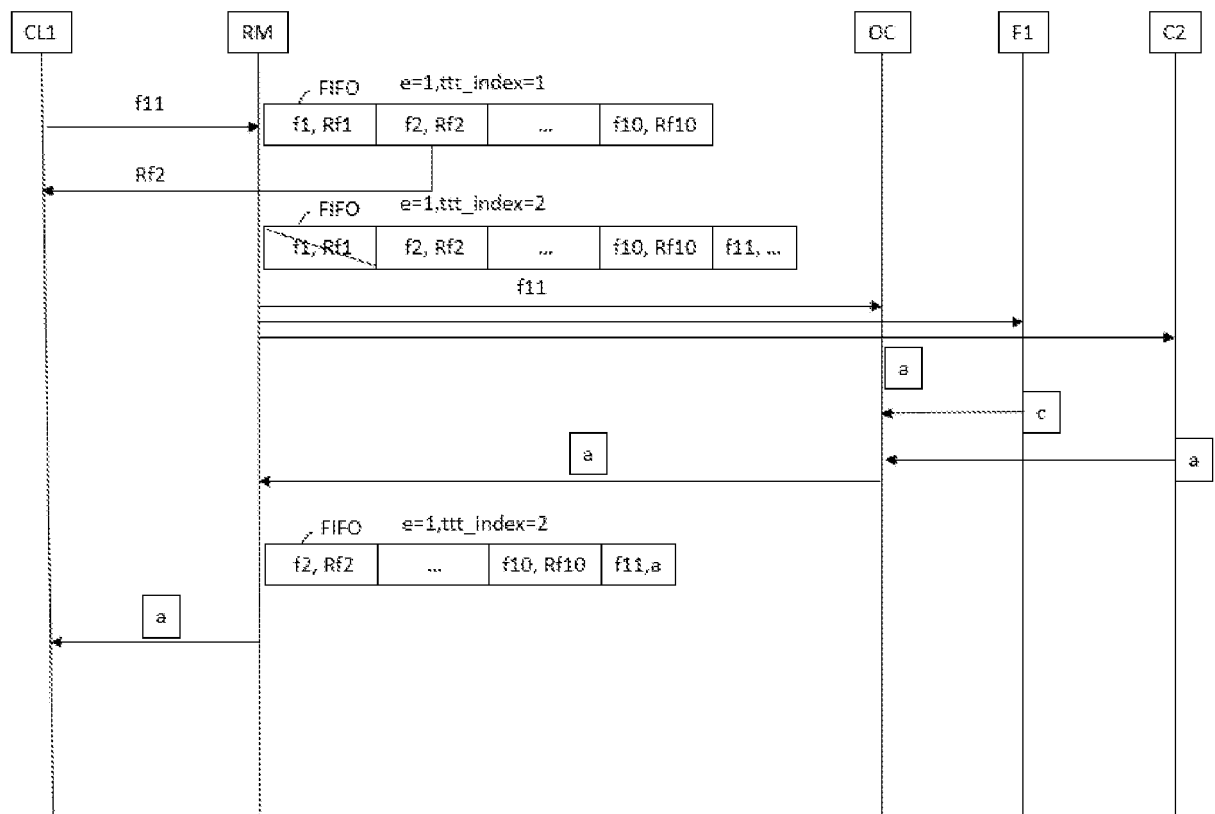
[Fig. 1]



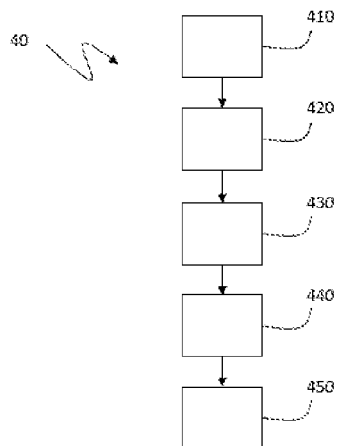
[Fig. 2]



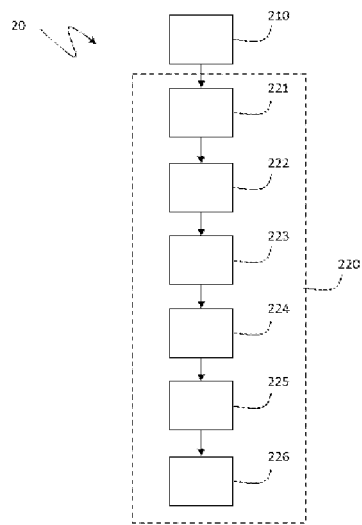
[Fig. 3]



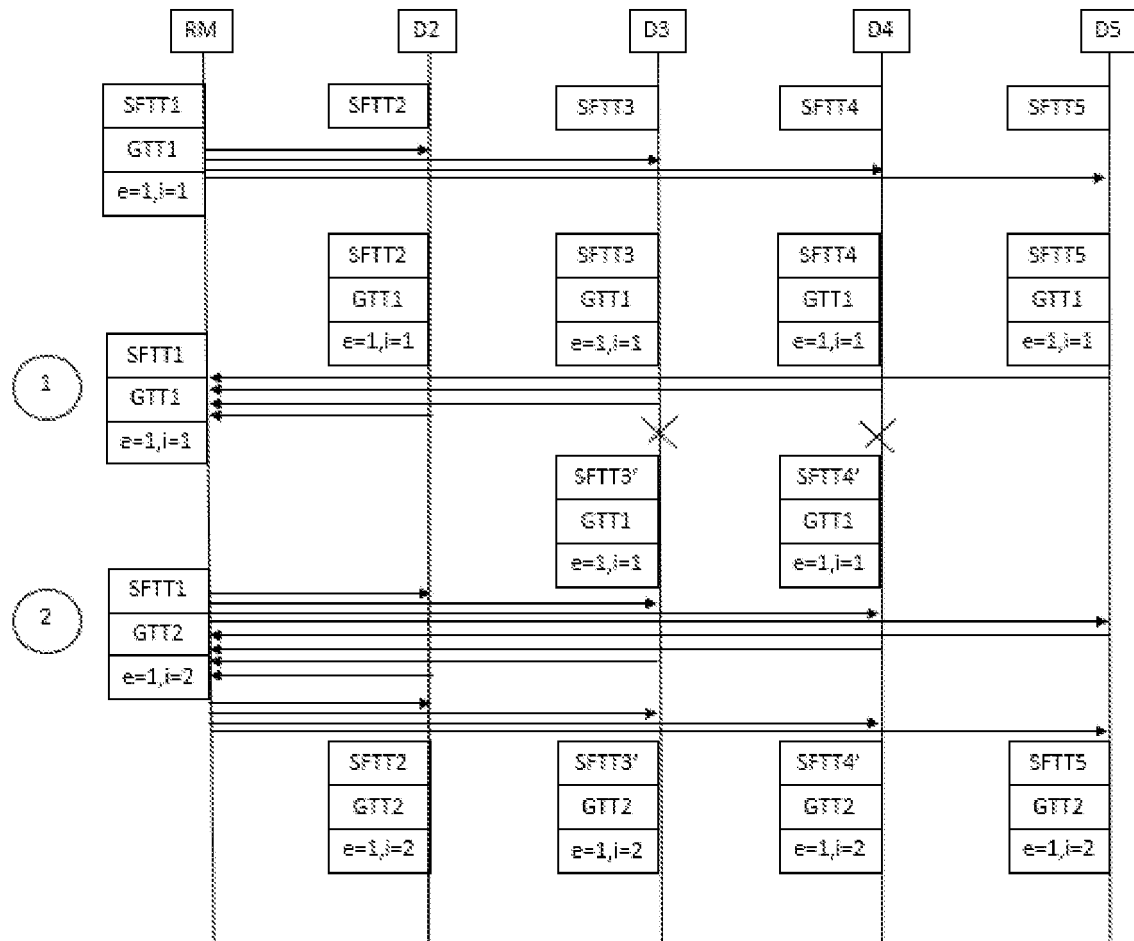
[Fig. 4]



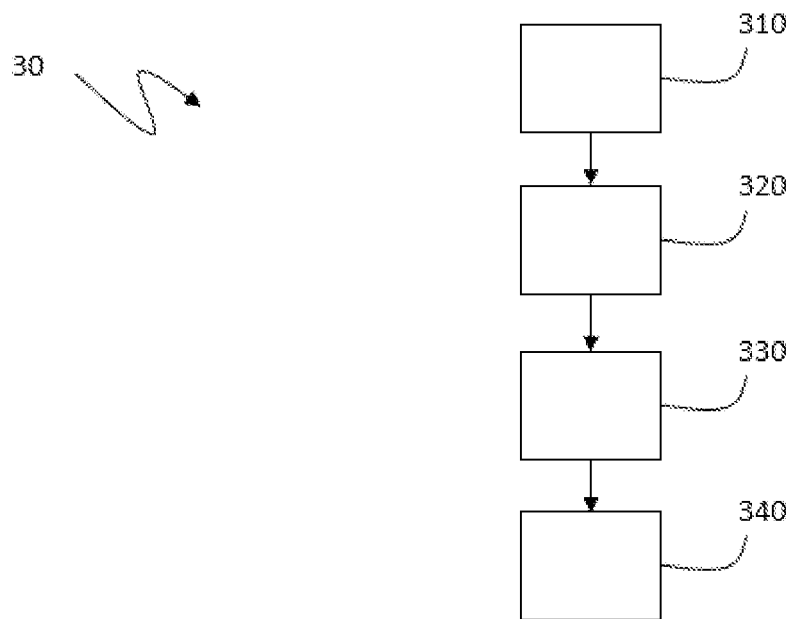
[Fig. 5]



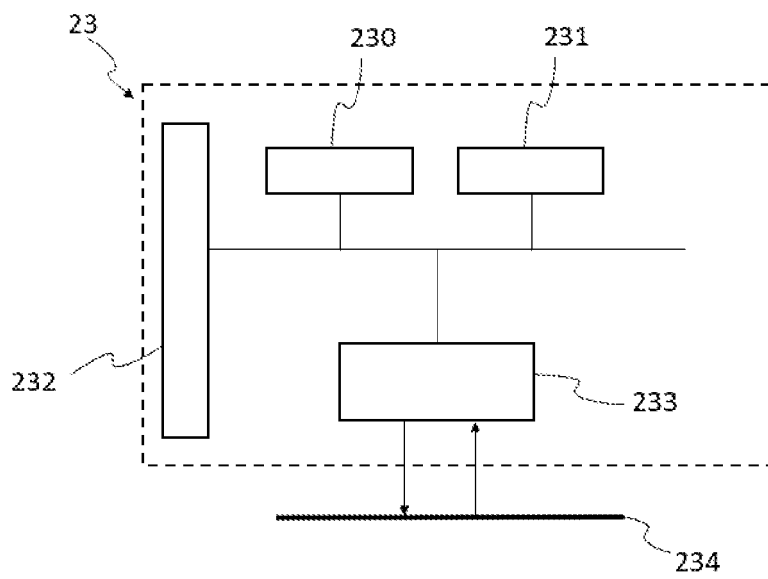
[Fig. 6]



[Fig. 7]



[Fig. 8]



RAPPORT DE RECHERCHE

articles L.612-14, L.612-53 à 69 du code de la propriété intellectuelle

OBJET DU RAPPORT DE RECHERCHE

L'I.N.P.I. annexe à chaque brevet un "RAPPORT DE RECHERCHE" citant les éléments de l'état de la technique qui peuvent être pris en considération pour apprécier la brevetabilité de l'invention, au sens des articles L. 611-11 (nouveau) et L. 611-14 (activité inventive) du code de la propriété intellectuelle. Ce rapport porte sur les revendications du brevet qui définissent l'objet de l'invention et délimitent l'étendue de la protection.

Après délivrance, l'I.N.P.I. peut, à la requête de toute personne intéressée, formuler un "AVIS DOCUMENTAIRE" sur la base des documents cités dans ce rapport de recherche et de tout autre document que le requérant souhaite voir prendre en considération.

CONDITIONS D'ETABLISSEMENT DU PRESENT RAPPORT DE RECHERCHE

☐ Le demandeur a présenté des observations en réponse au rapport de recherche préliminaire.

☒ Le demandeur a maintenu les revendications.

☐ Le demandeur a modifié les revendications.

☐ Le demandeur a modifié la description pour en éliminer les éléments qui n'étaient plus en concordance avec les nouvelles revendications.

☐ Les tiers ont présenté des observations après publication du rapport de recherche préliminaire.

☐ Un rapport de recherche préliminaire complémentaire a été établi.

DOCUMENTS CITES DANS LE PRESENT RAPPORT DE RECHERCHE

La répartition des documents entre les rubriques 1, 2 et 3 tient compte, le cas échéant, des revendications déposées en dernier lieu et/ou des observations présentées.

☐ Les documents énumérés à la rubrique 1 ci-après sont susceptibles d'être pris en considération pour apprécier la brevetabilité de l'invention.

☒ Les documents énumérés à la rubrique 2 ci-après illustrent l'arrière-plan technologique général.

☐ Les documents énumérés à la rubrique 3 ci-après ont été cités en cours de procédure, mais leur pertinence dépend de la validité des priorités revendiquées.

☐ Aucun document n'a été cité en cours de procédure.

1. ELEMENTS DE L'ETAT DE LA TECHNIQUE SUSCEPTIBLES D'ETRE PRIS EN CONSIDERATION POUR APPRECIER LA BREVETABILITE DE L'INVENTION

NEANT

2. ELEMENTS DE L'ETAT DE LA TECHNIQUE ILLUSTRANT L'ARRIERE-PLAN TECHNOLOGIQUE GENERAL

BALACHANDRAN SWEE ET AL: "Distributed Consensus to Enable Merging and Spacing of UAS in an Urban Environment",
2018 INTERNATIONAL CONFERENCE ON UNMANNED AIRCRAFT SYSTEMS (ICUAS), IEEE,
12 juin 2018 (2018-06-12), pages 670-675,
XP033396751,
DOI: 10.1109/ICUAS.2018.8453460
[extrait le 2018-08-31]

US 2022/358118 A1 (PALUCH MICHAL [PL] ET AL) 10 novembre 2022 (2022-11-10)

Diego Ongaro ET AL: "In Search of an Understandable Consensus Algorithm",
Proceedings of USENIX ATC '14: 2014 USENIX Annual Technical Conference, June 19–20, 2014 • Philadelphia, PA,
1 juin 2014 (2014-06-01), pages 305-319,
XP055630887,
Extrait de l'Internet:
URL: <https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf>
[extrait le 2019-10-10]

3. ELEMENTS DE L'ETAT DE LA TECHNIQUE DONT LA PERTINENCE DEPEND DE LA VALIDITE DES PRIORITES

NEANT