

12

DEMANDE DE BREVET D'INVENTION

A1

22 Date de dépôt : 21.12.21.

30 Priorité :

43 Date de mise à la disposition du public de la  
demande : 23.06.23 Bulletin 23/25.

56 Liste des documents cités dans le rapport de  
recherche préliminaire : *Se reporter à la fin du  
présent fascicule*

60 Références à d'autres documents nationaux  
apparentés :

○ Demande(s) d'extension :

71 Demandeur(s) : THALES Société anonyme — FR.

72 Inventeur(s) : GAYRAUD Lionel, THIRIET Aurélien et  
BARRERE Rémi.

73 Titulaire(s) : THALES Société anonyme.

74 Mandataire(s) : Lavoix.

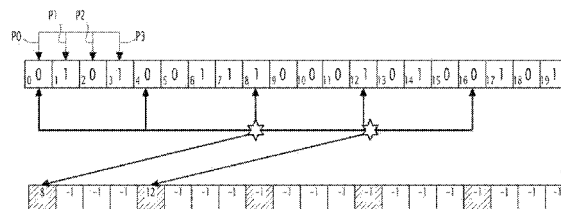
54 PROCÉDE DE DETERMINATION D UNE GRANDEUR PHYSIQUE ET SYSTEME DE DETERMINATION  
ASSOCIE.

57 Procédé de détermination d'une grandeur physique et  
système de détermination associé

La présente invention concerne un procédé de détermination d'une grandeur physique vérifiant un système d'équations respectant une contrainte, le procédé étant mis en œuvre par un système comprenant une pluralité de cœurs, le procédé déterminant la grandeur physique par résolution du système d'équations différentielles en plusieurs itérations comportant :

- une transformation d'une liste courante en une liste compactée comprenant :
- l'initialisation de la liste compactée,
- pour chaque cœur, la détermination pour chaque point de la liste courante présentant un des indices spécifiques au cœur si une condition est vérifiée, et pour chaque point déterminé, modification d'une valeur de la liste compactée,
- la lecture des valeurs de la liste compactée différentes de la valeur d'initialisation,
- des calculs sur les points déterminés, et
- l'actualisation de la liste, et ce jusqu'à vider la liste courante.

Figure pour l'abrégé : figure 3



## Description

### Titre de l'invention : Procédé de détermination d'une grandeur physique et système de détermination associé

- [0001] La présente invention concerne un procédé de détermination d'une grandeur physique. Elle se rapporte également à un système de détermination d'une grandeur physique correspondante.
- [0002] Les équations différentielles d'Hamilton-Jacobi régissent de nombreux phénomènes physiques et se retrouvent donc impliquées dans de multiples applications telles que le traitement d'images, par exemple les images médicales, les géosciences ou la planification de trajectoires.
- [0003] En effet, les équations différentielles d'Hamilton-Jacobi ont des équations associées à une transformation du hamiltonien dans l'espace des phases, et qui permettent de simplifier la résolution des équations du mouvement. Tout domaine impliquant des équations de mouvement est donc concerné par des phénomènes physiques régis par des équations différentielles d'Hamilton-Jacobi.
- [0004] Une équation d'Hamilton-Jacobi prend la forme :
- [0005]  $H(x, \nabla u(x)) = 0$
- [0006] Où :
- [0007] • Où H est un opérateur Hamiltonien,
- $u(x)$  est une solution de l'équation qui est la grandeur physique cherchée lorsque l'on souhaite résoudre l'équation d'Hamilton-Jacobi,
- $x$  est généralement dénommée un nœud, c'est-à-dire une valeur possible pour la variable de  $u(x)$ . Par ailleurs,  $x \in \Omega$ ,  $\Omega$  étant un domaine inclus dans  $\mathbb{R}^n$ ,
- $\nabla$  désigne le gradient de u par rapport à x.
- [0008] Les méthodes les plus répandues pour résoudre une telle équation sont des méthodes causales.
- [0009] Une méthode causale implique l'emploi d'une liste prioritaire au sein de laquelle chaque nœud  $x \in \Omega$  est classé par valeur croissante de  $u(x)$ . Ainsi, une méthode causale repose sur un ordre précis de résolution avec une pluralité d'itérations. Selon cet ordre, à chaque itération, le nœud de plus faible valeur de  $u(x)$  est choisi et considéré ensuite comme « accepté ». Lorsqu'un nœud  $x$  est accepté, la valeur  $u(x)$  n'est plus jamais réévaluée dans les itérations subséquentes de la méthode causale.
- [0010] Du fait de cette évaluation unique de la valeur de chaque nœud, une méthode causale est également qualifiée de méthode à passe unique (plus souvent désignée sous la dénomination anglaise correspondante de « single pass »).
- [0011] Cela permet de diminuer fortement la complexité algorithmique au prix d'une

hypothèse de causalité.

- [0012] Toutefois, dans certains cas, cette hypothèse n'est pas valable, ce qui fait qu'une méthode causale n'est pas universelle.
- [0013] En outre, une méthode causale présente une forte dépendance entre les données, ce qui ne permet pas une implémentation physique efficace des calculs, notamment sur une architecture parallèle de type GP-GPU (l'abréviation « GP-GPU » renvoyant à la dénomination anglaise de « general purpose computing on graphical process units » qui signifie calcul générique sur processeur graphique).
- [0014] Il est également connu des méthodes de résolution itérative n'impliquant pas de faire l'hypothèse de causalité au prix d'un plus grand nombre de calculs. Les méthodes de résolution itérative n'impliquent l'emploi d'aucun mécanisme d'ordonnancement particulier. De fait, ce ne sont pas des méthodes de type passe unique comme les méthodes causales, mais plutôt des méthodes de type passe multiple dans lesquelles un nœud n'est jamais définitivement considéré comme résolu, et est donc susceptible d'être réévalué à tout moment.
- [0015] Ces méthodes de résolution itérative sont donc moins efficaces que les méthodes de résolution causale sur des architectures non parallèles (tel qu'un microprocesseur classique) dans la mesure où le nombre de calculs rend leur exécution lente.
- [0016] Il existe donc un besoin pour un procédé de détermination d'une grandeur physique impliquant une résolution d'un système d'équations différentielles d'Hamilton-Jacobi permettant une implémentation efficace dans un calculateur parallèle.
- [0017] A cet effet, la description décrit un procédé de détermination d'une grandeur physique, la grandeur physique vérifiant un système d'équations différentielles d'Hamilton-Jacobi pour respecter une contrainte variant spatialement dans un environnement prédéfini, le procédé étant mis en œuvre par un système de détermination de la grandeur physique, le système de détermination comprenant une pluralité de cœurs, le procédé comportant :
  - [0018] - une phase de réception de données d'entrées, les données d'entrées comprenant des mesures de la contrainte en plusieurs points de l'environnement prédéfini,
  - [0019] - une phase de détermination de la grandeur physique par utilisation d'une technique de résolution du système d'équations différentielles, pour obtenir la grandeur physique respectant la contrainte, la technique de résolution comportant plusieurs itérations, chaque itération comportant les étapes suivantes:
  - [0020] - la transformation d'une liste courante en une liste compactée courante, la liste courante comportant des points de l'environnement prédéfini auxquels sont attribués un nombre, chaque point étant repéré par un indice, des indices spécifiques étant attribués à chaque cœur de la pluralité de cœurs, l'ensemble des indices de la liste courante étant attribué à l'un des cœurs, la transformation comprenant :

- [0021] - l'initialisation de la liste compactée à une valeur d'initialisation,
- [0022] - pour chaque cœur de la pluralité de cœurs :
- [0023] - la détermination pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur considéré si une condition binaire est vérifiée, pour obtenir un élément déterminé lorsque la condition binaire est vérifiée, et
- [0024] - pour chaque point de la liste courante déterminé, la modification d'une valeur de la liste compactée n'ayant pas été modifiée et présentant l'indice le plus petit de l'ensemble des indices spécifiques au cœur considéré,
- [0025] - la lecture des valeurs de la liste compactée courante différentes de la valeur d'initialisation pour obtenir les points vérifiant la condition binaire,
- [0026] - la mise en œuvre de calculs uniquement sur les points déterminés, et
- [0027] - l'actualisation de la liste en fonction des calculs mis en œuvre,
- [0028] les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.
- [0029] Selon des modes de réalisation particuliers, le procédé de détermination présente une ou plusieurs des caractéristiques suivantes, prise(s) isolément ou selon toutes les combinaisons techniquement possibles :
- [0030] - le nombre d'indices spécifiques est le même pour chaque cœur.
- [0031] - à chaque cœur est attribué un reste respectif de la division euclidienne par le nombre de cœurs, l'ensemble des indices spécifiques au cœur considéré comprenant l'ensemble des indices dont le reste de la division euclidienne de l'indice par le nombre de cœurs est égal au reste respectif.
- [0032] - chaque nombre attribué est un nombre binaire et la condition binaire est que le nombre binaire attribué est égal à une valeur prédéfinie.
- [0033] - la valeur d'initialisation est une valeur négative.
- [0034] - l'opération d'initialisation est également mise en œuvre par chaque cœur uniquement pour les indices spécifiques au cœur considéré.
- [0035] - la mise en œuvre itérative de calculs est réalisée jusqu'à ce que la liste courante soit vide.
- [0036] - les calculs comportent une opération de test de convergence de la valeur du point.
- [0037] - pour chaque point, il est défini un stencil négatif, les calculs comportent également une opération d'élimination du point ayant fait l'objet de l'opération de test de la liste courante, et une opération d'ajout de points uniquement si le test de convergence n'est pas satisfait, les points ajoutés étant de préférence les points du stencil négatif.
- [0038] - les contraintes sont des mesures d'au moins un capteur
- [0039] La description décrit aussi un système de détermination d'une grandeur physique, la grandeur physique vérifiant un système d'équations différentielles d'Hamilton-Jacobi pour respecter une contrainte variant spatialement dans un environnement prédéfini, le système de détermination comprenant une pluralité de cœurs, le système de déter-

mination étant propre à mettre en œuvre un procédé de détermination d'une grandeur physique, le procédé comportant :

- [0040] - une phase de réception de données d'entrées, les données d'entrées comprenant des mesures de la contrainte en plusieurs points de l'environnement prédéfini,
- [0041] - une phase de détermination de la grandeur physique par utilisation d'une technique de résolution du système d'équations différentielles, pour obtenir la grandeur physique respectant la contrainte, la technique de résolution comportant plusieurs itérations, chaque itération comportant les étapes suivantes:
- [0042] - la transformation d'une liste courante en une liste compactée courante, la liste courante comportant des points de l'environnement prédéfini auxquels sont attribués un nombre, chaque point étant repéré par un indice, des indices spécifiques étant attribués à chaque cœur de la pluralité de cœurs, l'ensemble des indices de la liste courante étant attribué à l'un des cœurs, la transformation comprenant :
- [0043] - l'initialisation de la liste compactée à une valeur d'initialisation,
- [0044] - pour chaque cœur de la pluralité de cœurs :
- [0045] - la détermination pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur considéré si une condition binaire est vérifiée, pour obtenir un élément déterminé lorsque la condition binaire est vérifiée, et
- [0046] - pour chaque point de la liste courante déterminé, la modification d'une valeur de la liste compactée n'ayant pas été modifiée et présentant l'indice le plus petit de l'ensemble des indices spécifiques au cœur considéré,
- [0047] - la lecture des valeurs de la liste compactée courante différentes de la valeur d'initialisation pour obtenir les points vérifiant la condition binaire,
- [0048] - la mise en œuvre de calculs uniquement sur les points déterminés, et
- [0049] - l'actualisation de la liste en fonction des calculs mis en œuvre,
- [0050] les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.
- [0051] Des caractéristiques et avantages de l'invention apparaîtront à la lecture de la description qui va suivre, donnée uniquement à titre d'exemple non limitatif, et faite en référence aux dessins annexés, sur lesquels :
- [0052] - [Fig.1] la [Fig.1] est une représentation schématique d'un exemple de système de détermination,
- [0053] - [Fig.2] la [Fig.2] est une illustration schématique des notions de stencil direct et de stencil indirect,
- [0054] - [Fig.3] la [Fig.3] est une vue schématique d'un exemple de mise en œuvre d'une partie d'un exemple de technique de compactage,
- [0055] - [Fig.4] la [Fig.4] est une vue schématique d'un exemple de mise en œuvre d'une autre partie d'un exemple de technique de compactage,
- [0056] - [Fig.5] la [Fig.5] est une vue schématique d'un exemple de mise en œuvre d'une

- autre partie d'un exemple de technique de compactage,
- [0057] - [Fig.6] la [Fig.6] est une vue schématique d'un exemple de mise en œuvre d'une autre partie d'un exemple de technique de compactage, et
- [0058] - [Fig.7] la [Fig.7] est une vue schématique d'un exemple de mise en œuvre d'une autre partie d'un exemple de technique de compactage.
- [0059] La [Fig.1] est une représentation schématique d'un système de détermination 10 d'une grandeur physique.
- [0060] Le système de détermination 10 est propre à déterminer une grandeur physique.
- [0061] Plus précisément, le système de détermination 10 est propre à partir d'au moins une donnée d'entrée 12 à obtenir au moins une donnée de sortie 16, l'au moins une donnée de sortie 16 correspondant à la grandeur physique à déterminer.
- [0062] Une grandeur physique correspond, dans ce contexte, à l'évolution spatiale d'une mesure physique (c'est-à-dire une mesure obtenue par un capteur) dans un environnement prédéfini.
- [0063] Dans la suite, la grandeur physique est notée  $U$  et l'environnement prédéfini est noté  $\Omega$ . Il peut être remarqué que l'environnement prédéfini est souvent dénommé domaine. L'expression « domaine  $\Omega$  » sera donc utilisée. Le domaine  $\Omega$  est un ensemble de points  $x$ .
- [0064] Avec de telles notations, une mesure physique s'écrit donc  $U(x)$  qui est la valeur de la grandeur physique  $U$  évaluée au point  $x$ .
- [0065] Par ailleurs, la grandeur  $U$  vérifie en tout point du domaine  $\Omega$  un système d'équations différentielles dit d'Hamilton-Jacobi ainsi que des conditions initiales (typiquement  $U(x)=0$ ) sur les points d'un ensemble  $D$ .
- [0066] A titre d'illustration, la grandeur physique est liée au mouvement, c'est par exemple, une position ou une vitesse.
- [0067] Selon l'exemple décrit, la grandeur  $U$  est en chaque point  $x$  de  $\Omega$  la longueur de la trajectoire (ou géodésique) la plus courte permettant d'atteindre le point  $x$  depuis un ensemble  $D$  de points appelé ensemble de départ.
- [0068] Dans ce cas, la grandeur physique est la position d'un ensemble de points, la position pouvant être évaluée par une ou plusieurs coordonnées.
- [0069] Selon l'exemple décrit, les données d'entrée 12 comprennent :
- [0070] • un domaine  $\Omega$  contenant un ensemble de points  $x$ ,
- l'opérateur Hamiltonien  $H$  en tout point  $x$  du domaine  $\Omega$ , c'est-à-dire l'équation  $H(x, \nabla u(x))=0$  à vérifier au point  $x$ , et
- pour tout point  $x$  d'un ensemble de départ  $D$ , la valeur  $U(x)$  en ce point.
- [0071] A titre de remarque, il est intéressant de noter que la manière dont sont fournis ces données est indifférente. Typiquement, il est également possible de fournir un opérateur Hamilton paramétrique (une équation, qui sera la même en tout point  $x$  du

domaine à quelques valeurs de paramètres près) ainsi que des tables permettant d'associer à chaque point du domaine les valeurs idoines de ces paramètres (il peut par exemple s'agir d'une carte de coût). Ceci permettra de « paramétrer » l'opérateur d'Hamilton de façon différente d'un point à l'autre du domaine.

- [0072] Plusieurs exemples d'utilisation du système de détermination 10 à des fins de planification de trajectoire sont maintenant décrits. Pour chacun de ces exemples, le système de détermination 10 produit une grandeur  $U(x)$  associée à chaque point  $x$  du domaine  $\Omega$  qui permettra à un autre système, non détaillé ici, de rechercher des trajectoires optimisées partant d'un des points de l'ensemble de départ  $D$  (le plus favorable) et arrivant en tout  $x$  point du domaine  $\Omega$ .
- [0073] Selon un premier exemple, il s'agit de minimiser les risques d'enlèvement d'un véhicule terrestre. Dans un tel exemple, les points du domaine  $\Omega$  sont caractérisés par deux coordonnées et sont donc des 2-uplets de la forme  $(x,y)$ , la trajectoire à déterminer étant la trajectoire du véhicule dans le plan. L'opérateur d'Hamilton associé à tout point  $x$  traduit la plus ou moins grande adhérence du sol en ce point  $x$ . Les conditions initiales sont typiquement  $U(x) = 0$  pour tout point  $x$  correspondant à une position de départ envisageable du véhicule terrestre.
- [0074] Selon un deuxième exemple, il s'agit d'obtenir la trajectoire d'un hélicoptère dans un espace aérien sans percuter un élément de l'espace aérien. Dans un tel exemple, les points sont caractérisés par trois coordonnées et sont donc des 3-uplets de la forme  $(x,y,z)$ , la trajectoire à déterminer étant la trajectoire de l'hélicoptère dans l'espace. L'opérateur d'Hamilton associé à tout point  $x$  traduit le risque de collision avec le sol en ce point. Les conditions initiales sont typiquement  $U(x) = 0$  pour tout point  $x$  correspondant à une position de départ envisageable de l'hélicoptère.
- [0075] Selon un troisième exemple, il s'agit d'obtenir une trajectoire évoluant sur du plat pour un véhicule terrestre. Dans un tel exemple, les points sont caractérisés par trois coordonnées et sont donc des 3-uplets de la forme  $(x,y,\theta)$ , où  $\theta$  est une information de cap (en radians ou degrés). L'opérateur d'Hamilton associé à tout point  $x$  traduit le caractère plus ou moins plat du sol à la position  $(x,y)$  lorsqu'on la traverse via le cap  $\theta$ . En variante ou en complément, il est également possible d'introduire dans l'opérateur d'Hamilton une contrainte de rayon de virage minimum à respecter en ce point. Les conditions initiales sont typiquement  $U(x) = 0$  pour tout point  $x$  correspondant à une position de départ envisageable du véhicule terrestre.
- [0076] A travers ces exemples, il apparaît bien que chaque point est caractérisé par un  $n$ -uplet quelconque, où  $n$  est la dimension du domaine  $\Omega$ , la dimension du domaine dépendant de la grandeur  $U$  à déterminer ainsi que de l'opérateur d'Hamilton à appliquer à chaque point.
- [0077] Pour transformer au moins une donnée d'entrée 12 en au moins une donnée de sortie

16, le système de détermination 10 est propre à mettre en œuvre un procédé de détermination d'une grandeur physique, procédé qui sera décrit ultérieurement.

[0078] Le système de détermination 10 est un calculateur comportant plusieurs cœurs 18.

[0079] Un cœur 18 est un ensemble de circuits capables d'exécuter des programmes de façon autonome.

[0080] L'implémentation physique de chaque cœur 18 est indifférente pour ce qui suit, un cœur pouvant être une unité de traitement graphique (GP-GPU), un microcontrôleur et un processeur de signal numérique (DSP), un circuit logique programmable (comme un circuit intégré spécifique à une application (ASIC), un réseau de portes programmables *in situ* (FPGA), un dispositif logique programmable (PLD) et des réseaux logiques programmables (PLA)), une machine à états, une porte logique et des composants matériels discrets.

[0081] Dans la [Fig.1], seuls deux cœurs 18 sont présentés, sachant que le nombre de cœurs 18 peut être aussi élevé que désiré.

[0082] Le fonctionnement du système de détermination 10 est maintenant décrit en référence à un exemple de mise en œuvre du procédé de détermination illustré par les figures 2 à 7.

[0083] Avant de détailler l'ensemble du procédé, deux techniques, une technique de résolution itérative et une technique de compactage impliquées dans le procédé de détermination vont d'abord être décrites.

[0084] Pour mettre au point le procédé de détermination, le demandeur a choisi une technique de résolution itérative.

[0085] Comme expliqué précédemment, la technique de résolution ne peut pas être une méthode causale du fait de la forte dépendance entre les données. En effet, une telle dépendance empêche une parallélisation et donc une implémentation physique efficace des calculs sur une architecture parallèle.

[0086] Il convient de choisir une technique de résolution qui est une technique de résolution itérative n'impliquant pas de faire l'hypothèse de causalité.

[0087] Parmi les différentes techniques, le demandeur a trouvé comme particulièrement pertinente la technique connue d'un article de F. Bornemann et al. intitulé « *Finite-element discretization of static hamilton-jacobi equations based on local variational principe* » daté de 2006 une technique de résolution adaptative par itérations de Gauss-Seidel. Une telle technique est appelée AGSI, l'abréviation AGSI renvoyant à la dénomination anglaise correspondante de « *Adaptive Gauss Seidel Iterative* ».

[0088] La technique AGSI utilise les notions de stencils qui sont représentés schématiquement sur la [Fig.2].

[0089] Le stencil direct est l'ensemble des points  $y \in \Omega$  impliqués dans le calcul de la valeur  $u(x)$  et est représenté par des rectangles sur la figure 2. Le stencil direct est noté

$V^+(x)$  dans ce qui suit.

- [0090] Le stencil indirect d'un point  $x \in \Omega$  désigne, quant à lui, l'ensemble des points de  $y \in \Omega$  contenant  $x$  dans leur stencil direct :  $V^-(x) = \{y \in \Omega, x \in V^+(y)\}$ . Le stencil indirect est noté  $V^-(x)$ .
- [0091] La technique AGSI utilise également une formule d'Hopf-Lax, qui est une équation permettant de calculer la distance de  $x$  connaissant celle des voisins de  $x$ , au sens du stencil direct. Cette formule est notée  $F(x, V^+(x))$ . La formule d'Hof-Lax exprime l'opérateur d'Hamilton à vérifier au point  $x \in \Omega$  sous la forme d'une équation impliquant  $x$  et l'ensemble de ses voisins au sens du stencil indirect.
- [0092] La technique AGSI revient ainsi à approximer l'opérateur d'Hamilton  $H(x, \nabla u(x))$  en considérant qu'un point  $x$  n'interagit qu'avec ses proches voisins, les points  $y \in \Omega$  de  $V^+(x)$ . Le vecteur gradient  $\nabla u(x)$  ne dépend alors plus que des points  $y \in \Omega$  de  $V^+(x)$ , et l'opérateur  $H(x, \nabla u(x))$  devient une simple équation liant la valeur  $U$  en  $x$  aux valeurs de  $U$  sur les points  $y \in \Omega$  de  $V^+(x)$ .
- [0093] En outre, il est également défini un critère de convergence  $\varepsilon$  arbitraire désignant la tolérance liée à la bonne convergence d'un nœud  $x$  ainsi qu'une liste non ordonnée de nœuds actifs  $L$ . A chaque itération, seuls les nœuds de la liste  $L$  sont évalués.
- [0094] La technique AGSI peut se résumer selon l'ensemble des actions ci-après :
- [0095] *Initialisation :*
- [0096]  $U(x) = \infty, \forall x \in \Omega$
- [0097]  $\forall x_0 \in D, U(x_0) = g(x_0)$
- [0098] –  $\forall x_0 \in D, \text{ajouter } x_0 \text{ à } L$
- [0099] *puis*
- [0100] *Tant que  $L$  n'est pas vide :*
- [0101] – *Retirer  $x$  de  $L$*
- [0102]  $p = U(x)$
- [0103]  $q = F(x, V^+(x))$
- [0104] – *Si  $q < p$  et si  $|q - p| > \varepsilon$  :*
- [0105]  $U(x) = p$
- [0106] •  $\forall y \in V(x), \text{ajouter } y \text{ à } L$
- [0107] Cette série d'instructions peut être résumée de la façon suivante : à chaque itération, tous les nœuds de la liste  $L$  sont réévalués, c'est-à-dire ceux dont au moins un voisin (au sens du stencil indirect) a été amélioré à l'itération précédente (valeur absolue de l'amélioration supérieure au seuil de convergence  $\varepsilon$ ). La réévaluation de chacun de ces nœuds provoquera la réévaluation de tous leurs voisins (au sens du stencil direct). Cette procédure se répète jusqu'à ce que plus aucun nœud ne change de valeurs. Les nœuds initiaux sont l'ensemble des nœuds de  $D$ .

[0108] Il est aussi connu une autre technique de la demande US 2011/0046927 A1, la technique étant itérative et correspondant au schéma de résolution :

[0109] *Tant que  $L$  n'est pas vide :*

[0110] – *Pour tout  $x$  de  $L$  :*

[0111]  $p = U(x)$

[0112]  $q = F(x, V^+(x))$

[0113] • *Si  $q < p$  :  $U(x) = q$*

• *Si  $|q - p| < \varepsilon$  :*

•  *$\forall y \in V(x) :$*

• *Si  $y$  n'appartient pas à  $L$  :*

[0114]  $p = U(y)$

[0115]  $q = F(x, V^+(y))$

[0116] • *Si  $p > q$  :*

[0117]  $U(y) = q$

[0118] • *Ajouter  $y$  à  $L$*

• *Retirer  $x$  de  $L$*

[0119] Pour la suite, une telle technique est appelée technique FIM, l'abréviation FIM renvoyant à la dénomination de « *Fast Iterative Marching* » qui signifie littéralement marche itérative rapide.

[0120] Les deux techniques précitées, à savoir la technique FIM et la technique AGSI, sont très proches, ce qui n'est guère surprenant puisque ces techniques résolvent le même problème et relèvent toutes les deux d'une technique itérative de résolution.

[0121] Néanmoins, malgré leurs similitudes, ces deux techniques ne sont pas strictement équivalentes.

[0122] En effet, entre les deux techniques, il y a une inversion de l'utilisation du test de convergence par rapport au seuil  $\varepsilon$ .

[0123] De fait, pour la technique FIM, la convergence est testée pour chaque point de la liste  $L$ . En outre, l'activation potentielle des voisins correspond au fait que le nœud antérieur respecte le critère de convergence. Au contraire, dans la technique AGSI, l'activation potentielle des voisins n'a lieu que si le nœud antérieur est modifié, ce qui implique que le nœud antérieur ne respecte pas le critère de convergence.

[0124] De plus, le rôle des listes est différent entre les deux techniques.

[0125] Pour le cas de la technique FIM, puisqu'un nœud reste dans la liste sauf s'il a convergé, la liste correspond à l'ensemble des nœuds qui doivent encore converger. Par contraste, dans la technique AGSI, un nœud est systématiquement retiré de la liste,

quel que soit le résultat du test de convergence. Il n'y sera réintroduit que si un nœud dont il dépend (même stencil) réalise une amélioration. Aussi, la liste dans le cas de la technique AGSI peut être comprise comme jouant un mécanisme de requête/acquittement. L'insertion d'un nœud dans la liste peut être perçue comme une requête de mise à jour, qui est acquittée de façon systématique.

[0126] Du fait de ce constat, la technique AGSI a été sélectionnée par le demandeur comme une bonne technique de résolution itérative candidate pour une parallélisation, c'est-à-dire un traitement au moins en partie parallèle dans plusieurs cœurs 18.

[0127] La technique AGSI technique peut être décrite comme la mise en œuvre de deux phases :

[0128] – une phase de réception des données d'entrées suivantes:

- Un domaine  $\Omega$  contenant un ensemble de points  $x$  associés chacun à un indice unique  $i$
- Pour tout point  $x$  du domaine  $\Omega$ , les ensembles  $V^+(x)$  et  $V^-(x)$  appelés stencils directs et indirects, ainsi que la formule d'Hopf-Lax  $F(x, V^+(x))$  en ce point
- Pour tout point  $x$  d'un ensemble  $D$ , la valeur  $U(x)$  en ce point
- Une valeur de seuil  $\varepsilon$

– une phase de détermination de la grandeur physiques  $U(x)$  en tout point de  $\Omega \setminus D$  par utilisation d'itérations AGSI successives, chaque itération comprenant la mise en œuvre de calculs uniquement sur des points contenus dans une liste  $L$  et l'actualisation de la liste en fonction des calculs mis en œuvre, les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.

[0129] En l'espèce, les calculs comportent une opération de réévaluation de la valeur de  $U$  pour tout point de la liste  $L$ , une opération d'élimination de ce point de la liste  $L$ , et une opération d'ajout à la liste  $L$  des points appartenant au stencil direct des points pour lesquels la réévaluation a conduit à une amélioration d'amplitude supérieure à  $\varepsilon$  en valeur absolue.

[0130] Pour des raisons pratiques liées à la parallélisation massive du procédé, la liste  $L$  est implémentée sous la forme d'une table contenant une cellule par point  $x$  du domaine  $\Omega$ . Dans une réalisation, la liste  $L$  est une liste d'entiers (aussi dénommés sous l'appellation anglaise correspondante d'« integer ») prenant les valeurs 0 ou 1 à la position correspondant à l'indice  $i$  du point  $x$ .

[0131] L'utilisation d'une liste contenant autant de cellules que de points  $x$  dans le domaine  $\Omega$  permet à  $N$  processus indépendants d'ajouter un nœud  $x$  à la liste  $L$  en évitant qu'un nœud n'apparaisse plusieurs fois. Ainsi,  $N$  processus indépendants peuvent simultanément écrire 1 dans la cellule du nœud d'indice  $i$  correspondant à un point  $x$ . Le

point  $x$  ne sera introduit qu'une fois et une seule dans la liste  $L$ . Par ailleurs, le fait que les processus écrivent cette valeur simultanément n'est pas gênant car ils y inscrivent tous la valeur 1 (il n'y a pas de conflit en écriture).

- [0132] Afin d'améliorer l'efficacité du procédé de résolution la liste  $L$  est compactée à chaque itération de la phase de détermination de la grandeur physique  $U(x)$ . La phase de détermination de la grandeur physique  $U(x)$  repose sur deux listes notées  $L$  et  $cL$ . La liste  $cL$  est implémentée sous la forme d'une table contenant une cellule par point  $x$  du domaine  $\Omega$ , chaque cellule prenant pour valeur soit l'indice  $i$  d'un point du domaine  $\Omega$ , soit une valeur indiquant que la cellule est vide. Dans une réalisation, les cellules vides contiennent la valeur -1. Un exemple de listes  $L$  et  $cL$  est présenté sur les deux tableaux suivants.

- [0133] [Tableaux1]

| L |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

- [0134] [Tableaux2]

| cL |   |   |   |    |    |    |    |    |    |
|----|---|---|---|----|----|----|----|----|----|
| 2  | 3 | 7 | 8 | -1 | -1 | -1 | -1 | -1 | -1 |

- [0135] La technique de compactage permettant de déterminer  $cL$  à partir de  $L$  choisie par le demandeur est illustrée par les schémas des figures 3 à 7.
- [0136] La liste  $L$  comporte un ensemble de 20 éléments (représentés chacun sous la forme d'une case).
- [0137] Pour la suite, les éléments sont ordonnés de la gauche vers la droite avec un indice qui est positionné en bas à gauche dans la liste  $L$  sur les figures 3 à 6.
- [0138] Le premier indice est 0, le deuxième indice est 1 et ainsi de suite jusqu'au dernier indice de 19.
- [0139] En l'espèce, chaque élément de la liste  $L$  est soit un « 0 » soit un « 1 ». Plus précisément, les éléments de la liste  $L$  dont les indices sont 1, 3, 7, 8, 12, 14, 17 et 19 sont un « 1 ». Les éléments de la liste  $L$  ayant un autre indice sont un « 0 ».
- [0140] Similairement, sur les figures 3 à 6, la liste compactée  $cL$  comprend aussi un ensemble de 20 éléments (également représentés sous la forme d'une case).
- [0141] Le même système d'indice est utilisé pour la liste compactée  $cL$ , les indices n'étant pas repris sur les figures dans un souci de lisibilité des figures.
- [0142] Initialement, chaque élément de la liste compactée  $cL$  est un « -1 ».
- [0143] Les figures 3 à 6 illustrent comment les éléments de la liste compactée  $cL$  sont remplis en fonction de liste  $L$ , chaque [Fig.3] à 6 correspondant à des éléments respectifs.

- [0144] Il convient également de noter que l'opération de remplissage de la liste compactée cL est ici spécifique en ce sens que l'opération de remplissage est une opération de remplacement de la valeur « -1 » lorsqu'au moins une condition est remplie dans la liste L.
- [0145] La [Fig.3] montre comment les éléments d'indice 0, 4, 8, 12 et 16 sont traités, c'est-à-dire tous les éléments dont l'indice est proportionnel à 4 soit égal à 0 modulo 4 ; la [Fig.4] montre comment les éléments d'indice 1, 5, 9, 13 et 17 sont traités, c'est-à-dire tous les éléments dont l'indice est égal à 1 modulo 4 ; la [Fig.5] montre comment les éléments d'indice 2, 6, 10, 14 et 18 sont traités, c'est-à-dire tous les éléments dont l'indice est égal à 2 modulo 4 et la [Fig.6] montre comment les éléments d'indice 3, 7, 11, 15 et 19 sont traités, c'est-à-dire tous les éléments dont l'indice est égal à 3 modulo 4.
- [0146] Du fait de ce partage, l'opération de traitement des éléments de la liste compactée cL visée par la [Fig.3] correspond à un processus dénommé premier processus P0, l'opération de traitement des éléments de la liste compactée cL visée par la [Fig.4] correspond à un processus dénommé deuxième processus P1, l'opération de traitement des éléments de la liste compactée cL visée par la [Fig.5] correspond à un processus dénommé troisième processus P2 et l'opération de traitement des éléments de la liste compactée cL visée par la [Fig.6] correspond à un processus dénommé quatrième processus P3.
- [0147] Chacun de ces processus va être décrit de manière plus détaillée.
- [0148] Comme expliqué précédemment, le premier processus P0 est illustré sur la [Fig.3]. Le premier processus P0 vise à effectuer l'opération de traitement des éléments dont l'indice est égal à 0 modulo 4.
- [0149] Pour cela, dans l'exemple décrit, le premier processus P0 commence par lire la valeur du premier élément de la liste L concerné par ce processus (indice 0) et lit la valeur « 0 ». Le premier processus P0 ne fait alors aucune opération sur la liste compactée cL.
- [0150] Ensuite, le premier processus P0 lit la valeur du deuxième élément de la liste L concerné par ce processus (indice 4) et lit la valeur « 0 ». Le premier processus P0 ne fait alors aucune opération sur la liste compactée cL.
- [0151] Puis, le premier processus P0 lit la valeur du troisième élément de la liste L concerné par le processus (indice 8) et lit la valeur « 1 ». Le premier processus P0 modifie alors la valeur de l'élément de la liste compactée cL ayant le premier indice accessible de la liste compactée cL car c'est la première fois qu'une valeur « 1 » est lue par le premier processus P0. Le premier indice de la liste compactée cL accessible au premier processus P0 est, en l'occurrence, le premier indice de la liste compactée cL (l'indice 0). Le premier processus P0 change alors la valeur initiale « -1 » de l'élément d'indice 0 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la

liste L lu, à savoir la valeur « 8 ».

- [0152] Ensuite, le premier processus P0 lit la valeur du quatrième élément de la liste L concerné par le processus (indice 12) et lit la valeur « 1 ». Le premier processus P0 modifie alors la valeur de l'élément de la liste compactée cL ayant le deuxième indice accessible de la liste compactée cL car c'est la deuxième fois qu'une valeur « 1 » est lue par le premier processus P0. Le deuxième indice de la liste compactée cL accessible au premier processus P0 est, en l'occurrence, le cinquième indice de la liste compactée cL (l'indice 4). Le premier processus P0 change alors la valeur initiale « -1 » de l'élément d'indice 4 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 12 ».
- [0153] Enfin, le premier processus P0 lit la valeur du cinquième élément de la liste L concerné par le processus (indice 16) et lit la valeur « 0 ». Le premier processus P0 ne fait alors aucune opération sur la liste compactée cL.
- [0154] A l'issue du premier processus P0, les valeurs des éléments des autres indices accessibles de la liste compactée cL, en l'occurrence les indices 8, 12 et 16, ne sont pas modifiées et restent donc à la valeur initiale de « -1 ».
- [0155] Comme expliqué précédemment, le deuxième processus P1 est illustré sur la [Fig.4]. Le deuxième processus P1 vise à effectuer l'opération de remplissage sur les éléments dont l'indice est égal à 1 modulo 4.
- [0156] Pour cela, dans l'exemple décrit, le deuxième processus P1 lit la valeur du premier élément de la liste L concerné par le processus (indice 1) et lit la valeur « 1 ». Le deuxième processus P1 modifie alors la valeur de l'élément de la liste compactée cL ayant le premier indice accessible de la liste compactée cL car c'est la première fois qu'une valeur « 1 » est lue par le deuxième processus P1. Le premier indice de la liste compactée cL accessible au deuxième processus P1 est, en l'occurrence, le deuxième indice de la liste compactée cL (l'indice 1). Le deuxième processus P1 change alors la valeur initiale « -1 » de l'élément d'indice 1 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 12 ».
- [0157] Puis, le deuxième processus P1 lit la valeur du deuxième élément concerné par le processus (indice 5) et lit la valeur « 0 ». Le deuxième processus P1 ne fait alors aucune opération sur la liste compactée cL.
- [0158] Ensuite, le deuxième processus P1 lit la valeur du troisième élément de la liste L concerné par le processus (indice 9) et lit la valeur « 0 ». Le deuxième processus P1 ne fait alors aucune opération sur la liste compactée cL.
- [0159] Puis, le deuxième processus P1 lit la valeur du quatrième élément de la liste L concerné par le processus (indice 13) et lit la valeur « 0 ». Le deuxième processus P1 ne fait alors aucune opération sur la liste compactée cL.
- [0160] Enfin, le deuxième processus P1 lit la valeur du cinquième élément de la liste L

concerné par le processus (indice 17) et lit la valeur « 1 ». Le deuxième processus P1 modifie alors la valeur de l'élément de la liste compactée cL ayant le deuxième indice accessible de la liste compactée cL car c'est la deuxième fois qu'une valeur « 1 » est lue par le deuxième processus P1. Le deuxième indice de la liste compactée cL accessible au deuxième processus P1 est, en l'occurrence, le sixième indice de la liste compactée cL (l'indice 5). Le deuxième processus P1 change alors la valeur initiale « -1 » de l'élément d'indice 5 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 17 ».

- [0161] A l'issue du deuxième processus P1, les valeurs des éléments des autres indices accessibles de la liste compactée cL, en l'occurrence les indices 9, 13 et 17, ne sont pas modifiées et restent donc à la valeur initiale de « -1 ».
- [0162] Comme expliqué précédemment, le troisième processus P2 est illustré sur la [Fig.5]. Le troisième processus P2 vise à effectuer l'opération de remplissage sur les éléments dont l'indice est égal à 2 modulo 4.
- [0163] Le troisième processus P2 lit la valeur du premier élément de la liste L concerné par le processus (indice 2) et lit la valeur « 0 ». Le troisième processus P2 ne fait alors aucune opération sur la liste compactée cL.
- [0164] Le troisième processus P2 lit la valeur du second élément de la liste L concerné par le processus (indice 6) et lit la valeur « 1 ». Le troisième processus P2 modifie alors la valeur de l'élément de la liste compactée cL ayant le premier indice accessible de la liste compactée cL car c'est la première fois qu'une valeur « 1 » est lue par le troisième processus P2. Le premier indice de la liste compactée cL accessible au troisième processus P2 est, en l'occurrence, le troisième indice de la liste compactée cL (l'indice 2). Le troisième processus P2 change alors la valeur initiale « -1 » de l'élément d'indice 2 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 6 ».
- [0165] Le troisième processus P2 lit la valeur du troisième élément de la liste L concerné par le processus (indice 10) et lit la valeur « 0 ». Le troisième processus P2 ne fait alors aucune opération sur la liste compactée cL.
- [0166] Le troisième processus P2 lit la valeur du quatrième élément de la liste L concerné par le processus (indice 14) et lit la valeur « 1 ». Le troisième processus P2 modifie alors la valeur de l'élément de la liste compactée cL ayant le deuxième indice accessible de la liste compactée cL car c'est la deuxième fois qu'une valeur « 1 » est lue par le troisième processus P2. Le deuxième indice de la liste compactée cL accessible au troisième processus P2 est, en l'occurrence, le huitième indice de la liste compactée cL (l'indice 7). Le troisième processus P2 change alors la valeur initiale « -1 » de l'élément d'indice 7 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 14 ».

- [0167] Le troisième processus P2 lit la valeur du cinquième élément de la liste L concerné par le processus (indice 18) et lit la valeur « 0 ». Le troisième processus P2 ne fait alors aucune opération sur la liste compactée cL.
- [0168] A l'issue du troisième processus P2, les valeurs des éléments des autres indices accessibles de la liste compactée cL, en l'occurrence les indices 10, 14 et 18, ne sont pas modifiées et restent donc à la valeur initiale de « -1 ».
- [0169] Comme expliqué précédemment, le quatrième processus P3 est illustré sur la [Fig.6]. Le quatrième processus P3 vise à effectuer l'opération de remplissage sur les éléments dont l'indice est égal à 3 modulo 4.
- [0170] Pour cela, dans l'exemple décrit, le quatrième processus P3 lit la valeur du premier élément de la liste L concerné par le processus (indice 3) et lit la valeur « 1 ». Le quatrième processus P3 modifie alors la valeur de l'élément de la liste compactée cL ayant le premier indice accessible de la liste compactée cL car c'est la première fois qu'une valeur « 1 » est lue par le quatrième processus P3. Le premier indice de la liste compactée cL accessible au quatrième processus P3 est, en l'occurrence, le quatrième indice de la liste compactée cL (l'indice 3). Le quatrième processus P3 change alors la valeur initiale « -1 » de l'élément d'indice 3 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 3 ».
- [0171] Puis, le quatrième processus P3 lit la valeur du second élément de la liste L concerné par le processus (indice 7) et lit la valeur « 1 ». Le quatrième processus P3 modifie alors la valeur de l'élément de la liste compactée cL ayant le deuxième indice accessible de la liste compactée cL car c'est la deuxième fois qu'une valeur « 1 » est lue par le quatrième processus P3. Le deuxième indice de la liste compactée cL accessible au quatrième processus P3 est, en l'occurrence, le huitième indice de la liste compactée cL (l'indice 7). Le quatrième processus P3 change alors la valeur initiale « -1 » de l'élément d'indice 7 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément de la liste L lu, à savoir la valeur « 7 ».
- [0172] Le quatrième processus P3 lit la valeur du troisième élément de la liste L concerné par le processus (indice 11) et lit la valeur « 0 ». Le quatrième processus P3 ne fait alors aucune opération sur la liste compactée cL.
- [0173] Le quatrième processus P3 lit la valeur du quatrième élément de la liste L concerné par le processus (indice 15) et lit la valeur « 0 ». Le quatrième processus P3 ne fait alors aucune opération sur la liste compactée cL.
- [0174] Le quatrième processus P3 lit la valeur du cinquième élément de la liste L concerné par le processus (indice 19) et lit la valeur « 1 ». Le quatrième processus P3 modifie alors la valeur de l'élément de la liste compactée cL ayant le troisième indice accessible de la liste compactée cL car c'est la troisième fois qu'une valeur « 1 » est lue par le quatrième processus P3. Le troisième indice de la liste compactée cL accessible

au quatrième processus P3 est, en l'occurrence, le douzième indice de la liste compactée cL (l'indice 11). Le quatrième processus P3 change alors la valeur initiale « -1 » de l'élément d'indice 19 de la liste compactée cL en la remplaçant par la valeur de l'indice de l'élément L lu, à savoir la valeur « 19 ».

- [0175] A l'issue du quatrième processus P3, les valeurs des éléments des autres indices accessibles de la liste compactée cL, en l'occurrence les indices 15 et 19, ne sont pas modifiées et restent donc à la valeur initiale de « -1 ».
- [0176] La [Fig.7] illustre les résultats des quatre compactages effectués par chacun des processus P0, P1, P2 et P3.
- [0177] Plus précisément, sur cette [Fig.7], la première liste modifiée par le premier processus P0 est représentée à la première ligne ; la deuxième liste modifiée par le deuxième processus P1 est représentée à la deuxième ligne en dessous de la première ligne ; la troisième liste modifiée par le troisième processus P2 est représentée à la troisième ligne en dessous de la deuxième ligne et la quatrième liste modifiée par le quatrième processus P3 est représentée à la quatrième ligne en dessous de la troisième ligne.
- [0178] Dans cette représentation, les éléments concernés par les processus sont représentés en grisé.
- [0179] La liste compactée cL finale correspond à la valeur grisée. Il est ainsi obtenu une liste dans laquelle des éléments d'indice 0 à 7 ont la valeur d'un indice de la liste L ainsi que l'élément d'indice 11 alors que les éléments ayant un indice compris entre 8 et 10 et un indice supérieur ou égal à 12 ont une valeur restée à la valeur initiale.
- [0180] La technique de compactage illustrée à travers les figures 3 à 7 permet de paralléliser la tâche de compactage de la liste L.
- [0181] En effet, les processus P0 à P3 peuvent être conçus comme le passage d'un « peigne » sur la liste L comme schématiquement représenté sur les figures 3 à 6. Autrement formulé, les processus P0 à P3 sont indépendants les uns des autres et peuvent être réalisés simultanément par un cœur 18 différent, ici quatre cœurs 18.
- [0182] Autrement formulé, dans le cas proposé, chaque cœur 18 traite tous les indices ayant le même reste de la division euclidienne par le nombre 4 (en désignant par abus de langage le fait de traiter l'élément ayant l'indice par le fait de traiter l'indice).
- [0183] En l'occurrence, au premier cœur 18 est attribué le reste 0, au deuxième cœur 18 est attribué le reste 1, au troisième cœur 18 est attribué le reste 2 et au quatrième cœur 18 est attribué le reste 3. Ainsi, mathématiquement, cela signifie que le premier cœur 18 traite tous les indices proportionnels à 4, le deuxième cœur 18 traite les indices égaux à 1 modulo 4, le troisième cœur 18 traite les indices égaux à 2 modulo 4 et le quatrième cœur 18 traite les indices égaux à 3 modulo 4.
- [0184] En considérant un nombre quelconque de cœurs 18, cela signifie qu'à chaque

cœur 18 est attribué un reste respectif de la division euclidienne par le nombre de cœurs et que chaque cœur traite l'ensemble des indices dont le reste de la division euclidienne de l'indice par le nombre de cœurs est égal au reste respectif.

- [0185] Le principe d'attribution d'indices spécifiques à chaque cœur 18 qui vient d'être décrit peut être généralisé.
- [0186] En effet, la seule contrainte est que l'ensemble des indices de la liste courante soit attribué à l'un au moins des cœurs 18.
- [0187] Il est à noter qu'un principe d'attribution confiant le même nœud à plusieurs cœurs 18 entraînerait des évaluations redondantes dans la technique AGSI sans que cela ne fausse les calculs. Un tel principe d'attribution, a priori moins favorable, pourrait toutefois être choisi s'il permettait une exécution plus rapide par chaque cœur 18.
- [0188] La technique de compactage peut également être réalisée avec une opération d'initialisation différente. Il convient seulement que la valeur inscrite soit reconnue de façon non équivoque comme non valide lors de la technique AGSI. Autrement formulé, cette valeur ne correspond à aucune valeur possible d'indice d'élément.
- [0189] Par exemple, une autre valeur que la valeur « -1 » peut être utilisée comme valeur d'initialisation au début du processus pour la liste compactée cL.
- [0190] En variante, il est possible de prendre toute valeur négative en valeur d'initialisation pour la liste compactée cL.
- [0191] Selon une autre variante, si le nombre d'éléments au sein du domaine  $\Omega$  est  $N$ , alors il est possible de prendre toute valeur  $I \geq N - 1$  comme valeur d'initialisation.
- [0192] En tenant compte de tous les modes de réalisation précités, la technique de compactage peut être décrite comme suit.
- [0193] La technique de compactage comporte une opération d'initialisation de la liste compactée à une valeur d'initialisation.
- [0194] La technique de compactage comporte également, pour chaque cœur 18, une opération de détermination et une opération de modification éventuelle. Pour chaque cœur 18, l'opération de détermination est mise en œuvre pour plusieurs indices spécifiques au cœur 18 considéré. Pour chaque indice appartenant à ces indices spécifiques, l'opération de détermination consiste à vérifier si l'élément ayant l'indice considéré vérifie une condition binaire.
- [0195] L'opération de détermination permet ainsi de déterminer l'ensemble des éléments dont l'indice appartient aux indices spécifiques et qui vérifie la condition binaire.
- [0196] L'opération de modification éventuelle est uniquement mise en œuvre pour les éléments déterminés à l'opération de détermination.
- [0197] Autrement formulé, l'opération de modification est une modification d'une valeur de la liste compactée uniquement s'il est déterminé que la condition binaire est vérifiée.
- [0198] La valeur modifiée est celle correspondant à l'indice le plus petit de l'ensemble des

indices spécifiques au cœur et n'ayant pas subi de modifications.

[0199] D'un point de vue mathématique, on peut écrire le pseudocode qui suit :

[0200] initialisation : pour chaque élément  $x \in \llbracket 0, N - 1 \rrbracket : cL(x) = -1$

[0201] compactage : Pour chaque processus  $p \in \llbracket 0, N_L - 1 \rrbracket :$

[0202]  $x = p$

[0203]  $i_W = p$

[0204] –  $Na = 0$

– Tant que  $x < N :$

• si  $L(x) = VRAI :$

[0205]  $cL(i_W) = x$

[0206]  $i_W \leftarrow i_W + N_L$

[0207] Le procédé de compactage de liste choisi par le demandeur présente la particularité de laisser des trous au sein de la liste compactée. On voit sur la [Fig.7] trois valeurs non valides (-1) encadrées. A titre de comparaison, la méthode proposée par (A massively parallel Eikonal solver on unstructured meshes, D. Ganellari, G. Haase, 2018) produit une liste sans trou. La présence de trou dans la liste compactée cL se traduit par une occupation a priori moins optimale des processeurs 18 (certains étant affectés à des tâches vides). Toutefois, la méthode de compactage du demandeur présente un temps d'exécution plus réduite que celle évoquée par D. Ganellari et al. Il s'en suit que la performance de l'ensemble est avantageuse.

[0208] Il s'en suit également qu'une information supplémentaire, notée Na, doit être collectée pendant ou après la phase de compactage de liste. Na désigne le plus grand indice de cL contenant une valeur d'indice valide.

[0209] Sur l'exemple rapporté en [Fig.7],  $Na = 10$ , dit autrement, tout indice de cL de valeur supérieure à 10 contient -1.

[0210] Le procédé de détermination correspond à l'utilisation de la technique de compactage dans la technique AGSI.

[0211] La technique AGSI peut être décrite comme la mise en œuvre de deux phases :

[0212] – une phase de réception des données d'entrées suivantes:

- Un domaine  $\Omega$  contenant un ensemble de points  $x$  associés chacun à un indice unique  $i$
- Pour tout point  $x$  du domaine  $\Omega$ , les ensembles  $V^+(x)$  et  $V^-(x)$  appelés stencils directs et indirects, ainsi que la formule d'Hopf-Lax  $F(x, V^+(x))$  en ce point
- Pour tout point  $x$  d'un ensemble  $D$ , la valeur  $U(x)$  en ce point
- Une valeur de seuil  $\varepsilon$

– une phase de détermination de la grandeur physique  $U(x)$  en tout point de  $\Omega_D$

par utilisation d'itérations AGSI successives, chaque itération comprenant la mise en œuvre de calculs uniquement sur des points contenus dans une liste L et l'actualisation de la liste en fonction des calculs mis en œuvre, les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.

- [0213] Aussi, le procédé de détermination comporte une phase de réception et une phase de détermination.
- [0214] Chaque itération comporte une étape de transformation, une étape de lecture, une étape de mise en œuvre et une étape d'actualisation.
- [0215] L'étape de transformation transforme une liste courante L en une liste compactée courante cL.
- [0216] La liste courante comporte des points de l'environnement prédéfini auxquels sont attribués un nombre.
- [0217] Chaque point est repéré par un indice.
- [0218] Des indices spécifiques sont attribués à chaque cœur 18 de la pluralité de cœurs 18, l'ensemble des indices de la liste courante étant attribué à l'un des cœurs 18.
- [0219] La transformation comporte l'initialisation de la liste compactée à une valeur d'initialisation.
- [0220] La transformation comporte également pour chaque cœur 18 de la pluralité de cœurs 18 la détermination pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur 18 considéré si une condition binaire est vérifiée, pour obtenir un élément déterminé lorsque la condition binaire est vérifiée.
- [0221] La transformation comprend aussi, pour chaque point de la liste courante déterminé, la modification d'une valeur de la liste compactée n'ayant pas été modifiée et présentant l'indice le plus petit de l'ensemble des indices spécifiques au cœur 18 considéré.
- [0222] La lecture est une lecture des valeurs de la liste compactée courante différentes de la valeur d'initialisation pour obtenir les points vérifiant la condition binaire.
- [0223] La mise en œuvre est une étape de mise en œuvre de calculs uniquement sur les points déterminés.
- [0224] Les calculs comportent une opération de test de convergence de la valeur du point, une opération d'élimination du point ayant fait l'objet de l'opération de test de la liste courante, et une opération d'ajout de points uniquement si le test de convergence n'est pas satisfait, les points ajoutés étant de préférence les points du stencil négatif.
- [0225] L'étape d'actualisation est une étape d'actualisation de la liste en fonction des calculs mis en œuvre.
- [0226] Les itérations sont mises en œuvre jusqu'à ce que la liste courante soit vide.
- [0227] Ce qui vient d'être décrit pour le procédé de détermination peut être décrit de manière équivalente comme suit.

- [0228] Le domaine  $\Omega$  peut-être découpé en sous-domaines, de telle façon que chaque élément  $x$  du domaine  $\Omega$  appartient à un sous-domaine et un seul. Les listes  $L$  et  $cL$  ne désignent alors plus les indices des éléments  $x$ , mais les indices des sous-domaines auxquels ils sont associés.
- [0229] Le découpage de  $\Omega$  en  $N$  sous-domaines accroît l'efficacité de résolution par une architecture parallèle car il permet de regrouper astucieusement les nœuds de façon à optimiser les accès à la mémoire centrale. Par exemple, il peut être avantageusement envisagé une stratégie de définition de sous-domaines au sein desquels tous les nœuds partagent certaines valeurs de paramètres, ce qui permet de les charger une fois pour toute en mémoire locale et de les utiliser ensuite pour chacun des nœuds du sous-domaine. Une telle stratégie est a priori plus favorable puisqu'elle repose sur l'emploi de mémoire locale et minimise les accès à la mémoire centrale.
- [0230] Par ailleurs, le découpage de  $\Omega$  en  $N$  sous-domaines permet également de découpler la taille de la liste à compacter à chaque itération du nombre total d'éléments  $x$  du domaine  $\Omega$  et donc de compacter des listes plus courtes. Par exemple, en groupant les nœuds par sous-domaines de taille 64, une liste 64 fois moins grande à chaque itération est compactée.
- [0231] Le domaine  $\Omega$  contient  $N_n$  points et est découpé en  $N$  sous-domaines.
- [0232] Les notations qui suivent sont utilisées dans la série d'instructions qui décrit le procédé :
- [0233] –  $SOUS\_DOMAINE(x)$  est une fonction retournant l'indice du sous-domaine auquel appartient le nœud  $x$ . Cette fonction peut être implémentée de diverses façons, comme par exemple une table calculée à l'avance.
- $F(x, V^+(x))$  est la fonction d'Hopf-Lax utilisée
- $L_1$  et  $L_2$  sont deux tableaux de booléens de taille  $N$  appelés « liste courante des sous-domaines actifs » et « liste suivante des sous-domaines actifs »
- $U_1(x)$  et  $U_2(x)$  sont deux tableaux de taille  $N_n$  appelés « distances courantes des sous-domaines actifs » et « distances suivantes des sous-domaines actifs »
- $cL$  est un tableau d'entiers de taille  $N$  appelé « liste compactée des sous-domaines actifs »
- $N_{an}$  est le plus grand indice de  $cL$  contenant une valeur valide
- [0234] Avec ces notations, le procédé s'écrit ainsi :
- [0235] Boucle principale :
- [0236] AMELIORATION = VRAI
- [0237] Tant que AMELIORATION = VRAI :
- [0238] – AMELIORATION = FAUX
- [0239]  $cL \leftarrow COMPACT(L)$
- [0240]  $N_{an} \leftarrow RECUP\_Na(cL)$

- [0241] – Lancer en parallèle  $SOLVER(a, cL)$  pour  $a = 0 \rightarrow N_{an} - 1$
- [0242]  $U_2 = U_1$
- [0243]  $L_2 = L_1$
- [0244] RECUP\_NA(cL) :
- [0245] Cette fonction peut être réalisée en lançant en parallèle N processus  
 $p \in \llbracket 0, N - 1 \rrbracket$  réalisant chacun :
- [0246] – Si  $cL(p) \neq -1$  :  $ATOMIC\_MAX(N_{an}, p)$
- [0247] Cette fonction peut également être réalisée pendant la fonction COMPACTER. Le pseudo code de COMPACTER est alors modifié comme suit (en gras):
- [0248] D'un point de vue mathématique, on peut écrire le pseudocode qui suit :
- [0249] initialisation : pour chaque élément  $x \in \llbracket 0, N - 1 \rrbracket$  :  $cL(x) = -1$
- [0250] compactage : Pour chaque processus  $p \in \llbracket 0, N_L - 1 \rrbracket$  :
- [0251]  $x = p$
- [0252]  $i_w = p$
- [0253] – **Na = 0**  
– Tant que  $x < N$  :  
• si  $L(x) = VRAI$  :
- [0254]  $cL(i_w) = x$
- [0255] • **Na = max(Na,  $i_w$ )**
- [0256]  $i_w \leftarrow i_w + N_L$
- [0257] COMPACTER(L):
- [0258] Compacter L suivant la méthode détaillée précédemment.
- [0259] SOLVE(a, cL) :
- [0260]  $s = cL(a)$
- [0261] Si s est non valide (ici: si  $s = -1$ ), quitter.
- [0262] Pour tout nœud  $p$  appartenant au sous-domaine s :
- [0263]  $d_{CURR} = U1(CURR, p)$
- [0264]  $d_{NEXT} = F(p, V^+(p))$
- [0265] Si  $d_{NEXT} < d_{CURR}$  et si  $|d_{CURR} - d_{NEXT}| < \varepsilon$  :
- [0266] AMELIORATION = VRAI
- [0267] Pour tout voisin  $x_{RV}$  de  $V(p)$  :
- [0268]  $s_{RV} = SOUS\_DOMAINE(x_{RV})$
- [0269]  $L_2(s_{RV}) = VRAI$
- [0270]  $L_1(s) = FAUX$
- [0271] Un tel procédé permet de résoudre l'équation avec efficacité pour obtenir la grandeur respectant les contraintes tout en permettant une parallélisation dans l'implémentation informatique du procédé.

- [0272] Le procédé de compactage bénéficie de l'avantage de sa simplicité et de sa rapidité d'exécution, qui compense la présence de trous, et donc d'une occupation parfois sous optimale de processeurs 18.
- [0273] Ainsi, le procédé de détermination d'une grandeur physique bénéficie de l'avantage de la généralité de la technique AGSI et de sa simplicité de mise en œuvre et compense sa complexité algorithmique par l'emploi d'une technique de compactage permettant une mise en œuvre parallèle.
- [0274] Le procédé peut être utilisé pour d'autres techniques itératives utilisant une liste évolutive comme la technique AGSI. Par exemple, il pourrait être envisagé d'utiliser une technique appelée « Hamilton Fast Marching Dubins ».
- [0275] Le procédé vient d'être décrit pour la détermination de trajectoires mais pourrait également être utilisé pour le domaine des images.
- [0276] Par exemple, le système de détermination 10 délimite l'artère (segmentation d'image) ou détermine le débit dans une artère (déduction d'une mesure physique sur la base d'images).
- [0277] Dans chacun des cas qui viennent d'être décrits, il apparaît bien que le procédé présenté comporte deux phases : une phase de réception de données d'entrées et une phase de détermination de la grandeur physique par utilisation d'une technique de résolution du système d'équations différentielles, pour obtenir la grandeur physique respectant la contrainte.
- [0278] Les données d'entrées comprennent des mesures de la contrainte en plusieurs points de l'environnement prédéfini.
- [0279] La technique de résolution est une technique itérative avec des itérations d'une opération de transformation, d'initialisation, de détermination, de modification, de lecture, de mise en œuvre et d'actualisation. Les itérations sont mises en œuvre jusqu'à ce que la liste courante soit vide.
- [0280] L'opération de transformation est une opération de transformation d'une liste courante en une liste compactée courante, la liste courante comportant des points de l'environnement prédéfini auxquels sont attribués un nombre, chaque point étant repéré par un indice, des indices spécifiques étant attribués à chaque cœur 18, l'ensemble des indices de la liste courante étant attribué à l'un des cœurs 18.
- [0281] La transformation comprend l'opération d'initialisation qui correspond à l'initialisation de la liste compactée à une valeur d'initialisation.
- [0282] Par ailleurs, les opérations de détermination et de modification d'une valeur de la liste compactée sont mises en œuvre pour chaque cœur 18 durant la transformation.
- [0283] En particulier, la détermination cherche à savoir si pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur 18 considéré, la condition binaire précitée est vérifiée.

- [0284] Ensuite, la valeur de la liste compactée n'ayant pas été modifiée et ayant l'indice le plus petit est modifiée pour chaque point de la liste courante déterminé.
- [0285] Lors de l'opération de lecture, les valeurs de la liste compactée courante différentes de la valeur d'initialisation sont lues pour obtenir les points vérifiant la condition binaire.
- [0286] Ensuite, il est mis en œuvre les calculs uniquement sur les points déterminés et la liste est actualisée en fonction des calculs mis en œuvre.
- [0287] Un tel procédé permet donc bien de gagner en rapidité sans perte en précision.

## Revendications

[Revendication 1]

Procédé de détermination d'une grandeur physique, la grandeur physique vérifiant un système d'équations différentielles d'Hamilton-Jacobi pour respecter une contrainte variant spatialement dans un environnement prédéfini, le procédé étant mis en œuvre par un système de détermination (10) de la grandeur physique, le système de détermination (10) comprenant une pluralité de cœurs (18), le procédé comportant :

- une phase de réception de données d'entrées, les données d'entrées comprenant des mesures de la contrainte en plusieurs points de l'environnement prédéfini,
- une phase de détermination de la grandeur physique par utilisation d'une technique de résolution du système d'équations différentielles, pour obtenir la grandeur physique respectant la contrainte, la technique de résolution comportant plusieurs itérations, chaque itération comportant les étapes suivantes:
  - la transformation d'une liste courante en une liste compactée courante, la liste courante comportant des points de l'environnement prédéfini auxquels sont attribués un nombre, chaque point étant repéré par un indice, des indices spécifiques étant attribués à chaque cœur (18) de la pluralité de cœurs (18), l'ensemble des indices de la liste courante étant attribué à l'un des cœurs (18), la transformation comprenant :
    - l'initialisation de la liste compactée à une valeur d'initialisation,
    - pour chaque cœur (18) de la pluralité de cœurs (18) :
      - la détermination pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur (18) considéré si une condition binaire est vérifiée, pour obtenir un élément déterminé lorsque la condition binaire est vérifiée, et
      - pour chaque point de la liste courante déterminé, la modification d'une valeur de la liste compactée n'ayant pas été modifiée et présentant l'indice le plus petit de l'ensemble des indices spécifiques au cœur (18) considéré,
    - la lecture des valeurs de la liste compactée courante différentes de la valeur d'initialisation pour obtenir les points vérifiant la condition binaire,
    - la mise en œuvre de calculs uniquement sur les points déterminés, et
    - l'actualisation de la liste en fonction des calculs mis en œuvre,

les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.

- [Revendication 2] Procédé selon la revendication 1, dans lequel le nombre d'indices spécifiques est le même pour chaque cœur (18).
- [Revendication 3] Procédé selon la revendication 1 ou 2, dans lequel, à chaque cœur (18) est attribué un reste respectif de la division euclidienne par le nombre de cœurs (18), l'ensemble des indices spécifiques au cœur (18) considéré comprenant l'ensemble des indices dont le reste de la division euclidienne de l'indice par le nombre de cœurs (18) est égal au reste respectif.
- [Revendication 4] Procédé selon l'une quelconque des revendications 1 à 3, dans lequel chaque nombre attribué est un nombre binaire et la condition binaire est que le nombre binaire attribué est égal à une valeur prédéfinie.
- [Revendication 5] Procédé selon l'une quelconque des revendications 1 à 4, dans lequel la valeur d'initialisation est une valeur négative.
- [Revendication 6] Procédé selon l'une quelconque des revendications 1 à 5, dans lequel l'opération d'initialisation est également mise en œuvre par chaque cœur (18) uniquement pour les indices spécifiques au cœur (18) considéré.
- [Revendication 7] Procédé selon l'une quelconque des revendications 1 à 6, dans lequel la mise en œuvre itérative de calculs est réalisée jusqu'à ce que la liste courante soit vide.
- [Revendication 8] Procédé selon l'une quelconque des revendications 1 à 7, dans lequel, les calculs comportent une opération de test de convergence de la valeur du point.
- [Revendication 9] Procédé selon la revendication 8, dans lequel, pour chaque point, il est défini un stencil négatif, les calculs comportent également :
  - une opération d'élimination du point ayant fait l'objet de l'opération de test de la liste courante, et
  - une opération d'ajout de points uniquement si le test de convergence n'est pas satisfait, les points ajoutés étant de préférence les points du stencil négatif.
- [Revendication 10] Procédé selon l'une quelconque des revendications 1 à 9, dans lequel les contraintes sont des mesures d'au moins un capteur.
- [Revendication 11] Système de détermination (10) d'une grandeur physique, la grandeur physique vérifiant un système d'équations différentielles d'Hamilton-Jacobi pour respecter une contrainte variant spatialement dans un environnement prédéfini, le système de détermination (10)

comprenant une pluralité de cœurs (18), le système de détermination (18) étant propre à mettre en œuvre un procédé de détermination d'une grandeur physique, le procédé comportant :

- une phase de réception de données d'entrées, les données d'entrées comprenant des mesures de la contrainte en plusieurs points de l'environnement prédéfini,

- une phase de détermination de la grandeur physique par utilisation d'une technique de résolution du système d'équations différentielles, pour obtenir la grandeur physique respectant la contrainte, la technique de résolution comportant plusieurs itérations, chaque itération comportant les étapes suivantes:

- la transformation d'une liste courante en une liste compactée courante, la liste courante comportant des points de l'environnement prédéfini auxquels sont attribués un nombre, chaque point étant repéré par un indice, des indices spécifiques étant attribués à chaque cœur (18) de la pluralité de cœurs (18), l'ensemble des indices de la liste courante étant attribué à l'un des cœurs (18), la transformation comprenant :

- l'initialisation de la liste compactée à une valeur d'initialisation,

- pour chaque cœur (18) de la pluralité de cœurs (18) :

- la détermination pour chaque point de la liste courante présentant un indice compris parmi des indices spécifiques au cœur (18) considéré si une condition binaire est vérifiée, pour obtenir un élément déterminé lorsque la condition binaire est vérifiée, et

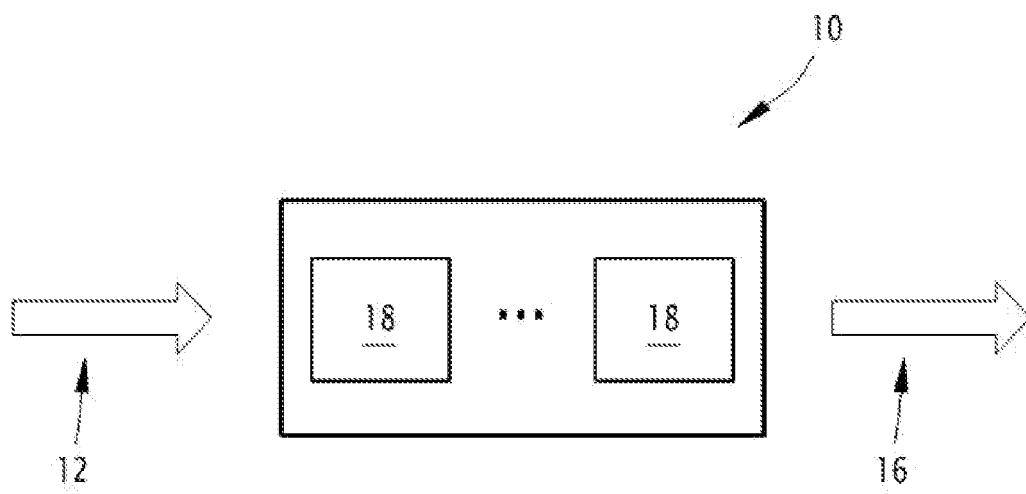
- pour chaque point de la liste courante déterminé, la modification d'une valeur de la liste compactée n'ayant pas été modifiée et présentant l'indice le plus petit de l'ensemble des indices spécifiques au cœur (18) considéré,

- la lecture des valeurs de la liste compactée courante différentes de la valeur d'initialisation pour obtenir les points vérifiant la condition binaire,

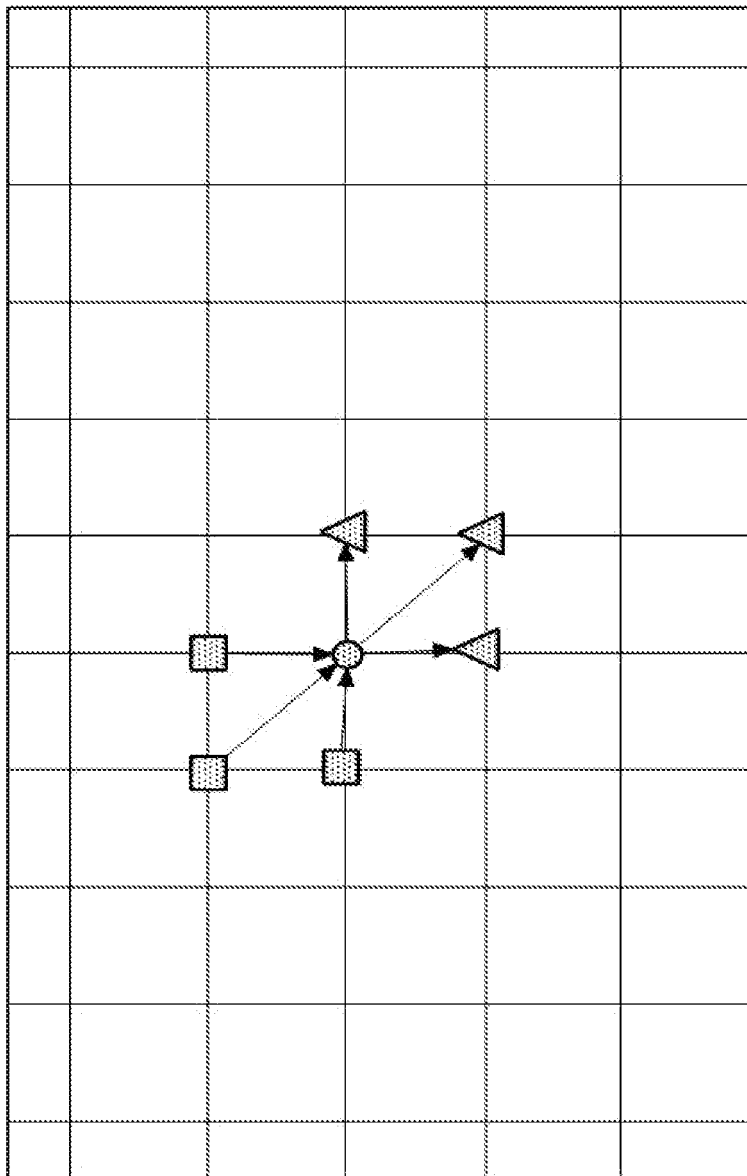
- la mise en œuvre de calculs uniquement sur les points déterminés, et

- l'actualisation de la liste en fonction des calculs mis en œuvre, et les itérations étant mises en œuvre jusqu'à ce que la liste courante soit vide.

[Fig. 1]

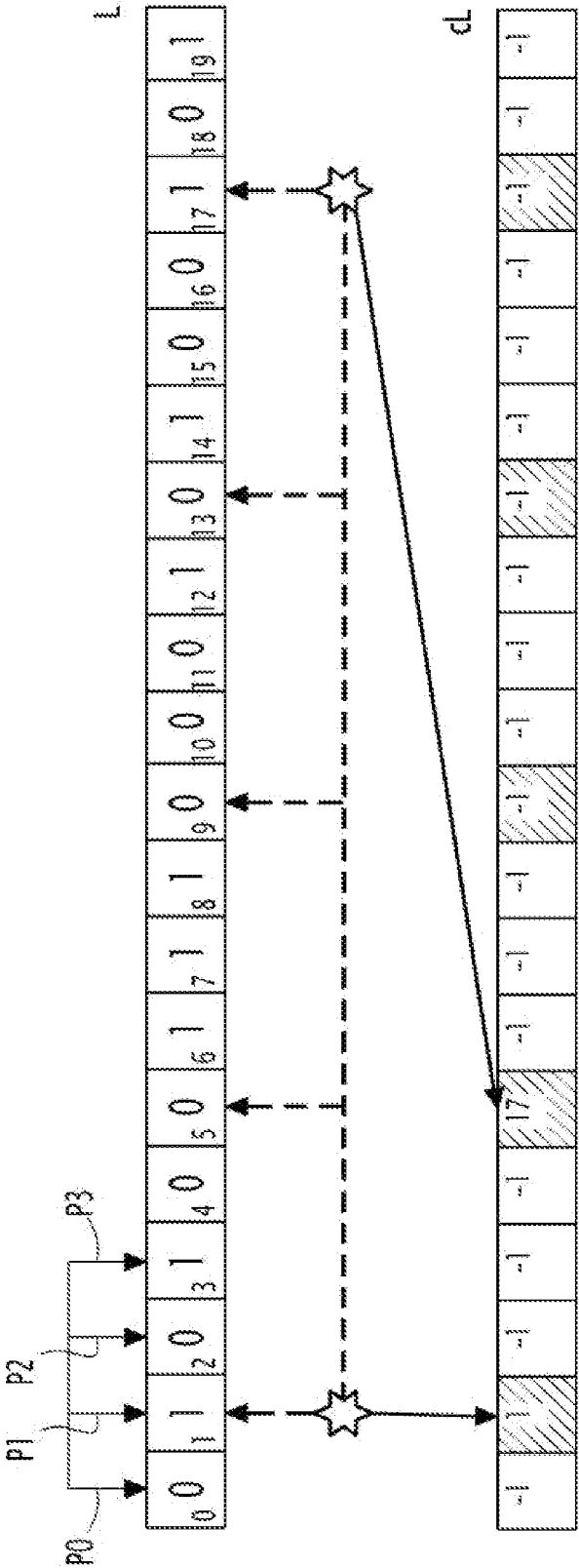


[Fig. 2]

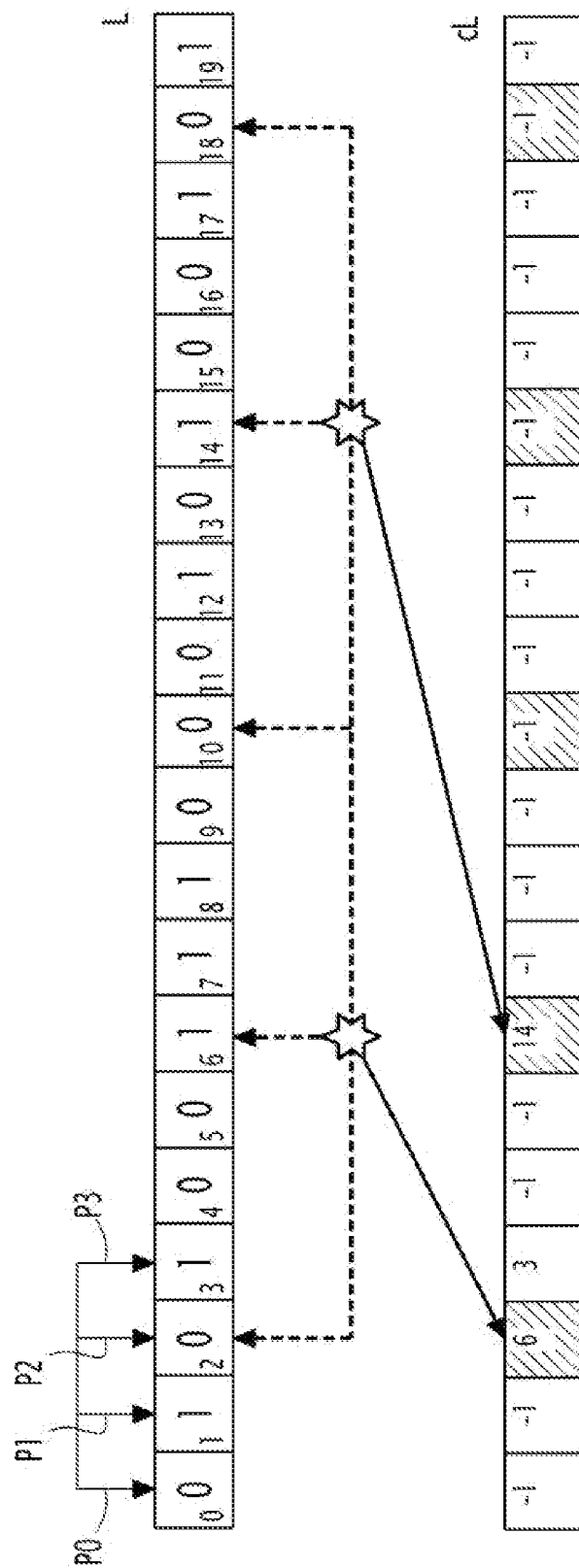




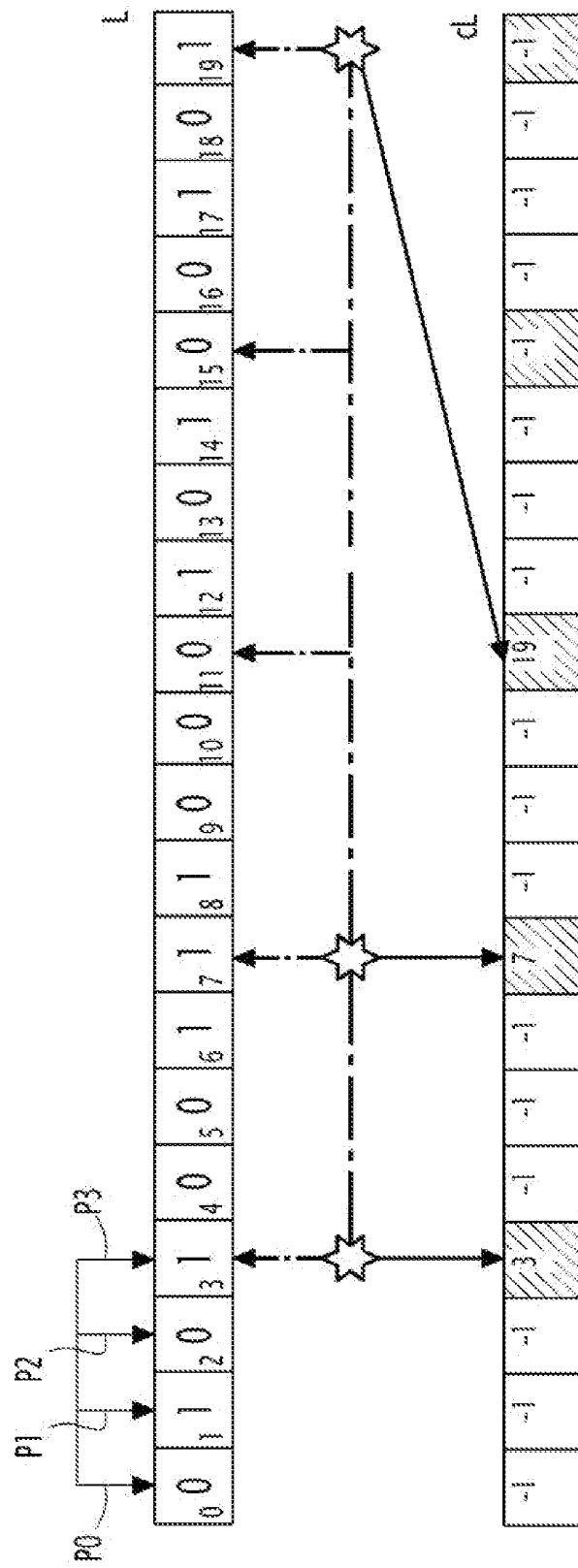
[Fig. 4]



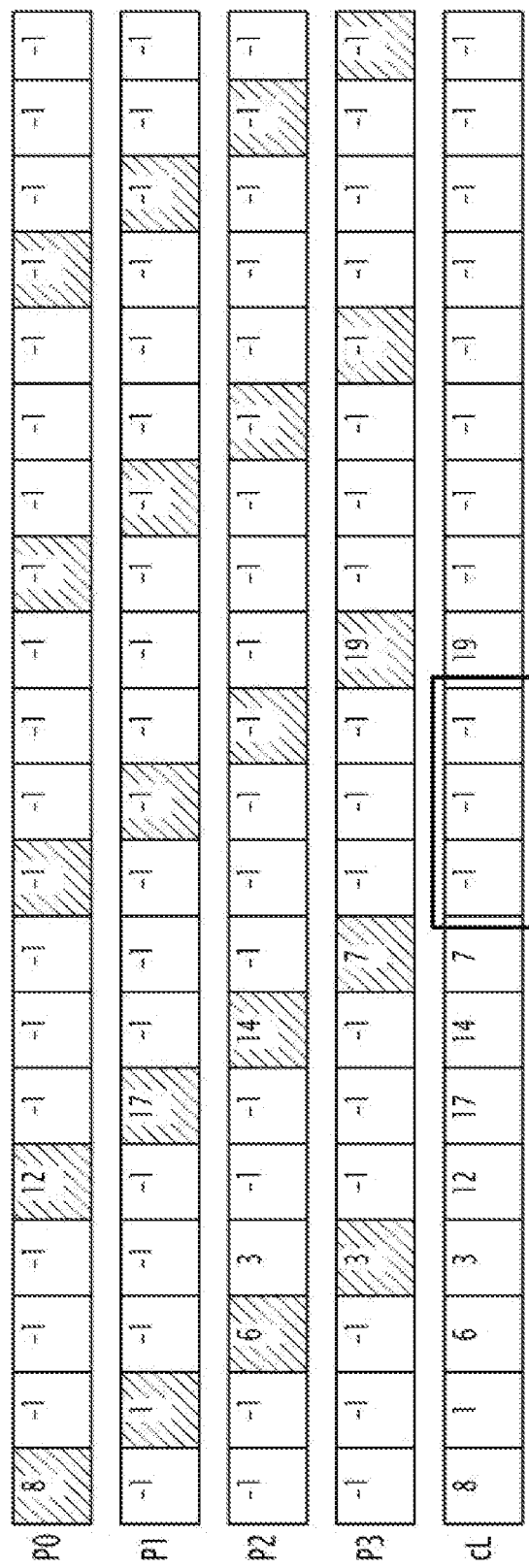
[Fig. 5]



[Fig. 6]



[Fig. 7]



## RAPPORT DE RECHERCHE PRÉLIMINAIRE

établi sur la base des dernières revendications  
déposées avant le commencement de la recherche

N° d'enregistrement  
national

FA 904181  
FR 2114111

| DOCUMENTS CONSIDÉRÉS COMME PERTINENTS                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Revendication(s)<br>concernée(s)                                                                                                                                                                                                                                                                                                                             | Classement attribué<br>à l'invention par l'INPI             |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|
| Catégorie                                                                                                                                                                                                                                      | Citation du document avec indication, en cas de besoin,<br>des parties pertinentes                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                              |                                                             |
| X                                                                                                                                                                                                                                              | <p>GILLBERG TOR ET AL: "Parallel solutions of static Hamilton-Jacobi equations for simulations of geological folds", JOURNAL OF MATHEMATICS IN INDUSTRY, [Online]<br/>vol. 4, no. 1,<br/>31 décembre 2014 (2014-12-31), page 10, XP055957633,<br/>ISSN: 2190-5983, DOI: 10.1186/2190-5983-4-10<br/>Extrait de l'Internet:<br/>URL:https://mathematicsinindustry.springeropen.com/track/pdf/10.1186/2190-5983-4-10.pdf&gt;<br/>[extrait le 2022-09-07]<br/>* Appendix 2;<br/>pages 1, 6 - page 10 *</p> | 1-11                                                                                                                                                                                                                                                                                                                                                         | <p>G06F17/13<br/>G06F7/544<br/>G06F7/38</p>                 |
| A                                                                                                                                                                                                                                              | <p>US 2016/343105 A1 (NOACK MARCUS [NO])<br/>24 novembre 2016 (2016-11-24)<br/>* figures 5-9 *</p>                                                                                                                                                                                                                                                                                                                                                                                                     | 1-11                                                                                                                                                                                                                                                                                                                                                         | <p>DOMAINES TECHNIQUES<br/>RECHERCHÉS (IPC)</p> <p>G06F</p> |
| Date d'achèvement de la recherche                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Examineur                                                                                                                                                                                                                                                                                                                                                    |                                                             |
| 7 septembre 2022                                                                                                                                                                                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Virnik, Elena                                                                                                                                                                                                                                                                                                                                                |                                                             |
| CATÉGORIE DES DOCUMENTS CITÉS                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <p>T : théorie ou principe à la base de l'invention<br/>E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.<br/>D : cité dans la demande<br/>L : cité pour d'autres raisons<br/>.....<br/>&amp; : membre de la même famille, document correspondant</p> |                                                             |
| <p>X : particulièrement pertinent à lui seul<br/>Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie<br/>A : arrière-plan technologique<br/>O : divulgation non-écrite<br/>P : document intercalaire</p> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                                                              |                                                             |

## ANNEXE AU RAPPORT DE RECHERCHE PRÉLIMINAIRE

FR 2114111 FA 904181

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche préliminaire visé ci-dessus.

Les dits membres sont contenus au fichier informatique de l'Office européen des brevets à la date du **07-09-2022**

Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets, ni de l'Administration française

Document brevet cité  
au rapport de recherche

Date de  
publication

Membre(s) de la  
famille de brevet(s)

Date de publication

US 2016343105      A1      24-11-2016      AUCUN

Pour tout renseignement concernant cette annexe : voir Journal Officiel de l'Office européen des brevets, No.12/82