



(12) **DEMANDE DE BREVET CANADIEN  
CANADIAN PATENT APPLICATION**

(13) **A1**

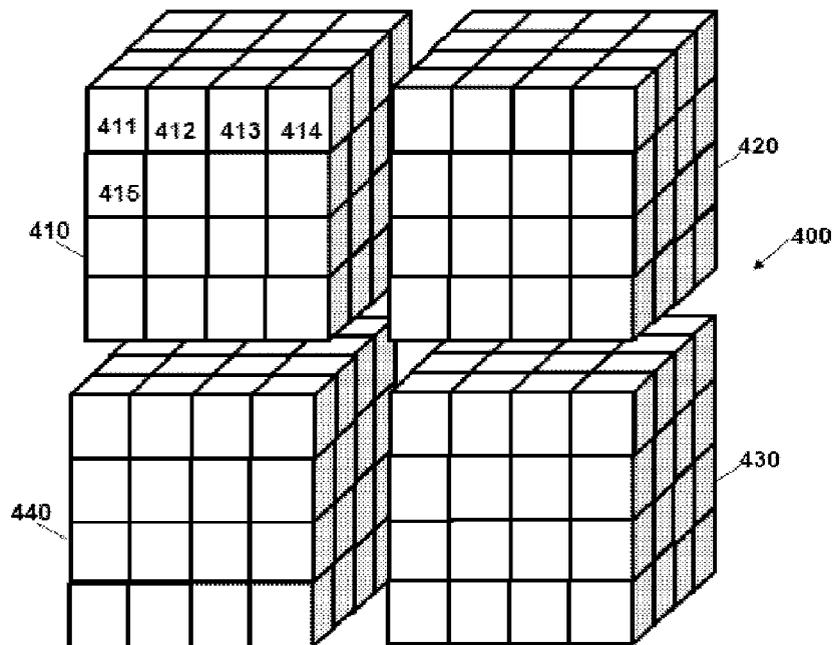
(86) **Date de dépôt PCT/PCT Filing Date:** 2022/05/13  
(87) **Date publication PCT/PCT Publication Date:** 2022/11/24  
(85) **Entrée phase nationale/National Entry:** 2023/11/20  
(86) **N° demande PCT/PCT Application No.:** FR 2022/050913  
(87) **N° publication PCT/PCT Publication No.:** 2022/243626  
(30) **Priorité/Priority:** 2021/05/21 (FR FR2105352)

(51) **Cl.Int./Int.Cl. G06T 17/00** (2006.01),  
**G06T 17/20** (2006.01), **G06T 19/20** (2011.01)  
(71) **Demandeur/Applicant:**  
NOVAQUARK, FR  
(72) **Inventeurs/Inventors:**  
GRIS, GUILLAUME, FR;  
TRETTNER, PHILIP ULRICH, DE  
(74) **Agent:** ROBIC

(54) **Titre : GENERATION D'UN MAILLAGE SURFACIQUE A PARTIR D'UNE MODELISATION EN VOXELS D'UN ENVIRONNEMENT TRIDIMENSIONNEL**

(54) **Title: GENERATION OF AN AREAL MESH FROM A VOXEL MODEL OF A THREE-DIMENSIONAL ENVIRONMENT**

[Fig. 5]



(57) **Abrégé/Abstract:**

Procédé pour la génération d'une vue d'un environnement tridimensionnel sur un ensemble de terminaux de communication à partir d'une modélisation de l'environnement stockée sur un serveur sous la forme d'une grille de voxels, ce serveur mettant en oeuvre des étapes de - partition (S1) de la grille en un ensemble de sous-grilles; - génération (S2) d'un maillage surfacique pour chaque sous-grille - pour chaque sous-grille, simplification (S3) d'une sous partie du maillage excluant une zone de bordure du maillage; et création d'une table associant chaque vertex du bord à un voxel; - assemblage (S4) des maillages des sous-grilles, à l'aide de cette table, afin de former un maillage surfacique global; - transmission (S5) du maillage surfacique global vers les terminaux de communication, et génération (S6) d'une vue de l'environnement tridimensionnel en fonction du maillage surfacique global et d'un point de vue associé à chacun des terminaux de communication.

**Date de soumission :** 2023/11/20

**No de la demande can. :** 3219703

**Abrégé:**

Procédé pour la génération d'une vue d'un environnement tridimensionnel sur un ensemble de terminaux de communication à partir d'une modélisation de l'environnement stockée sur un serveur sous la forme d'une grille de voxels, ce serveur mettant en œuvre des étapes de - partition (S1) de la grille en un ensemble de sous-grilles; - génération (S2) d'un maillage surfacique pour chaque sous-grille - pour chaque sous-grille, simplification (S3) d'une sous partie du maillage excluant une zone de bordure du maillage; et création d'une table associant chaque vertex du bord à un voxel; - assemblage (S4) des maillages des sous-grilles, à l'aide de cette table, afin de former un maillage surfacique global; - transmission (S5) du maillage surfacique global vers les terminaux de communication, et génération (S6) d'une vue de l'environnement tridimensionnel en fonction du maillage surfacique global et d'un point de vue associé à chacun des terminaux de communication.

# Génération d'un maillage surfacique à partir d'une modélisation en voxels d'un environnement tridimensionnel

## [0001] DOMAINE DE L'INVENTION

[0002] La présente invention est relative à la génération, par un ensemble de terminaux, d'une vue d'un environnement tridimensionnel géré par un même serveur informatique. Elle s'applique notamment au domaine des univers virtuels multi-utilisateurs, notamment de type « jeux vidéo ».

## [0003] CONTEXTE DE L'INVENTION

[0004] Le domaine des environnements tridimensionnels virtuels est en plein essor. Ce type d'environnement permet à un grand nombre d'utilisateurs d'interagir sur un l'environnement en en modifiant certaines caractéristiques et en y naviguant de la façon la plus réaliste possible.

[0005] Les déplacements d'un utilisateur impactent son point et son champ de vue, de sorte que l'affichage de l'environnement tridimensionnel doit être modifié constamment. En outre, le déplacement d'un utilisateur peut être visible d'autres utilisateurs, ce qui impacte également la propre vue que ceux-ci doivent avoir du monde.

[0006] Par ailleurs, certains environnements tridimensionnels permettent des modifications par les utilisateurs : ceux-ci peuvent impacter l'environnement en creusant des objets existants, par exemple, ou en créant des objets nouveaux. Ces modifications également doit être répercutées sur la vue qu'ont les autres utilisateurs de l'environnement tridimensionnel.

[0007] Parmi les différents environnements tridimensionnels existant, on peut notamment citer celui mis en œuvre par les Demandeurs et nommé « Dual Universe ». *Dual Universe* est un jeu vidéo de simulation spatiale massivement multijoueur. Le jeu se déroule dans un univers « unique, continu, et non divisé » et se joue à la première personne.

[0008] Des explications sur l'environnement mis en œuvre par le jeu peuvent être trouvées sur le site web : <https://www.dualuniverse.game>, ou sur la page Wikipédia

qui, elle-même, fournit de nombreuses références documentaires :  
[https://en.wikipedia.org/wiki/Dual\\_Universe](https://en.wikipedia.org/wiki/Dual_Universe)

[0009] Bien évidemment, d'autres environnements tridimensionnels sont disponibles, comme notamment ceux mis en œuvre dans le jeu « Minecraft ».

[0010] Il existe différentes façons de modéliser un environnement tridimensionnel. Une possibilité est de discrétiser l'environnement en une matrice, ou « grille », de voxels, de la même façon qu'une image numérique en deux dimensions peut être discrétisée un tableau de pixels. L'environnement « *Dual Universe* », notamment, fonctionne sur la base d'une telle modélisation.

[0011] Afin de pouvoir visualiser l'environnement tridimensionnel, des données sur cet environnement doivent être transmises aux terminaux de communication associés aux utilisateurs. Du fait des interactions récurrentes des utilisateurs, ces données doivent être transmises, au moins en partie, de façon très régulière afin de « rafraîchir » la vue que les utilisateurs ont de l'environnement tridimensionnel.

[0012] Or, une modélisation en voxels d'un environnement de grande taille tel qu'un univers utilisé dans un jeu immersif comme *Dual Universe*, Minecraft, etc. implique un nombre extrêmement important de données. Le volume de données dépend directement de la taille de l'environnement et de la discrétisation utilisée.

[0013] Dès lors, le volume à transférer depuis le serveur qui gère l'environnement tridimensionnel et les différents terminaux de communication est également très important et peut impliquer de très fortes contraintes à la fois sur le réseau de communication et sur les ressources computationnelles mettant en œuvre le serveur et/ou les terminaux de communication.

[0014] Toutefois, dès lors qu'on opte pour ne pas limiter (drastiquement) la taille de l'environnement géré par un unique serveur, deux voies semblent possibles.

[0015] Il est tout d'abord possible de transmettre la modélisation en voxels aux utilisateurs. Toutefois une telle façon de faire peut provoquer un engorgement du réseau de communication, nuisant à la qualité de service perçu par les utilisateurs, notamment ceux ne disposant pas d'une connexion à très haut débit. En outre, elle nécessite que les terminaux de communication des utilisateurs soient suffisamment puissants pour effectuer les conversions de la modélisation en voxels en une structure de données permettant son rendu sur un écran ou toute autre interface homme-

machine (masque à réalité virtuelle, etc.). Également, un hébergeur fait typiquement payer l'utilisation de bande passante sortante, et cette façon de faire peut donc avoir un impact économique important.

[0016] Une autre voie consiste à effectuer cette conversion sur le serveur et de transmettre les données à visualiser à chaque utilisateur. Ce mécanisme se heurte toutefois au fait que le coût des calculs, côté serveur, est proportionnel au nombre d'utilisateurs et que les données à calculer dépendent du point de vue des utilisateurs dans l'environnement et sont donc différentes pour chaque utilisateur. Dès lors, le serveur nécessite d'importantes ressources computationnelles, ce qui augmente son coût, ce qui peut devenir rédhibitoire pour l'exploitant d'un tel environnement tridimensionnel.

[0017] Une proposition adoptée par certains systèmes consiste alors à limiter la taille des univers et à associer un serveur distinct à chaque univers, sur lequel ne peuvent interagir qu'un nombre également limité d'utilisateurs. Toutefois, cette proposition forme une limitation préjudiciable aux possibilités ludiques et immersives que peuvent offrir ces environnements tridimensionnels.

#### [0018] RESUME DE L'INVENTION

[0019] Un objectif de la présente invention est de fournir une solution palliant au moins partiellement les inconvénients précités. Il s'agit notamment de permettre la visualisation d'un unique environnement tridimensionnel par une multitude de terminaux mobiles, en minimisant les ressources nécessaires (transmission et calcul), sans impacter la qualité tant de la visualisation elle-même que de l'expérience utilisateur (en termes, notamment de gigue, temps de réaction, etc.).

[0020] À cette fin, selon un premier aspect, la présente invention peut être mise en œuvre par un procédé pour la génération d'une vue d'un environnement tridimensionnel sur un ensemble de terminaux de communication à partir d'une modélisation dudit environnement stockée sur un serveur sous la forme d'une grille de voxels, ledit serveur mettant en œuvre des étapes de

- partition d'au moins une partie de ladite grille en un ensemble de sous-grilles;
- génération d'un maillage surfacique pour chaque sous-grille dudit ensemble,

- pour chacune desdites sous-grille, simplification d'une sous partie dudit maillage surfacique excluant une zone de bordure dudit maillage surfacique ; et création d'une table associant chaque vertex dudit bord à un voxel ;
- assemblage des maillages surfaciques desdites sous-grilles dudit ensemble, à l'aide de ladite table, afin de former un maillage surfacique global ;
- transmission dudit maillage surfacique global vers lesdites terminaux de communication, et génération d'une vue dudit environnement tridimensionnel en fonction dudit maillage surfacique global et d'un point de vue associé à chacun desdits terminaux de communication.

[0021] Suivant des modes de réalisation préférés, l'invention comprend une ou plusieurs des caractéristiques suivantes qui peuvent être utilisées séparément ou en combinaison partielle entre elles ou en combinaison totale entre elles :

- une surface de transition est associée auxdits voxels et en ce que lesdits maillages surfaciques sont générés en fonction desdites surfaces de transition ;
- lorsque ledit serveur détecte une modification de ladite modélisation, il insère dans une file d'attente, une demande de recalcul du maillage surfacique correspondant à ladite modification, et ce que les demandes de recalcul présentes dans ladite file et portant sur une même sous-grille sont traités de sorte à effectuer qu'une seule génération d'un maillage surfacique pour ladite sous-grille ;
- l'étape d'assemblage comporte une sous-étape de simplification supplémentaire du maillage surfacique global.

[0022] Selon un autre aspect, l'invention peut également être mise en œuvre par un programme d'ordinateur comportant des instructions de pour mettre en œuvre le procédé tel que précédemment décrit lorsqu'exécuté sur une ou plusieurs plateformes de traitement de l'information.

[0023] D'autres modes de réalisation de l'invention sont relatives à un serveur pour la génération d'une vue d'un environnement tridimensionnel à partir d'une modélisation dudit environnement, comportant une mémoire pour stocker ladite modélisation sous la forme d'une grille de voxels, et des moyens de traitement pour mettre en œuvre :

- partition d'au moins une partie de ladite grille en un ensemble de sous-grilles;

- génération d'un maillage surfacique pour chaque sous-grille dudit ensemble,
- pour chacune desdites sous-grille, simplification d'une sous partie dudit maillage surfacique excluant une zone de bordure dudit maillage surfacique ; et création d'une table associant chaque vertex dudit bord à un voxel ;
- assemblage des maillages surfaciques desdites sous-grilles dudit ensemble, à l'aide de ladite table, afin de former un maillage surfacique global ;
- transmission dudit maillage surfacique global vers un ensemble de terminaux de communication, chacun étant prévu pour la génération d'une vue dudit environnement tridimensionnel en fonction dudit maillage surfacique global et d'un point de vue associé à chacun desdits terminaux de communication.

[0024] Suivant des modes de réalisation préférés, l'invention comprend une ou plusieurs des caractéristiques suivantes qui peuvent être utilisées séparément ou en combinaison partielle entre elles ou en combinaison totale entre elles :

- une surface de transition est associée auxdits voxels et en ce que lesdits maillages surfaciques sont générés en fonction desdites surfaces de transition ;
- Le serveur est, en outre configuré pour, lorsque ledit serveur détecte une modification de ladite modélisation, il insère dans une file d'attente, une demande de recalcul du maillage surfacique correspondant à ladite modification, et ce que les demandes de recalcul présentes dans ladite file et portant sur une même sous-grille sont traités de sorte à effectuer qu'une seule génération d'un maillage surfacique pour ladite sous-grille ;
- l'assemblage comporte une simplification supplémentaire du maillage surfacique global.

[0025] D'autres modes de réalisation de l'invention sont relatives à un système comportant un serveur tel que précédemment défini et au moins un terminal de communication

[0026] D'autres caractéristiques et avantages de l'invention apparaîtront à la lecture de la description qui suit d'un mode de réalisation préféré de l'invention, donnée à titre d'exemple et en référence aux dessins annexés

[0027] BREVE DESCRIPTION DES FIGURES

[0028] Les dessins annexés illustrent l'invention :

La [Fig. 1] représente schématiquement un contexte général dans lequel peut s'inscrire l'invention un mode de réalisation.

La [Fig. 2] illustre schématiquement le principe des voxels et des vertex, selon un mode de réalisation de l'invention

La [Fig. 3] schématise un organigramme d'un procédé selon un mode de réalisation de l'invention :

La [Fig. 4] illustre un voxel et les vertex associés ;

La [Fig. 5] schématise un exemple de grille représentant un environnement tridimensionnel et une division en sous-grilles ;

#### [0029] DESCRIPTION DETAILLEE DE MODES DE REALISATION DE L'INVENTION

[0030] L'invention s'applique particulièrement aux jeux vidéo massivement multi-joueurs dans lesquels les différents joueurs peuvent interagir dans un environnement, ou univers, virtuel et tridimensionnel. Toutefois, l'invention peut également trouver des applications dans d'autres domaines d'activité nécessitant le partage d'un même environnement tridimensionnel entre plusieurs utilisateurs connectés simultanément à un serveur gérant cet environnement.

[0031] La figure 1 illustre un contexte dans lequel l'invention est susceptible de s'inscrire selon un mode de réalisation.

[0032] Au moins un terminal de communication 31, 32, 33 est connecté à un serveur 10 via un réseau de communication 20.

[0033] Ces terminaux de communication peuvent être toutes plateformes de traitement de l'information permettant de communiquer de l'information au travers de réseaux de télécommunication. Il peut notamment s'agir d'ordinateurs, de consoles connectées à un ordinateur, de smartphones, etc.

[0034] Ces terminaux de communication disposent également d'au moins un processeur et d'une mémoire permettant de stocker des instructions informatiques. Les moyens de communication peuvent permettre la connexion à différents types de réseau d'accès : réseaux cellulaires, notamment de 4<sup>e</sup> ou 5<sup>e</sup> génération, réseaux locaux de type WLAN ou WIFI, ou de proximité de type Bluetooth ou NFD (Near Field Communication), etc.

[0035] Le réseau de communication 20 peut être vu comme une interconnexion de sous-réseaux qui peuvent comprendre un réseau global comme Internet afin de permettre à des utilisateurs distants de se connecter à un même serveur. Le réseau de communication peut également comprendre des réseaux d'accès permettant la connexion des terminaux de communication 31, 32, 33 au réseau global « internet ».

[0036] Le serveur 10 peut être vu comme un dispositif de traitement de l'information unique, ou comme un serveur fonctionnel pouvant être déployé sur une pluralité de dispositifs arrangés sous la forme d'une ferme de serveur ou d'une infrastructure d'informatique en nuage (« *cloud computing* » en anglais).

[0037] Le serveur dispose d'une mémoire (ou d'un ensemble de mémoires) permettant de stocker une modélisation 11 de l'environnement tridimensionnel sous la forme d'une grille de voxels.

[0038] Le serveur dispose également des moyens logiciels pour générer, à partir de cette modélisation, un maillage surfacique global. Ce maillage surfacique global peut être transmis aux terminaux de communication par des messages 41, 42, 43, respectivement.

[0039] Dans le cadre de l'invention, une pluralité d'utilisateurs (ou joueurs) de terminaux de communication peuvent partager un même environnement tridimensionnel 11 géré par un unique serveur (fonctionnel) 10. Comme il sera vu plus loin, la génération d'un maillage surfacique global par le serveur permet de diminuer la quantité d'information à transmettre aux terminaux (par rapport à la transmission de la modélisation en voxels, par exemple).

[0040] Les terminaux peuvent ensuite générer une vue (subjective) de cet environnement tridimensionnel en fonction de ce maillage surfacique global. Ce maillage est le même pour l'ensemble des terminaux, mais la vue est subjective et dépend de la position et de paramètres de vue (direction, angle, etc.) de chaque utilisateur.

[0041] Dans la suite, on appelle « voxel » (pour « *Volume Element* » en anglais), l'équivalent en 3 dimensions d'un pixel en imagerie numérique 2D. Si l'on considère un univers 3D isotrope, un voxel a l'apparence d'un cube élémentaire.

[0042] De la même façon qu'un « pixel », on peut associer des données à un voxel. Ces données peuvent guider notamment l'apparence du voxel lors d'un affichage.

Elles peuvent notamment déterminer une couleur, un degré de transparence, une intensité lumineuse, etc.

[0043] La notion de voxel est classique en imagerie 3D et est expliquée en détail notamment sur la page Wikipédia : <https://fr.wikipedia.org/wiki/Voxel>

[0044] La figure 2 illustre un objet 200 discrétisé en voxels, certains étant visibles, d'autres étant masqués.

[0045] À chaque voxel 210 peut correspondre 8 vertex, 211, 212.... Un vertex est un sommet du voxel qui peut se caractériser par trois coordonnées x, y, z, dans un repère.

[0046] Un environnement tridimensionnel 11 peut être considéré comme une grille tridimensionnelle de voxels. Un objet de cet environnement est ainsi représenté par l'affectation d'une valeur particulière aux voxels correspondant à son emplacement. La valeur de ces voxels peut notamment correspondre à l'apparence de cet objet : couleur, matière, etc.

[0047] Un objet peut être de très grande taille (par exemple une planète), ou de taille plus réduite (vaisseaux spatiaux, personnages ou même outils).

[0048] La grille représentant l'environnement tridimensionnel peut être de taille considérable et dépend à la fois de la taille de cet environnement et de la résolution de la discrétisation (c'est-à-dire de la taille des voxels rapportée à la taille de l'univers).

[0049] Un des intérêts d'une telle modélisation d'un environnement sous la forme d'une grille de voxel est la relative simplicité de sa modification par les utilisateurs : il peut en effet suffire de modifier une valeur d'un voxel. Par exemple, construire un objet revient à modifier la valeur de voxels correspondant à l'emplacement et à la géométrie de cet objet. Creuser un objet (par exemple le sol d'une planète) revient à affecter une valeur « transparente » (correspondant à l'air) aux voxels de la zone extrudée, ce qui automatiquement laisse apparents les voxels masqués.

[0050] Selon différents modes de résolution, on peut associer à chaque voxel directement sa couleur ou une information de rendu, ou bien une information de plus haut-niveau sémantique, comme par exemple une matière (terre, roche, métal, plastique, air...). Dans ce dernier cas, le rendu (notamment la couleur, l'intensité...) sera déterminé à partir de cette information sémantique à une étape ultérieure, par exemple par le moteur de rendu déployé sur chaque terminal de communication.

[0051] En outre, d'autres types d'information peuvent être associés aux voxels de la grille. Par exemple, des informations géométriques peuvent y être associées, notamment des informations définissant une surface de transition.

[0052] Cette surface de transition permet d'influer sur l'apparence du voxel à un niveau d'échelle inférieure à celle du voxel. Elle permet notamment de modéliser des surfaces lisses (c'est-à-dire sans discontinuités) sur plusieurs voxels.

[0053] La figure 4 illustre un voxel pour lequel 8 vertex sont représentés  $V_{11}$ ,  $V_{12}$ ,  $V_{13}$ ,  $V_{14}$ ,  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$ , et une surface de transition A, B, C, D.

[0054] En indiquant, par exemple, la position des 4 sommets A, B, C, D, on peut définir un quadrilatère qui partitionne le voxel en deux parties :

- une première partie,  $V_{11}$ ,  $V_{12}$ ,  $V_{13}$ ,  $V_{14}$ ,  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$ , représentée par des traits pointillés sur la figure 4, peut correspondre à la valeur de rendu (couleur, matière...) du voxel, et,
- une seconde partie  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$ , A, B, C, D peut correspondre à la valeur de rendu du voxel voisin par la face  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$ .

[0055] Cette surface de transition est essentiellement opérante lorsque le voxel voisin est transparent. Auquel cas, le voxel  $V_{11}$ ,  $V_{12}$ ,  $V_{13}$ ,  $V_{14}$ ,  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$  pourra être représenté comme une forme  $V_{11}$ ,  $V_{12}$ ,  $V_{13}$ ,  $V_{14}$ ,  $V_{21}$ ,  $V_{22}$ ,  $V_{23}$ ,  $V_{24}$ , et non pas comme un cube.

[0056] Si les surfaces de transition sont correctement définies pour les voxels voisins, par les autres faces ( $V_{12}$ ,  $V_{13}$ ,  $V_{22}$ ,  $V_{23}$ ), ( $V_{11}$ ,  $V_{14}$ ,  $V_{21}$ ,  $V_{24}$ ), ( $V_{14}$ ,  $V_{13}$ ,  $V_{24}$ ,  $V_{23}$ ), ( $V_{11}$ ,  $V_{12}$ ,  $V_{21}$ ,  $V_{22}$ ), on peut ainsi obtenir, par la concaténation des surfaces d'interfaces respectives, une surface continue sur plusieurs voxels. Ce résultat permet un rendu esthétique plus naturel, et d'éviter une discrétisation des objets et du paysage comme par exemple avec l'environnement « Minecraft ».

[0057] À partir d'une telle modélisation de l'environnement, il est possible de générer un maillage surfacique global de l'environnement. Ce maillage est une structure de données représentant uniquement la surface de l'environnement tridimensionnel, c'est-à-dire ce qui est susceptible d'être visible aux différents utilisateurs en fonction de leur point de vue. Ce maillage, qui ne dépend pas des utilisateurs, est généré chaque fois que nécessaire et transmis aux différents terminaux de communication

connecté au serveur. Comme évoqué précédemment, chaque terminal peut alors générer une vue, ou un « rendu », de l'univers sur la base à la fois de ce maillage surfacique global et de ses données subjectives (position dans l'environnement, direction/angle de vue, etc.).

[0058] Selon l'invention, la génération du maillage surfacique global est réalisée par le serveur en plusieurs étapes.

[0059] Une première étape initiale, S1 sur la Figure 3, consiste à partitionner au moins une partie de la grille de l'environnement tridimensionnel en un ensemble de sous-grilles.

[0060] Il apparaît que considérer l'intégralité de la grille voxel à chaque étape est doublement inefficace :

- la consommation mémoire est de l'ordre de plusieurs giga-octets, uniquement pour la donnée voxel d'entrée de l'algorithme,
- En outre, la moindre modification de la grille voxel nécessiterait un recalcul complet du maillage et application de l'algorithme de simplification (étapes S2-S6 sur la figure 3)

[0061] Pour remédier à ces problèmes, on découpe la grille principale en sous-grilles. Ces sous-grilles peuvent être cubiques et de taille fixe.

[0062] La figure 5 schématise un exemple de grille 400 représentant un environnement tridimensionnel. Dans cet exemple, cette grille a été partitionnée en 4 sous-grilles, 410, 420, 430, 440. Chacune de ces sous-grilles 410 comportant une pluralité de voxels 411, 412, 413, 414, 415... Bien entendu, dans la réalité les sous-grilles comportent un nombre nettement plus substantiel de voxels, et la grille comporte un bien plus grand nombre de sous-grilles. Également, l'écart entre les sous-grilles dans la figure 5 vise uniquement à la clarté de la figure et ne représente aucune réalité concrète : ainsi, le vertex A se trouve sur l'arrête  $V_{11}$ - $V_{21}$ , le vertex B se trouve sur l'arrête  $V_{14}$ - $V_{24}$ , etc.

[0063] Chaque sous-grille forme donc une structure de données de taille substantiellement plus réduite que celle de la grille globale. En outre, il est possible de déterminer la taille des sous-grilles de façon optimale en fonction des ressources computationnelles dont dispose le serveur 10.

[0064] Les étapes subséquentes de la génération du maillage surfacique global peuvent être effectuées initialement, et à nouveau lorsque la modélisation 11 est modifiée.

[0065] Ces modifications peuvent avoir, comme évoqué précédemment d'autres causes : la création d'un nouvel objet, la destruction d'un objet, la modification d'un objet, etc. D'une façon générale, ces différentes causes résultent sur la modification d'une valeur associée à un voxel (couleur, matière, surface de transition...).

[0066] Selon un mode de réalisation, la détection d'une modification dans la modélisation engendre une demande de recalcul qui est placée dans une file d'attente. Cette file d'attente est consommée par un module de génération de maillage. Un mécanisme permet d'agréger plusieurs demandes de recalcul : ainsi, il serait contreproductif de calculer plusieurs fois un maillage surfacique pour plusieurs modifications successives d'une même sous-grille, par exemple.

[0067] Ainsi, selon un mode de réalisation de l'invention, lorsque le module de génération de maillage est disponible, il consomme la première demande dans la file et recherche d'autres demandes portant sur la même sous-grille que cette première demande (dans la totalité de la file d'attente ou selon une fenêtre temporelle...), et génère un maillage surfacique pour cette sous-grille qu'une unique fois, en prenant en compte les différentes modifications (correspondant aux demandes respectives).

[0068] Ainsi, selon ce mode de réalisation, une file d'attente de type FIFO (« First In, First Out ») est mise en place. Cette mise en œuvre est intéressante dans la mesure où l'on a une capacité de calcul fixe et finie sur le serveur et permet de réduire la fréquence de mise-à-jour des maillages si le serveur est saturé et d'avoir des mises-à-jour plus rapides dans le cas où il y a de la capacité de calcul disponible. Cette approche est également intéressante dans notre contexte car elle permet de réduire le nombre de transmission du maillage et d'ainsi réduire le coût en bande passante.

[0069] Dans un cas où la demande de mise-à-jour de maillage serait très variable, il pourrait être intéressant d'opter pour une autre mise en œuvre basée sur une infrastructure de calcul de taille dynamique. Une infrastructure de taille dynamique serait également pertinente si la fréquence de mise-à-jour des maillages était une métrique à garantir fortement.

[0070] Dans une étape S2, le serveur peut générer un maillage surfacique pour une ou plusieurs sous-grilles. Typiquement, donc, uniquement les sous-grilles impactées par une modification de la modélisation peuvent être générées à nouveau.

[0071] La donnée voxel ne peut typiquement pas être visualisée directement. Aussi, on génère un maillage représentant l'interface entre les voxels opaques et les voxels transparents pour permettre à un utilisateur de visualiser cette donnée. Dans le cas où seule l'apparence est nécessaire, toute l'information qui ne se trouve pas à cette interface est donc complètement superflue.

[0072] Pour générer un maillage représentant la donnée voxel, on peut considérer des paires de voxels adjacents. Si un des deux voxels a un matériau opaque et l'autre transparent, alors on génère un quadrilatère entre les deux voxels auquel on attribue le matériau opaque.

[0073] Selon un mode de réalisation, on ajuste la position des sommets du quadrilatère en utilisant les informations sur la surface de transition associée au voxel, comme expliqué précédemment.

[0074] De cette façon, en considérant l'ensemble des paires de voxels adjacents d'une sous-grille, on obtient effectivement un maillage surfacique, c'est-à-dire une structure de données décrivant uniquement la surface d'interface entre les voxels (ou parties de voxels) opaques (correspondant à une matière différente de l'air) et les voxels (ou parties de voxels) transparents (correspondant à l'air). Ce maillage surfacique correspond à la partie visible de l'environnement tridimensionnel.

[0075] Selon un mode de réalisation, la génération d'un maillage surfacique pour une sous-grille comprend une sous-étape de triangulation, consistant à transformer les quadrilatères correspondant à chaque voxel en triangles.

[0076] Cette triangulation engendre une tessellation, ou pavage, de la sous-grille. Le nombre de triangles peut néanmoins être très important, ce qui est sous-optimal pour un traitement efficace par le serveur.

[0077] La littérature propose de nombreuses solutions pour simplifier une telle triangulation. Notamment, il est possible d'agréger les triangles coplanaires pour constituer des polygones de plus grande taille.

[0078] Une simplification simple peut se baser sur l'analyse de chaque point (vertex) du maillage surfacique pour vérifier si les triangles adjacents sont coplanaires ou non, et si oui, ce point est supprimé et les triangles en question sont fusionnés en un polygone unique.

[0079] D'autres algorithmes existent et permettent d'optimiser la simplification du maillage.

[0080] Il est à noter que dans notre cas, les maillages de départ ne sont pas de « bonne qualité ». En particulier, le maillage disponible à ce stade n'est pas manifold. Un maillage est dit n-manifold quand, en une arête, sont connectées au maximum n surfaces planes. Un maillage 2-manifold est donc un réseau de surfaces pour lesquelles les arêtes sont la frontière d'au maximum 2 surfaces planes

[0081] Les maillages non-manifold sont généralement difficiles à manipuler et posent problèmes à de nombreux algorithmes de simplifications. Une sélection doit donc être opérée pour choisir des algorithmes convenants dans le cadre des contraintes spécifiques de l'invention.

[0082] Ainsi, on peut par exemple citer l'algorithme QSlim, disponible comme outil logiciel libre, et décrit dans la thèse de l'auteur M. Garland, « *Quadric-based Polygonal Surface Simplification* », School of Computer Science of Pittsburgh, Mai 1999.

[0083] On peut également citer l'algorithme décrit dans « *Fast and Robust QEF Minimization using Probabilistic Quadrics* » de Philip Trettner, Leif Kobbelt, disponible à l'adresse <http://www.graphics.rwth-aachen.de/publication/03308/>

[0084] Selon l'invention, l'algorithme de simplification est mis en œuvre sur uniquement une sous partie du maillage surfacique de la sous-grille. Cette sous-partie correspond à la sous-grille dont a été exclue une zone de bordure.

[0085] On exclue en effet du maillage une zone de bordure. Cette zone de bordure peut se définir par une marge sur les limites de la sous-grille.

[0086] En particulier, on peut définir la zone de bordure par l'ensemble des triangles du maillage dont au moins un des vertex appartient au bord du maillage de la sous-grille (c'est-à-dire à un voxel ayant un voisin immédiat dans le maillage d'une autre sous-grille).

[0087] Le but est d'éviter toute perte d'information, qui sera nécessaire pour déterminer la jonction entre les deux sous-grilles et évaluer l'erreur de simplification de la géométrie.

[0088] Cette table indique de quels voxels proviennent les différents sommets des triangles au bord de la zone.

[0089] Selon un mode de réalisation de l'invention, on associe un vertex à un unique voxel selon une règle prédéfinie. Par exemple, sur la base d'un repère donné, on associe arbitrairement chaque vertex au voxel en bas à gauche.

[0090] La table permet ainsi une mise en correspondance unique entre la grille de voxels et la grille de vertex : [(vertex 1, voxel 1), (vertex 2, voxel 2),...]

[0091] Chaque maillage peut être généré indépendamment des maillages voisins et ne nécessite que la mémoire nécessaire pour une sous-grille. En outre, le fait que les sous-grilles aient une taille fixe et connue permet de dimensionner le serveur de façon optimale en adéquation avec le choix de la taille des sous-grilles.

[0092] Par exemple, on peut choisir de travailler avec des sous-grilles de  $32^3 \cong 32000$  voxels. La structure de données de chaque voxel représente une trentaine d'octets d'information. Une sous-grille représente donc environ 1 méga-octet

[0093] En général, une majeure partie d'une sous-grille ne contient pas de géométrie (car il n'y a pas de transition entre air et matière). Le maillage correspondant est donc généralement petit devant la taille des données voxel. Pour des constructions non dégénérées, on est de l'ordre de 10 kilo-octets par maillage de sous-grille après l'étape de simplification partielle du maillage de la cellule.

[0094] Un exemple de maillage peut être composé de 4000 sous-grilles, ce qui correspond à une taille de traitement de quelques centaines de méga-octets pour la fabrication et la simplification du maillage complet.

[0095] Dans une étape S4, le serveur assemble les maillages surfaciques des sous-grilles afin de former un maillage surfacique global.

[0096] Comme vu précédemment, le serveur ne peut générer que les maillages de sous-grilles impactés par une modification de la modélisation de l'environnement. Dès lors, l'assemblage peut consister à assembler des sous-grilles d'ancienneté variée.

[0097] Les maillages intermédiaires générés à l'étape précédente, S3, sont relativement petits du fait de la simplification et de la taille contrôlée des sous-grilles. Il n'y a pas de problème particulier pour le serveur à les avoir tous en mémoire simultanément.

[0098] Cet assemblage consiste à concaténer les maillages de chaque sous-grille et à utiliser les tables associées à chaque sous-grille pour former la zone de jonction entre deux maillages contigus.

[0099] Selon un mode de réalisation de l'invention, on regarde tous les voxels de jonction dans les tables. S'il y a une transition d'air à matière dans la jonction (air dans une sous-grille et matière dans la sous-grille voisine), alors on ajoute des triangles en reliant les vertex associés à ces voxels.

[0100] Selon un mode de réalisation de l'invention, l'étape d'assemblage comprend en outre une sous-étape de simplification du maillage surfacique global

[0101] L'algorithme de cette simplification peut être similaire à celui déployé à l'étape S3 mais sur l'ensemble de la grille et sans procéder à l'exclusion d'une zone de bordure.

[0102] Cette nouvelle simplification permet d'une part de simplifier les zones de bordure (qui n'ont pu être simplifiées lors de l'étape S3), maintenant jointes deux à deux, et également de détecter des zones pouvant être simplifiées sur plusieurs sous-grilles. Par exemple, des polygones peuvent être coplanaires sur deux (ou davantage) sous-grilles et, auquel, il est possible de procéder à une simplification du maillage global en fusionnant ces polygones.

[0103] Ce maillage surfacique global, après simplification supplémentaire éventuelle, peut ensuite être transmis par le serveur, lors d'une étape S5, vers les terminaux de communication.

[0104] Chacun d'entre eux peut alors, indépendamment, générer, dans une étape S6, une vue de l'environnement tridimensionnel en fonction dudit maillage surfacique global et d'un point de vue associé au terminal. Cette étape de construction d'une vue à partir d'un maillage surfacique est classique en soi et peut être mise en œuvre par différentes techniques de rendu 3D.

[0105] À titre d'exemple, on peut citer la méthode bien connue du « Z buffer », qui permet de gérer le problème de la visibilité en déterminant quels éléments de la scène doivent être rendus, lesquels sont cachés par d'autres et dans quel ordre l'affichage des objets doit se faire.

[0106] Cette étape de génération d'une vue peut en outre comprendre une projection du maillage surfacique vers la surface bidimensionnelle représentant l'écran sur lequel l'utilisateur du terminal peut visualiser l'environnement.

[0107] Ainsi, selon l'invention, uniquement des données représentant un maillage surfacique simplifié sont transmis vers les terminaux de communication. Ces données sont faiblement volumineuse par rapport à celle de la modélisation en voxels. En outre, ce maillage surfactique est agnostique par rapport aux points de vue des utilisateurs. Ainsi, une transmission de type « diffusion » (« broadcasting ») peut être mise en œuvre et une partie des calculs déportés vers les terminaux des utilisateurs.

[0108] La division d'une grille en sous-grilles permet par ailleurs de garder la charge du serveur sous contrôle en permettant de manipuler des structures de données de taille réduite et en minimisant le volume de données à recalculer à chaque modification de l'environnement.

[0109] Ce mécanisme est particulièrement approprié dans le cas où le nombre de modifications de l'environnement est relativement faible par rapport au nombre d'utilisateurs navigant dans l'environnement et devant donc recevoir des mises à jour de l'environnement. En effet, une modification va n'impacter qu'une sous-grille et nécessiter qu'un minimum marginal réduit : recalcul de la sous-grille, et recalcul du maillage surfacique global par assemblage.

[0110] Bien entendu, la présente invention n'est pas limitée aux exemples et au mode de réalisation décrits et représentés, mais est défini par les revendications. Elle est notamment susceptible de nombreuses variantes accessibles à l'homme de l'art.

## Revendications

1. Procédé pour la génération d'une vue d'un environnement tridimensionnel sur un ensemble de terminaux de communication à partir d'une modélisation dudit environnement stockée sur un serveur sous la forme d'une grille de voxels, ledit serveur mettant en œuvre des étapes de

- partition (S1) d'au moins une partie de ladite grille en un ensemble de sous-grilles;
- génération (S2) d'un maillage surfacique pour chaque sous-grille dudit ensemble,
- pour chacune desdites sous-grille, simplification (S3) d'une sous partie dudit maillage surfacique excluant une zone de bordure dudit maillage surfacique ; et création d'une table associant chaque vertex dudit bord à un voxel ;
- assemblage (S4) des maillages surfaciques desdites sous-grilles dudit ensemble, à l'aide de ladite table, afin de former un maillage surfacique global ;
- transmission (S5) dudit maillage surfacique global vers lesdites terminaux de communication, et génération (S6) d'une vue dudit environnement tridimensionnel en fonction dudit maillage surfacique global et d'un point de vue associé à chacun desdits terminaux de communication.

2. Procédé selon la revendication précédente, dans lequel une surface de transition est associée auxdits voxels et en ce que lesdits maillages surfaciques sont générés en fonction desdites surfaces de transition.

3. Procédé selon l'une des revendications précédentes dans lequel lorsque ledit serveur détecte une modification de ladite modélisation, il insère dans une file d'attente, une demande de recalcul du maillage surfacique correspondant à ladite modification, et ce que les demandes de recalcul présentes dans ladite file et portant sur une même sous-grille sont traités de sorte à effectuer qu'une seule génération d'un maillage surfacique pour ladite sous-grille.

4. Procédé selon l'une des revendications précédentes, dans lequel l'étape d'assemblage (S4) comporte une sous-étape de simplification supplémentaire du maillage surfacique global.

5. Programme d'ordinateur comportant des instructions pour mettre en œuvre le procédé selon l'une quelconque des revendications précédentes lorsqu'exécuté sur une ou plusieurs plateformes de traitement de l'information.

6. Serveur pour la génération d'une vue d'un environnement tridimensionnel à partir d'une modélisation dudit environnement, comportant une mémoire pour stocker ladite modélisation sous la forme d'une grille de voxels, et des moyens de traitement pour mettre en œuvre :

- la partition (S1) d'au moins une partie de ladite grille en un ensemble de sous-grilles;
- la génération (S2) d'un maillage surfacique pour chaque sous-grille dudit ensemble,
- pour chacune desdites sous-grille, la simplification (S3) d'une sous partie dudit maillage surfacique excluant une zone de bordure dudit maillage surfacique ; et création d'une table associant chaque vertex dudit bord à un voxel ;
- l'assemblage (S4) des maillages surfaciques desdites sous-grilles dudit ensemble, à l'aide de ladite table, afin de former un maillage surfacique global ;
- la transmission (S5) dudit maillage surfacique global vers un ensemble de terminaux de communication, chacun étant prévu pour la génération (S6) d'une vue dudit environnement tridimensionnel en fonction dudit maillage surfacique global et d'un point de vue associé à chacun desdits terminaux de communication.

7. Serveur selon la revendication précédente, dans lequel une surface de transition est associée auxdits voxels et en ce que lesdits maillages surfaciques sont générés en fonction desdites surfaces de transition.

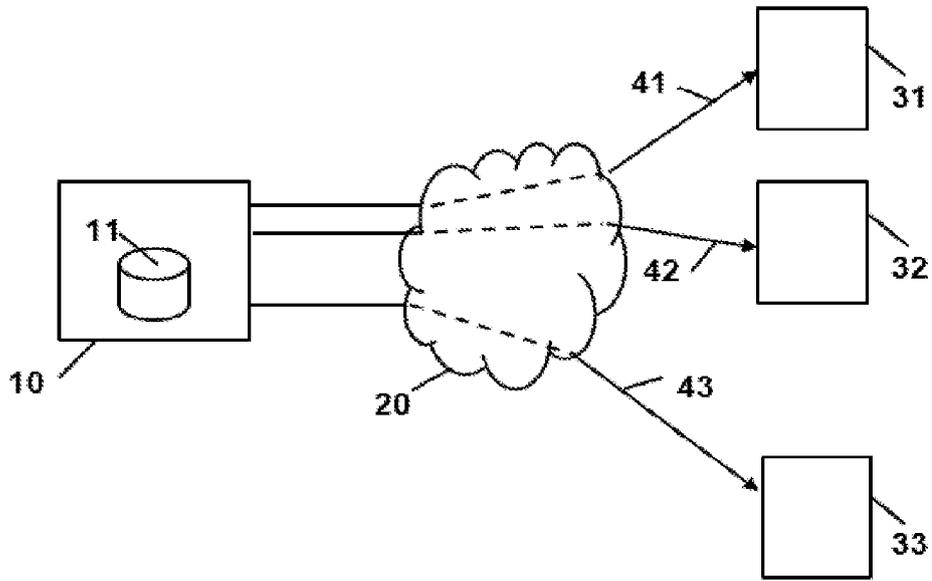
8. Serveur selon l'une des revendications 6 ou 7, en outre configuré pour, lorsque ledit serveur détecte une modification de ladite modélisation, il insère dans une file

d'attente, une demande de recalcul du maillage surfacique correspondant à ladite modification, et ce que les demandes de recalcul présentes dans ladite file et portant sur une même sous-grille sont traités de sorte à effectuer qu'une seule génération d'un maillage surfacique pour ladite sous-grille.

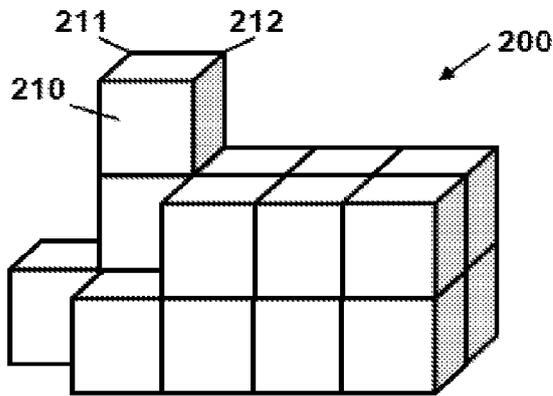
9. Serveur selon l'une des revendications 6 à 8, dans lequel l'assemblage (S4) comporte une simplification supplémentaire du maillage surfacique global.

10. Système comportant un serveur selon l'une des revendications 6 à 9 et au moins un terminal de communication.

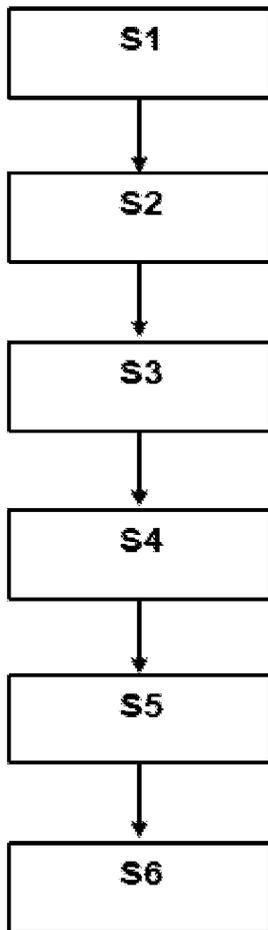
[Fig. 1]



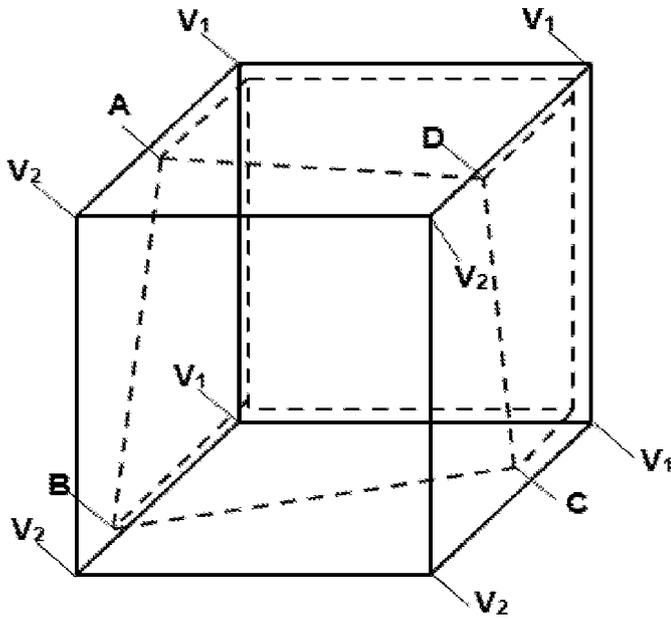
[Fig. 2]



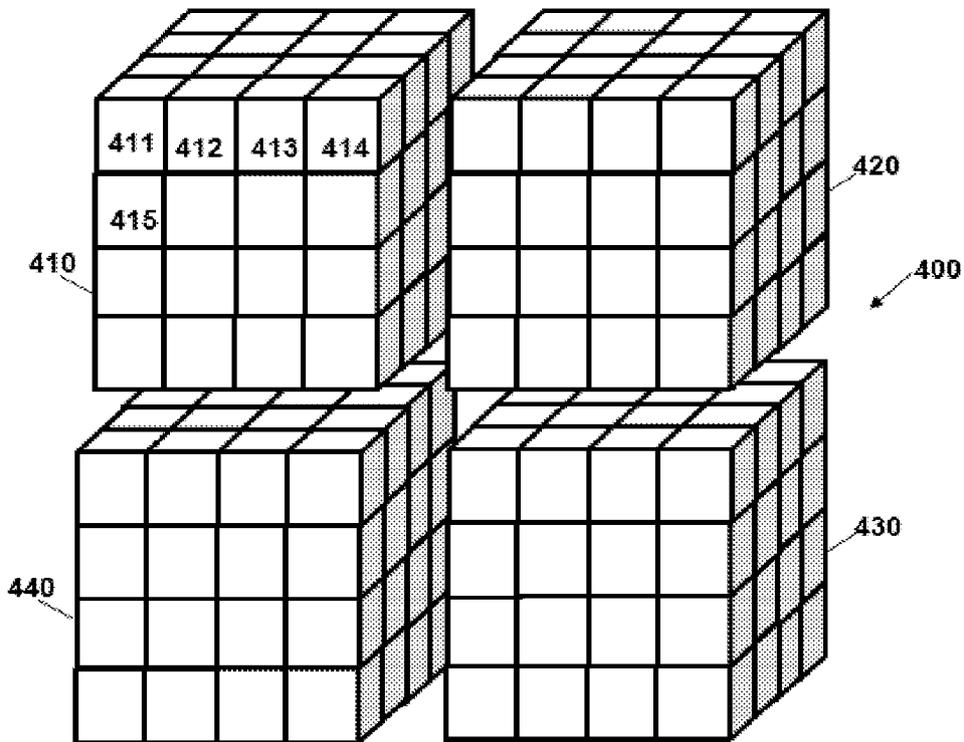
[Fig. 3]



[Fig. 4]



[Fig. 5]



[Fig. 5]

