



Arbres couvrants arête-disjoints dans les grilles toriques d-dimensionnelles pour la diffusion de messages longs

Eric Darrot

► To cite this version:

Eric Darrot. Arbres couvrants arête-disjoints dans les grilles toriques d-dimensionnelles pour la diffusion de messages longs. RR-3424, INRIA. 1998. inria-00073266

HAL Id: inria-00073266

<https://hal.inria.fr/inria-00073266>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

*Arbres couvrants arête-disjoints
dans les grilles toriques d-dimensionnelles
pour la diffusion de messages longs*

Eric Darrot

N° 3424

Mai 1998

THÈME 1



*Rapport
de recherche*

Arbres couvrants arête-disjoints dans les grilles toriques d -dimensionnelles pour la diffusion de messages longs

Eric Darrot

Thème 1 — Réseaux et systèmes
Projet SLOOP

Rapport de recherche n° 3424 — Mai 1998 — 26 pages

Résumé : La diffusion de messages longs est un schéma de communication globale dont l'optimisation est essentielle dans un contexte de calcul parallèle intensif. Après avoir exposé les arguments qui font préférer une approche *store-and-forward* plutôt que *wormhole* (on montre alors qu'il est optimal de découper le message initial et de «pipe-liner» les tronçons obtenus sur des arbres couvrants disjoints), je donne les définitions, conventions et quelques résultats qui me permettent de poser le problème dans le cas général. Je décris ensuite brièvement les résultats connus dans le cas bidimensionnel. Puis, après avoir fixé un certain nombre de notations, je donne une construction du maximum d'arbres couvrants arête-disjoints dans les tores d -dimensionnels, pour tout $d > 2$, en traitant le cas tridimensionnel à part, pour des raisons de clarté de l'exposé. Tous les arbres sont obtenus par le même algorithme, qui utilise la structure récursive du graphe. Leur profondeur tend vers le double du diamètre quand le nombre de dimensions tend vers l'infini. Enfin, je conclus par une étude comparative des performances estimées de l'algorithme de diffusion issu de cette construction, dans une optique SPMD et pour une machine représentative des supercalculateurs modernes.

Mots-clés : algorithmique parallèle, diffusion, *store-and-forward*, grille torique, arbres couvrants

SLOOP est un projet commun CNRS/INRIA/UNSA

Edge-disjoint spanning trees in d -dimensional toroidal grids for long messages broadcasting

Abstract: Broadcasting long messages is a communication scheme which is essential to optimize for parallel intensive calculation. I expose some arguments that make store-and-forward approach preferable to wormhole. In particular, it has been shown that in this context, it is optimal to cut up the initial message into packets and to pipeline them on disjoint spanning trees. The goal is then to find the maximum number of disjoint spanning trees in the network. I give the definitions, conventions and some results which permits to introduce the problem in the general case and I describe briefly the known results for the bidimensional case. Then, I give a construction of the maximum number of edge-disjoint spanning trees in d -dimensional toroidal grids, for any $d > 2$, solving the tridimensional case separately in order to make the proof clearer. All the trees are obtained by the same algorithm which uses the recursive structure of the graph. The depth of the trees tends to twice the diameter when the number of dimensions tends to infinity. I conclude by a study of estimated performances of the broadcasting algorithm deduced from this construction for a representative example of modern supercomputers.

Key-words: parallel computing, broadcasting, store-and-forward, toroidal mesh, spanning trees

1 Introduction

La diffusion est l'un des schémas de communication les plus fréquemment rencontrés lors de l'écriture de programmes parallèles. Plus encore, on oublie trop souvent que la diffusion est la première communication globale qui s'effectue lors de l'exécution d'un programme parallèle sur une machine à mémoire distribuée. En effet, le temps d'exécution d'un tel programme est égal à son temps de *chargement* dans la mémoire locale des nœuds de calcul du réseau, plus son temps d'exécution proprement-dit (le temps d'exécution algorithmique). Il est donc crucial d'optimiser la diffusion de messages longs au sein des réseaux d'interconnexion des machines distribuées.

REMARQUE. La notion de message long dépend de beaucoup de paramètres liés aux caractéristiques physiques du réseau. On peut cependant estimer que la taille d'un programme exécutable sur une machine parallèle (de l'ordre de quelques centaines de Kilo-octets à plusieurs Mega-octets, des portions de bibliothèques de communication étant souvent ajoutées lors de la compilation) est un message qui sera toujours considéré long dans les contextes connus actuellement (voir section 8). De même, les échanges de données entre nœuds de calcul portent souvent sur des portions de matrices ou des matrices entières, qui entreront souvent dans la catégorie des messages longs.

Les mécanismes mis en œuvre pour résoudre les communications entre deux nœuds d'un réseau peuvent être classifiés en deux catégories au principe fondamentalement différent : les routages par commutation de données, représentés par le *store-and-forward*, et les routages par commutation de circuits, représentés par le *wormhole* (voir [9]). Dans le livre [26], on trouvera des références ou le détail d'autres modes.

En *store-and-forward*, l'envoi d'un message entre deux nœuds quelconques se fait par une succession de communications élémentaires, c'est-à-dire ne mettant en rapport que des nœuds voisins, reliés physiquement par un lien. Les messages avancent dans le réseau vers leur destination en transitant dans les nœuds intermédiaires, où ils sont stockés avant d'être réemis (d'où le terme anglo-saxon consacré *store-and-forward*). A chaque étape, un message doit donc être arrivé dans sa totalité avant de pouvoir être réemis. Pour calculer les temps de communication, on utilise généralement le modèle suivant qui permet, si l'on arrive à déterminer avec précision les paramètres dépendant d'un contexte donné, d'estimer finement le temps d'exécution d'algorithmes de communication complexes. Dans ce modèle, on considère que le temps de communication d'un message de longueur L entre deux nœuds x et y à distance d l'un de l'autre est :

$$\textit{store-and-forward} : T_{x \rightarrow y} = d(\beta + L\tau) = d\beta + dL\tau$$

où β est un temps de *start-up* (temps d'initialisation de la communication) et τ est le taux de transmission d'un lien (inverse de la bande passante, exprimée en unité de quantité d'information par unité de temps, par exemple bits/s ou octets/s), que l'on suppose uniforme pour tout le réseau. Ce mode de commutation est donc très pénalisant pour d ou L grands. On verra plus loin comment on arrive à masquer les distances et à réduire le terme en $L\tau$ dans le cas de la diffusion de messages longs.

Dans le cas du *wormhole*, le message est découpé en paquets de taille fixe qui suivent le même itinéraire, établi par le paquet de tête. On modélise le temps de communication de façon générale par :

$$\textit{wormhole} : T_{x \rightarrow y} = \alpha + d\delta + L\tau$$

où α est aussi un temps de *start-up*, δ est le temps de commutation physique des routeurs de l'émetteur et des nœuds intermédiaires lors de l'établissement du circuit virtuel par le paquet de tête, et τ est toujours le taux de transmission uniforme des liens. Pour une synthèse sur l'étude des communications en *wormhole*, on pourra se référer aux thèses [10] et [14].

REMARQUE. Dans le cas de transmissions entre voisins, on a $d = 1$, soit : $T_{\text{voisins}} = \alpha + \delta + L\tau$. On peut alors considérer les deux mécanismes de routage équivalents en posant $\alpha + \delta = \beta$, ce qui revient à dire qu'on peut simuler un routage *store-and-forward* en *wormhole*.

Le mécanisme mis en œuvre en *wormhole* rend ce mode de commutation très sensible aux conflits (risque d'interblocages), donc mal adapté à la transmission de messages longs (voir [9]). Dans ce cas, on choisit donc de raisonner en considérant un routage de type commutation de messages (*store-and-forward*). Ce choix n'est pas réducteur, puisqu'on peut adapter les algorithmes *store-and-forward* au *wormhole* en n'autorisant que les communications entre voisins. Les contraintes physiques considérées correspondent au modèle H_* : liens bidirectionnels *half-duplex* (un seul message peut emprunter le lien à un instant donné) et nœuds Δ -port (un nœud peut recevoir ou envoyer sur tous ses ports simultanément), voir [9] ou [17].

Avec ces hypothèses sur le fonctionnement du réseau, on montre que la meilleure façon de réaliser une diffusion est d'utiliser la technique du pipe-line sur arbres couvrants disjoints (voir [9] ou [17]). La technique du pipe-line permet d'uniformiser le temps d'occupation des nœuds en masquant autant que possible les distances pour se rapprocher de l'effet *wormhole*. Quant à la recherche d'arbres couvrants disjoints, son but est d'introduire un découpage de plus haut niveau du message initial, en tronçons que l'on pipe-linera chacun dans un arbre différent. Evidemment, comme la technique du pipe-line requiert la disponibilité des liens empruntés par les paquets pendant toute la durée de la communication (on discrétise le temps de remplissage et de vidage du pipe-line), les arbres couvrants doivent être deux à deux *arête-disjoints*. On peut aussi utiliser les arbres ainsi construits pour effectuer simultanément plusieurs diffusions. Cependant, si de tels arbres existent, il n'est pas certain qu'ils soient de profondeur optimale (égale au diamètre). On doit donc aussi chercher à minimiser leur profondeur. En calculant une taille de paquet optimale (voir [1], [17] ou [27]), le temps global de diffusion pipe-linée d'un message de longueur L sur t arbres couvrants arête-disjoints de profondeur au plus p (et de même racine, correspondant au nœud initiateur de la communication) est noté $b(L, p, t)$ et vaut :

$$\left(\sqrt{(p-1)\beta} + \sqrt{\frac{L\tau}{t}} \right)^2$$

Il est évident que le gain de temps obtenu avec cette technique croît avec la longueur du message : tout est fait pour réduire au maximum le terme en $L\tau$, puisque l'on considère L très grand. Or, dans les études qui ont été faites jusqu'à présent en *wormhole*, on s'attache surtout à optimiser le nombre d'étapes de communications, une étape consistant en un ensemble de communications, simultanées et sans conflit, entre nœuds distants qui deviennent les origines des communications dans l'étape suivante. Il en résulte un terme en $L\tau$ très supérieur à celui obtenu ici, que l'optimisation des temps d'initialisation des communications a du mal à compenser dans le cas de messages longs. Actuellement, on commence donc à s'orienter vers des techniques de compromis entre nombre d'étapes et flot d'information (voir la thèse de Delmas [10] à ce sujet). D'une manière générale, et considérant les architectures actuelles et les réseaux implémentés sur ces machines, on préférera raisonner en *store-and-forward* et appliquer au *wormhole* les algorithmes obtenus, comme cela a été fait avec succès dans le cas de l'hypercube (voir [21] et [28]). On peut même imaginer n'avoir dans le futur que des machines implémentant un routage de type commutation de circuits, et y simuler des algorithmes de pipe-line en *store-and-forward* dans des cas bien définis : en fixant les valeurs de α , de δ et de τ , on peut donner la taille de message à partir de laquelle l'une ou l'autre des approches est la meilleure, pour une topologie donnée (voir section 8).

Enfin, même si ce n'est généralement pas la motivation essentielle de cette recherche, on peut aussi mettre en avant le fait que l'existence de t arbres couvrants arête-disjoints dans un graphe lui permet de réaliser au moins une diffusion en tolérant jusqu'à $t - 1$ pannes sur ses liens.

La diffusion pipe-linée sur arbres couvrants disjoints est présentée dans le cas général dans [1], [2] et dans l'article de synthèse [17]. Plus précisément, citons, en plus des résultats présentés dans la section suivante qui font référence au cas de la grille torique, les travaux de Bermond et Fraigniaud [3], récemment améliorés par Ge et Hakimi [18], et de Heydemann, Opatrny et Sotteau [19] dans le cas du graphe de de Bruijn, et ceux de Ho et Johnsson [20] dans le cas de l'hypercube.

Nous allons maintenant voir une construction du nombre maximum d'arbres couvrants arête-disjoints dans les grilles toriques pour toute dimension $d > 2$. Le cas tridimensionnel a déjà fait l'objet d'une publication dans [8].

2 Définitions et notations

DEFINITION. La grille de dimension d , notée $M(n_0, n_1, \dots, n_{d-1})$, est la somme cartésienne de d chaînes ($d > 1$) de n_i sommets ($n_i > 1, \forall i \in [0, d-1]$).

C'est un graphe irrégulier de degré maximum $\Delta = 2d$, à $N = \prod_{i=0}^{d-1} n_i$ sommets et $E = \sum_{i=0}^{d-1} \left(\frac{(n_i-1)N}{n_i} \right)$ arêtes. Son diamètre est $D = \sum_{i=0}^{d-1} (n_i - 1)$.

La définition générale de la grille torique est très similaire :

DEFINITION. La grille torique de dimension d (ou d -tore, ou *tore d -dimensionnel*), notée $TM(n_0, n_1, \dots, n_{d-1})$, est la somme cartésienne de d cycles ($d > 1$) de n_i sommets ($n_i > 1, \forall i \in [0, d-1]$).

C'est un graphe régulier de degré $\Delta = 2d$, à $N = \prod_{i=0}^{d-1} n_i$ sommets et $E = Nd$ arêtes. Son diamètre est $D = \sum_{i=0}^{d-1} \lfloor \frac{n_i}{2} \rfloor$.

REMARQUE. Dans le cas où on a $n_i = n, \forall i \in [0, d-1]$, on note $M(n)^d$ (resp. $TM(n)^d$) la grille (resp. la grille torique) de dimension d , que l'on dira alors *de côté n* .

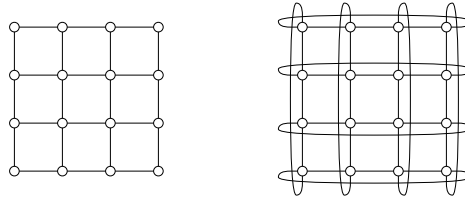


FIG. 1 – La grille $M(4)^2$ et la grille torique $TM(4)^2$

Ces quelques conventions seront valables dans toute la suite :

- On représente la grille et la grille torique dans un espace cartésien à d dimensions. Un sommet x de $M(n_0, n_1, \dots, n_{d-1})$ ou $TM(n_0, n_1, \dots, n_{d-1})$ est donc représenté par un vecteur \vec{x} de coordonnées $(x_0, x_1, \dots, x_{d-1})$, avec $x_i \in [0, n_i - 1]$.
- Une arête relie deux sommets x et y s'ils diffèrent de 1 sur exactement une de leurs coordonnées : si x a les coordonnées $(x_0, x_1, \dots, x_i, \dots, x_{d-1})$, alors un voisin y dans la dimension i a les coordonnées $(y_0, y_1, \dots, y_i, \dots, y_{d-1})$ où $y_i = (x_i \pm 1)$ (modulo n_i pour la grille torique) et $x_j = y_j, \forall j \neq i$. Cette arête est notée $[x, y]$ ou $[(x_0, x_1, \dots, x_{d-1}), (y_0, y_1, \dots, y_{d-1})]$.
- On appellera *axe i* , $\forall i \in [0, d-1]$, le graphe partiel de $TM(n_0, n_1, \dots, n_{d-1})$ dont les sommets ont les coordonnées $(0, 0, \dots, x_i, \dots, 0)$, avec $x_i \in [0, n_i]$, et dont les arêtes sont celles de $TM(n_0, n_1, \dots, n_{d-1})$ reliant ces sommets entre eux. L'axe i est donc un cycle de longueur n_i .
- On dit qu'une arête *suit* la dimension i si elle relie des sommets de coordonnées $(x_0, x_1, \dots, x_{d-1})$ et $(x_0, x_1, \dots, x_{d-1}) \pm \vec{e}_i$, où \vec{e}_i est le vecteur canonique $(0, 0, \dots, x_i = 1, \dots, 0)$.
- La grille torique étant un graphe sommet-transitif, on peut supposer sans perte de généralité que le sommet initiateur de la communication est le sommet de coordonnées $(0, 0, \dots, 0)$, que l'on appellera O .
- On décide d'étiqueter les arbres avec les entiers de 0 à $d-1$. L'arbre étiqueté i sera noté T_i ; on dira aussi que l'arbre est de *couleur i* . On notera $h(T_i)$ la profondeur de l'arbre T_i .
- Soit $\mathcal{T} = \{T_i \mid i \in [0, d-1]\}$, un ensemble de d arbres couvrants et deux à deux arête-disjoints de $TM(n_0, n_1, \dots, n_{d-1})$, on note $h(\mathcal{T})$ le maximum des $h(T_i)$, avec $T_i \in \mathcal{T}$. De même, on notera $h(TM(n_0, n_1, \dots, n_{d-1}))$ le minimum des $h(\mathcal{T})$, pour toutes les familles \mathcal{T} d'arbres couvrants et deux à deux arête-disjoints de $TM(n_0, n_1, \dots, n_{d-1})$.

3 Bornes inférieures

Avant d'introduire les résultats correspondant au cas bidimensionnel, on va établir le nombre maximum d'arbres couvrants arête-disjoints que l'on peut espérer construire et essayer de donner une borne inférieure sur leur profondeur.

Théorème 3.1 (Kundu, [23]) *Si G est un graphe non orienté d'arête-connexité $\lambda(G) = 2t$, alors il existe au moins t arbres couvrants arête-disjoints dans G .*

REMARQUE. Quand il s'agit d'arbres couvrant un graphe non orienté, il n'est pas nécessaire de préciser qu'ils sont de même racine (tout sommet peut être considéré comme racine de tous les arbres simultanément). Ce n'est plus vrai quand on s'intéresse à des graphes orientés, même symétriques : par exemple, il peut exister un sommet qui ne soit que feuille d'un des arbres orientés (par un arc entrant).

On déduit du théorème 3.1 le corollaire suivant, qui donne le nombre exact d'arbres dans le cas de la grille torique :

Corollaire 3.2 *Il existe exactement d arbres couvrants arête-disjoints dans une grille torique de dimension d .*

PREUVE. L'arête-connexité de $TM(n_0, n_1, \dots, n_{d-1})$ est $\Delta = 2d$. Le théorème 3.1 donne donc une borne inférieure égale à d . Par ailleurs, une grille torique de dimension d possède Nd arêtes. Or, par définition, un arbre couvrant les N sommets de cette grille torique possède $N - 1$ arêtes. Une borne supérieure évidente sur le nombre d'arbres est donnée par $\lfloor \frac{Nd}{N-1} \rfloor = d$ (car $N > d > 1$). Donc, il existe exactement d arbres couvrants arête-disjoints dans une grille torique de dimension d . \square

Le lemme suivant permet d'établir une borne inférieure immédiate sur la profondeur des arbres (proposition 3.4).

Lemme 3.3 *Soit $TM(n_0, n_1, \dots, n_{d-1})$ un tore de dimension d . Si p est le nombre de n_i pairs, $i \in [0, d-1]$, alors il existe exactement 2^{d-p} sommets à distance D d'un sommet quelconque de $TM(n_0, n_1, \dots, n_{d-1})$.*

PREUVE. Soient x et y , tels que $\vec{x} = (x_0, x_1, \dots, x_{d-1})$ et $\vec{y} = (y_0, y_1, \dots, y_{d-1})$, deux sommets quelconques du tore. Par définition, on a $y_i = (x_i + d_i) \bmod d, \forall i \in [0, d-1]$, avec $\sum_{i=0}^{d-1} |d_i| = d(x, y)$. Si on veut que $d(x, y)$ soit maximale, il faut que $\sum_{i=0}^{d-1} |d_i| = D$, donc il faut maximiser $|d_i|, \forall i \in [0, d-1]$. Or, pour tout $i \in [0, d-1]$, $|d_i|$ est maximal quand il vaut $\lfloor \frac{n_i}{2} \rfloor$. Quand la dimension i est d'ordre pair (c'est-à-dire si n_i est pair), $x_i + \lfloor \frac{n_i}{2} \rfloor = x_i - \lfloor \frac{n_i}{2} \rfloor$ (modulo d). Donc, en suivant la dimension i , on n'a qu'un seul choix pour se diriger vers les sommets à distance D . Au contraire, quand n_i est impair, $x_i + \lfloor \frac{n_i}{2} \rfloor \neq x_i - \lfloor \frac{n_i}{2} \rfloor$ (toujours modulo d), ce qui double les possibilités. Donc, si $d - p$ représente le nombre de dimensions d'ordre impair dans un tore de dimension d , on a bien 2^{d-p} sommets à distance D . \square

Proposition 3.4 *Soit la fonction \mathcal{W} satisfaisant $\mathcal{W}(x) \cdot e^{\mathcal{W}(x)} = x$, alors si $0 \leq p \leq d - \lfloor \frac{\mathcal{W}(2 \ln(2)(d+1))}{\ln(2)} \rfloor$, on ne peut pas construire d arbres couvrants arête-disjoints de profondeur D .*

PREUVE. D'après le lemme 3.3, il y a 2^{d-p} sommets à distance D d'un sommet quelconque du tore. Ils sont reliés entre eux par $(d-p) 2^{d-p-1}$ arêtes qui ne peuvent pas être utilisées par des arbres de profondeur D (sinon, la profondeur devient $D+1$). Donc, seules $dN - (d-p) 2^{d-p-1}$ arêtes peuvent être utilisées pour la construction d'arbres de profondeur D , alors que $d(N-1) = dN - d$ arêtes sont requises pour construire d arbres couvrants arête-disjoints. Donc, quand $(d-p) 2^{d-p-1} \geq d+1$, il existe au moins un arbre de profondeur au moins égale à $D+1$, c'est-à-dire quand $0 \leq p \leq d - \lfloor \frac{\mathcal{W}(2 \ln(2)(d+1))}{\ln(2)} \rfloor$. \square

Par exemple, dans le cas bidimensionnel, on a :

Corollaire 3.5 $h(TM(n)^2) \geq n$

PREUVE. D'après la proposition 3.4, pour $d = 2$ et $p = 0$, c'est-à-dire si n est impair, l'un au moins des arbres est de profondeur au moins $D+1$. Toujours d'après cette proposition, si n est pair, donc si $p = 2$, rien ne s'oppose à ce que tous les arbres soient de profondeur D . Or, dans $TM(n)^2$, $D = n$ quand n est pair et $D = n-1$ quand n est impair. Donc, on a bien $h(TM(n)^2) \geq n$. \square

La proposition 3.4 montre qu'il suffit de très peu de dimensions d'ordre impair pour empêcher la construction d'arbres de profondeur D . Je pense qu'on peut affiner cette borne inférieure par des considérations assez compliquées sur la configuration et la répartition des arêtes critiques entre les arbres, en procédant par élimination. C'est pourquoi je propose la conjecture suivante :

Conjecture 3.6 *La borne inférieure sur la profondeur de d arbres couvrants arête-disjoints d'un tore de dimension d est D , si $p = d$, et de l'ordre de $D + \omega(\ln(d - p))$, si $p < d$.*

Cas orienté symétrique

Lorsqu'on s'intéresse au mode *full-duplex*, on considère la version orientée symétrique G^* du graphe G représentant le réseau. A la place du théorème 3.1, on utilise un théorème similaire d'Edmonds [13], qui borne le nombre d'arbres par l'arc-connexité du graphe. C'est ce qui permet d'affirmer qu'il existe $2d$ arbres couvrants, deux à deux arc-disjoints et de même racine, dans une grille torique orientée symétrique de dimension d .

Dans [25], Michallon, Trystram et Villard donnent une construction de 4 arbres de profondeur optimale $D + 1$ dans $TM^*(n)^2$. Ce résultat a depuis été généralisé aux grilles toriques $TM^*(n)^3$ et $TM^*(n_0, n_1)$ par Michallon et Trystram, dans [24].

Plus généralement, Fraigniaud et Laforest montrent dans [16] qu'il est possible de construire une famille de $2d$ arbres de profondeur $D + \max_{i=0}^{d-1} (\lfloor \frac{n_i}{2} \rfloor) - 1$ dans $TM^*(n_0, n_1, \dots, n_{d-1})$.

4 Cas bidimensionnel

Dans le cas des grilles toriques bidimensionnelles, on cherche donc à construire deux arbres couvrants arête-disjoints.

REMARQUE. Afin de rester en conformité avec les articles cités et de bien faire ressortir les symétries des constructions, dans les figures, l'origine de la diffusion est placée au centre des représentations de la grille torique.

Saad & Schultz, 1989 [27]

La figure 2 montre deux arbres dans $TM(5)^2$. La construction se généralise de manière évidente dans le cas d'un tore $TM(n_0, n_1)$ et on peut voir que les arbres s'obtiennent facilement l'un de l'autre par une simple rotation de $\frac{\pi}{2}$. Cependant, même s'ils sont de profondeur optimale D , ce ne sont pas des arbres couvrants.

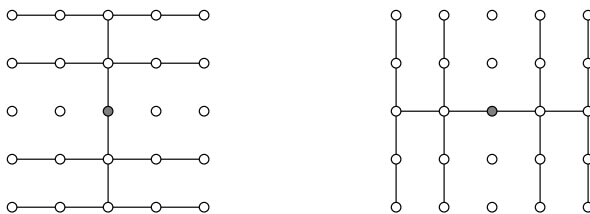


FIG. 2 – Arbres de Saad & Schultz dans $TM(5, 5)$

Pour atteindre le temps optimal $b(L, D, 2) = \left(\sqrt{(D-1)\beta} + \sqrt{\frac{L\tau}{2}} \right)^2$, les auteurs proposent de compléter la diffusion en pipe-plant les paquets manquants pendant le vidage du pipe-line. Mais dans [4], il est montré que cela ne peut être réalisé que pour des messages de taille inférieure ou égale à $\frac{\beta}{2\tau} \left\lfloor \frac{D^2}{D-1} \right\rfloor$ et que pour des messages plus longs, la diffusion est achevée en :

$$b(L, D, 2) + \frac{L\tau}{2} + \sqrt{\frac{(D-1)\beta L\tau}{2}} - \frac{D}{2} \left(\beta + \sqrt{\frac{\beta L\tau}{2(D-1)}} \right)$$

Simmen, 1991 [29]

Dans la figure 3, on peut voir les arbres de Simmen dans $TM(5)^2$. Ce sont des arbres couvrants mais pour un tore $TM(n_0, n_1)$, leur profondeur est $n_0 + n_1 - 1$, soit de l'ordre de $2D$ et ils sont moins faciles à construire (symétrie par rapport à la diagonale montante). Cependant, ces arbres peuvent être utilisés pour construire 4 arbres couvrants arc-disjoints dans la grille torique orientée symétrique, en les associant à leur symétrique par rapport à la diagonale descendante.

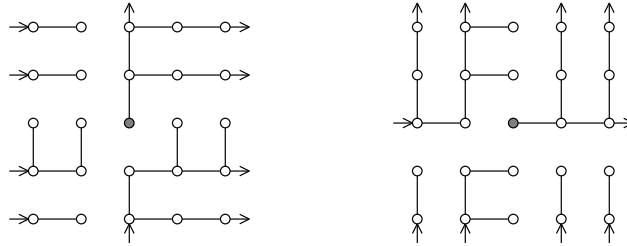


FIG. 3 – Arbres de Simmen dans $TM(5, 5)$

On peut améliorer la profondeur de ces arbres comme je le montre dans la figure 4. La plus grande profondeur est alors $\max(\lfloor \frac{n_0}{2} \rfloor + n_1 - 1, \lfloor \frac{n_1}{2} \rfloor + n_0 - 1)$, c'est-à-dire de l'ordre de $\frac{3D}{2}$ pour un tore de côté n . On peut à nouveau les obtenir l'un de l'autre par une simple rotation de $\frac{\pi}{2}$, mais ils ne sont pas utilisables en *full-duplex*.

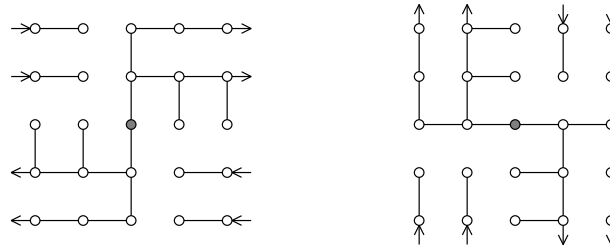


FIG. 4 – Arbres de Simmen modifiés dans $TM(5, 5)$

Bermond, Michallon & Trystram, 1992 [4]

Enfin, Bermond, Michallon et Trystram ont définitivement clos le cas bidimensionnel en construisant deux arbres de profondeur n dans un tore de côté n , ce qui est optimal d'après le corollaire 3.5. Une fois de plus, les arbres peuvent être obtenus l'un de l'autre par une rotation de $\frac{\pi}{2}$ comme le montre la figure 5 dans le cas du tore $TM(5)^2$.

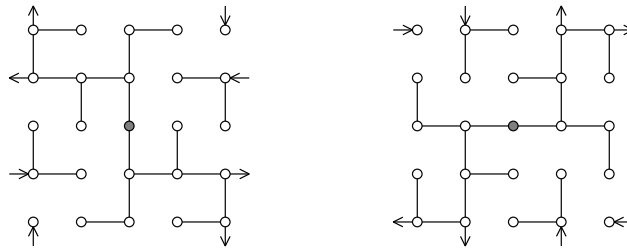
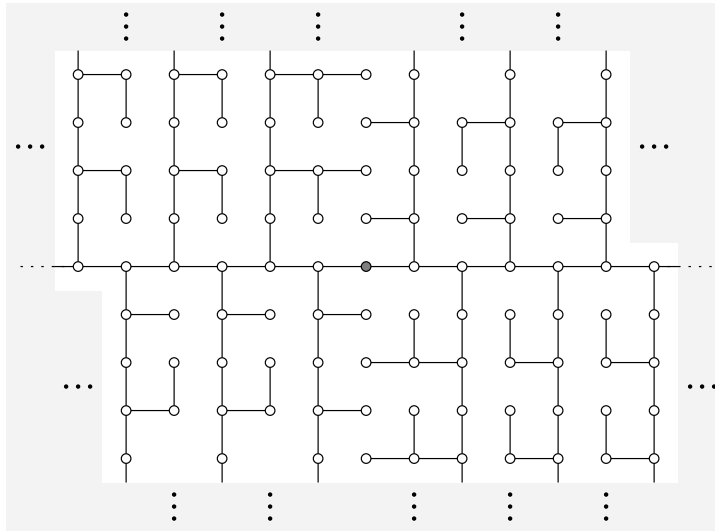


FIG. 5 – Arbres de Bermond, Michallon & Trystram dans $TM(5, 5)$

La figure 6 donne une idée de la construction générale pour $TM(n_0, n_1)$. Cette construction repose sur la division du tore bidimensionnel en 4 secteurs délimités par ses axes. Le détail des terminaisons des branches (zone grisée) dépend de la congruence de $n \bmod 4$ pour un tore de côté n . Il est donné dans [4] pour les cas $n = 10$, $n = 11$, $n = 12$ et $n = 13$. L'extension à des valeurs de n supérieures s'obtient en augmentant «de l'intérieur» ces constructions de base (c'est-à-dire en augmentant la construction générale de la figure 6, en conservant les mêmes terminaisons que la construction de base). Pour des tores $TM(n_0, n_1)$, les arbres peuvent être obtenus suivant le même principe.

FIG. 6 – Construction générale dans $TM(n_0, n_1)$

Cette construction permet donc d'affirmer le résultat suivant :

Théorème 4.1 (Bermond, Michallon, Trystram [4]) $h(TM(n)^2) = n$.

Mais si au moins n_0 ou n_1 est impair, alors au moins un des arbres de cette construction dans $TM(n_0, n_1)$ sera de profondeur $D + 1$. En effet, d'après la figure 6, les arêtes non utilisées (il y en a deux) appartiennent aux axes. Or, les arêtes critiques (celles qui ne peuvent pas être utilisées par des arbres de profondeur D) ne peuvent pas appartenir à un axe, puisqu'elles relient des sommets à distance D de l'origine, qui, par définition, ne sont pas sur les axes : ce sont des sommets (x_0, x_1) , tels que $x_i \neq 0, \forall i \in [0, 1]$.

5 Approche d -dimensionnelle

L'objectif était de trouver une construction relativement simple, dont on serait capable de déduire un algorithme de diffusion général efficace qui puisse être utilisé quels que soient la taille et le nombre de dimensions du tore. Ce but est atteint par une construction reposant sur une décomposition récursive particulière du tore d -dimensionnel, qu'on va répéter tant que $d > 2$ (voir figure 7 pour le cas tridimensionnel et figure 14 ou 15 pour le cas quadridimensionnel). Cette construction permet de réutiliser les résultats obtenus dans le cas bidimensionnel et de les étendre à une infinité de dimensions.

Le tore d -dimensionnel contient :

- Une grille d -dimensionnelle dont les sommets sont ceux ayant toutes leurs coordonnées différentes de 0. Il est clair que cette grille a $\prod_{i=0}^{d-1} (n_i - 1)$ sommets $(x_0, x_1, \dots, x_{d-1})$, tels que $x_i \neq 0, \forall i \in [0, d - 1]$.
- d sous-tores $(d - 1)$ -dimensionnels, dont les sommets $(x_0, x_1, \dots, x_{d-1})$ ont une même coordonnée x_i égale à 0.

REMARQUE. Une des difficultés vient du fait que ces sous-tores ne sont pas disjoints, puisqu'ils ont en commun des sous-tores $(d - 2)$ -dimensionnels.

Par la suite, on notera $STM(d_0, d_1, \dots, d_{i-1})$ le sous-tore i -dimensionnel dont les sommets ont pour coordonnées $(x_0, x_1, \dots, x_{d-1})$, avec $x_{d_j} = 0, \forall j \notin [0, i]$. Par exemple, $STM(0, 1)$ désignera le sous-tore bidimensionnel (ou 2-dimensionnel), dont les sommets ont leurs coordonnées $x_k = 0, \forall k \notin \{0, 1\}$. On utilisera la notation $SM(d_0, d_1, \dots, d_{i-1})$ pour désigner la sous-grille i -dimensionnelle incluse dans le sous-tore $STM(d_0, d_1, \dots, d_{i-1})$. Par exemple, la sous-grille $SM(0, 1, 2)$ contient les sommets $(x_0, x_1, \dots, x_{d-1})$, tels que $x_j \neq 0, \forall j \in \{0, 1, 2\}$, et $x_k = 0, \forall k \in [3, d - 1]$.

REMARQUE. Dans $STM(d_0, d_1, \dots, d_{i-1})$, s'il y a $k \geq 1$ dimensions $j \in [0, i]$, telles que $n_j = 2$, la sous-grille $SM(d_0, d_1, \dots, d_{i-1})$ n'est pas vraiment une grille i -dimensionnelle. C'est en fait une grille $(i-k)$ -dimensionnelle si $k < i-1$, une chaîne si $k = i-1$ et un sommet unique si $k = i$. On continuera cependant à l'appeler sous-grille i -dimensionnelle. De même, dans le cas tridimensionnel, les sous-tores bidimensionnels partagent des axes, qui ne sont pas à proprement parler des sous-tores $(d-2)$ -dimensionnels (on peut quand-même considérer qu'un cycle est un tore 1-dimensionnel).

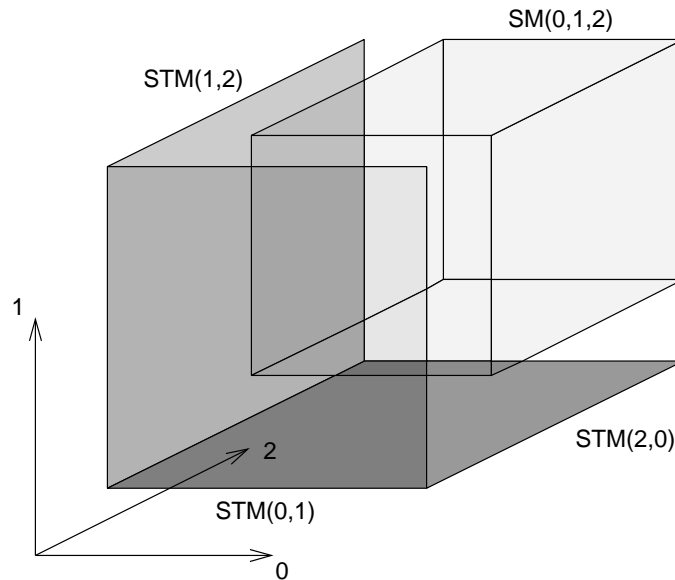


FIG. 7 – La sous-grille et les 3 sous-tores associés à $TM(n_0, n_1, n_2)$

REMARQUE. Pour la clarté des figures, j'ai choisi de placer l'origine de la diffusion sur le coin inférieur avant-gauche des représentations tridimensionnelles de la grille torique.

Une deuxième idée est d'utiliser des numéros de couleur en bijection avec les numéros de dimension et de faire en sorte que les arêtes d'un sous-tore $STM(d_0, d_1, \dots, d_{i-1})$ ne soient coloriées qu'avec les couleurs d_0, d_1, \dots, d_{i-1} . Autrement dit, l'arbre de couleur c n'a des arêtes dans $STM(d_0, d_1, \dots, d_{i-1})$ que si $c \in \{d_0, d_1, \dots, d_{i-1}\}$. Les sommets de $STM(d_0, d_1, \dots, d_{i-1})$ sont donc atteints par les arbres de couleur d_j ($j \in [0, i-1]$) avec des arêtes appartenant à $STM(d_0, d_1, \dots, d_{i-1})$ et par les arbres de couleur d_k ($k \notin [0, i-1]$) avec des arêtes de *raccord* qui n'appartiennent pas à $STM(d_0, d_1, \dots, d_{i-1})$. En particulier, on impose que les arêtes de l'axe d_i appartiennent à au plus un arbre, celui de couleur d_i . Plus exactement, on fera en sorte de fixer les $n_{d_i} - 1$ arêtes de l'axe d_i , utilisées par l'arbre de couleur d_i .

Dans les sous-tores bidimensionnels, la construction utilise deux arbres couvrants arête-disjoints, que l'on appellera *sous-arbres* des arbres construits dans le tore d -dimensionnel. Plus précisément, dans le sous-tore $STM(d_0, d_1)$, on plonge un arbre couvrant de couleur d_0 , qui utilise les $n_{d_0} - 1$ arêtes fixées de l'axe d_0 et aucune de l'axe d_1 , et un arbre couvrant de couleur d_1 , arête-disjoint du précédent, qui utilise les $n_{d_1} - 1$ arêtes fixées de l'axe d_1 et aucune de l'axe d_0 . Nous verrons dans le calcul de la profondeur des arbres obtenus à l'issue de la construction, que le choix de ces sous-arbres est important, en particulier dans le cas où les dimensions sont d'ordres différents, et que l'on a intérêt à minimiser la profondeur de l'un d'eux. Aussi différencierons-nous ces deux sous-arbres en les appelant respectivement *de type A* et *de type B*. Dans le sous-tore $STM(d_0, d_1)$, on aura par exemple le sous-arbre de couleur d_0 de type A et le sous-arbre de couleur d_1 de type B (ou l'inverse). On utilisera la notation condensée $A_{d_0}^{d_0, d_1} B_{d_1}^{d_0, d_1}$ pour indiquer que le sous-arbre de couleur d_0 dans $STM(d_0, d_1)$ est de type A et que celui de couleur d_1 est de type B.

REMARQUE. Pour ne pas surcharger les figures, on ne mentionne pas les dimensions du sous-tore qui figurent en exposants de A et de B dans la notation formelle. Il faut cependant garder à l'esprit que pour $k \neq c$, $k' \neq c$ et $k \neq k'$, le sous-arbre A_c de $STM(c, k)$ est différent du sous-arbre A_c de $STM(c, k')$.

On notera que les arbres issus de toutes les constructions présentées dans la section 4, à l'exception de [27] qui ne donne pas des arbres couvrants, satisfont toutes les conditions énumérées ci-dessus. En particulier, on impose qu'un arbre de type A_i ou B_i , avec $i \in \{d_0, d_1\}$, utilise les mêmes arêtes de l'axe i dans $TM(n_{d_0}, n_{d_1})$, quels que soient n_{d_0} et n_{d_1} .

6 Cas tridimensionnel

De manière à rendre les choses plus claires, je présente d'abord le cas $d = 3$.

Théorème 6.1 *Supposons $n_0 \leq n_1 \leq n_2$. Alors :*

$$h(TM(n_0, n_1, n_2)) \leq h(TM(n_0, n_1)) + n_2$$

PREUVE. Dans le cas tridimensionnel, il y a 3 sous-tores bidimensionnels $STM(0, 1)$, $STM(0, 2)$ et $STM(1, 2)$ et une sous-grille $SM(0, 1, 2)$, que l'on appellera plus simplement SM . Les couleurs et les dimensions étant numérotées de 0 à 2, les numéros de couleur et de dimension seront pris modulo 3. De plus, afin de faire ressortir la symétrie de la construction, on notera les sous-tores $STM(c, c + 1)$, c'est-à-dire que le sous-tore $STM(0, 2)$ sera noté $STM(2, 0)$, car $2 + 1 \equiv 0 \pmod{3}$.

Conformément à la section 5, dans chaque sous-tore $STM(c, c + 1)$, $\forall c \in [0, 2]$, on plonge deux arbres arête-disjoints : $A_c^{c, c+1}$ et $B_{c+1}^{c, c+1}$ (abrégiés en A_c et B_{c+1} , les exposants étant dans ce cas redondants). $STM(0, 1)$ contient donc les sous-arbres A_0 et B_1 , $STM(1, 2)$ les sous-arbres A_1 et B_2 , et $STM(2, 0)$ les sous-arbres A_2 et B_0 (voir figure 8).

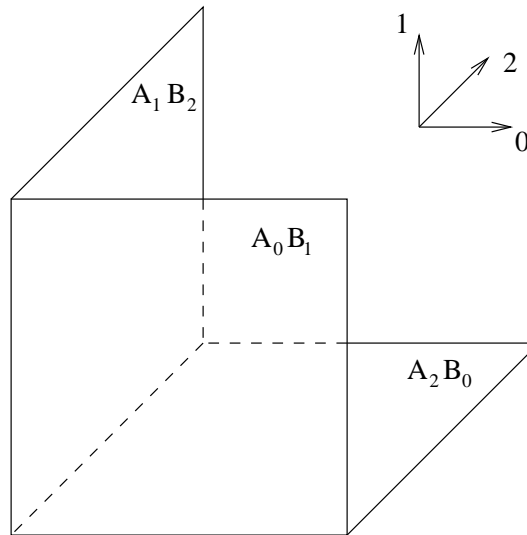


FIG. 8 – Répartition des sous-arbres dans les sous-tores en trois dimensions

Remarquons que sur chaque axe c , $\forall c \in [0, 2]$, ce sont exactement les mêmes arêtes qui sont coloriées avec la couleur c , une fois par le sous-arbre A_c dans le sous-tore $STM(c, c + 1)$ et une autre fois par le sous-arbre B_c dans le sous-tore $STM(c - 1, c)$. Il n'y a pas de conflit, car on a pris soin d'imposer que pour tout $k \neq c$, les sous-arbres de couleur c des sous-tores $STM(c, k)$ utilisent toujours les mêmes arêtes sur l'axe c et aucune arête de l'axe k .

En revanche, pour chaque couleur c , il existe donc un sous-tore dans lequel on n'a pas pu plonger de sous-arbre de couleur c , donc contenant des sommets qui n'ont pas été atteints par cette couleur : c'est le sous-tore $STM(c + 1, c - 1)$. Nous allons colorier ces sommets par des arêtes issues de la sous-grille SM : les arêtes de raccord. Pour cela, il faut donc que tous les sommets de la sous-grille aient été auparavant atteints par toutes les couleurs.

On décide donc d'attribuer à l'arbre T_c toutes les chaînes de la sous-grille SM parallèles à l'axe de la dimension $c+2 = c-1$ (voir figure 9 ou figure 11 pour le cas $c = 0$). Les chaînes d'une même couleur c sont reliées au reste de la construction par les arêtes de la dimension $c-1$ issues des sommets (x_0, x_1, x_2) , tels que $x_{c-1} = 0$, $x_c \neq 0$ et $x_{c+1} \neq 0$, atteints par le sous-arbre de couleur c du sous-tore ne contenant pas la dimension $c-1$, c'est-à-dire $STM(c, c+1)$.

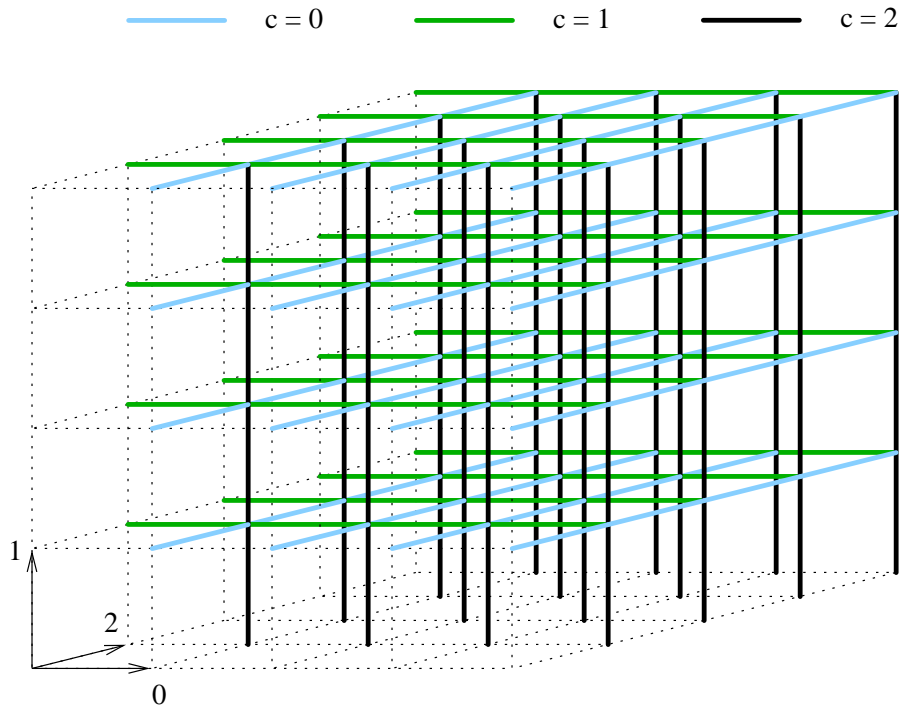


FIG. 9 – Diffusion dans la sous-grille en trois dimensions

Enfin, à partir des sommets (x_0, x_1, x_2) de SM , tels que $x_c = n_c - 1$, on atteint les sommets «orphelins» de la couleur c dans $STM(c+1, c-1)$, par une arête de raccord de la dimension c (voir figure 10).

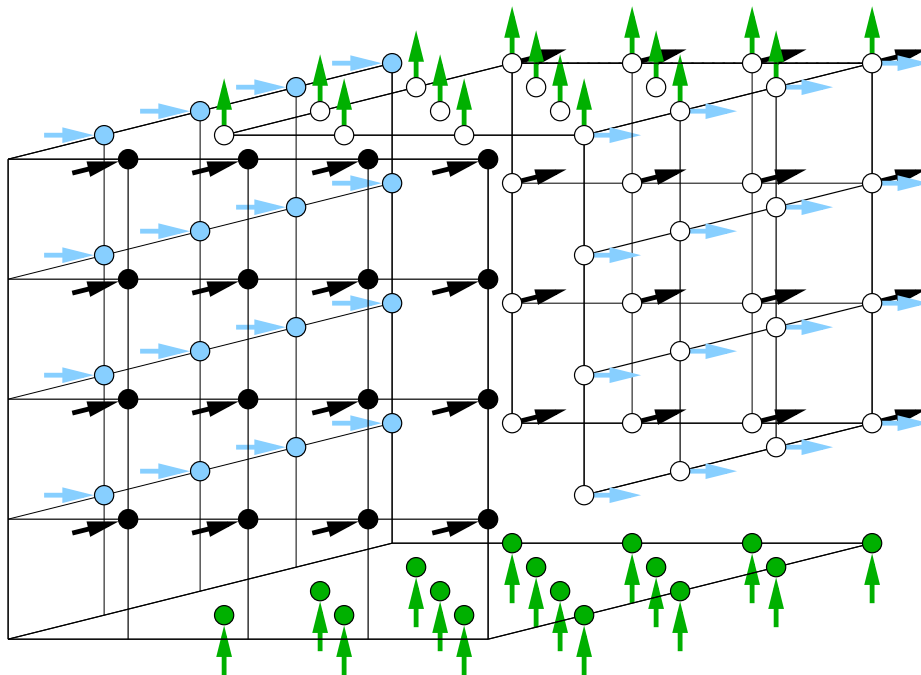


FIG. 10 – Arêtes de raccord en trois dimensions

La figure 11 représente l'arbre T_0 obtenu dans $TM(5)^3$, en prenant les arbres de la figure 5 pour A_0 et B_0 .

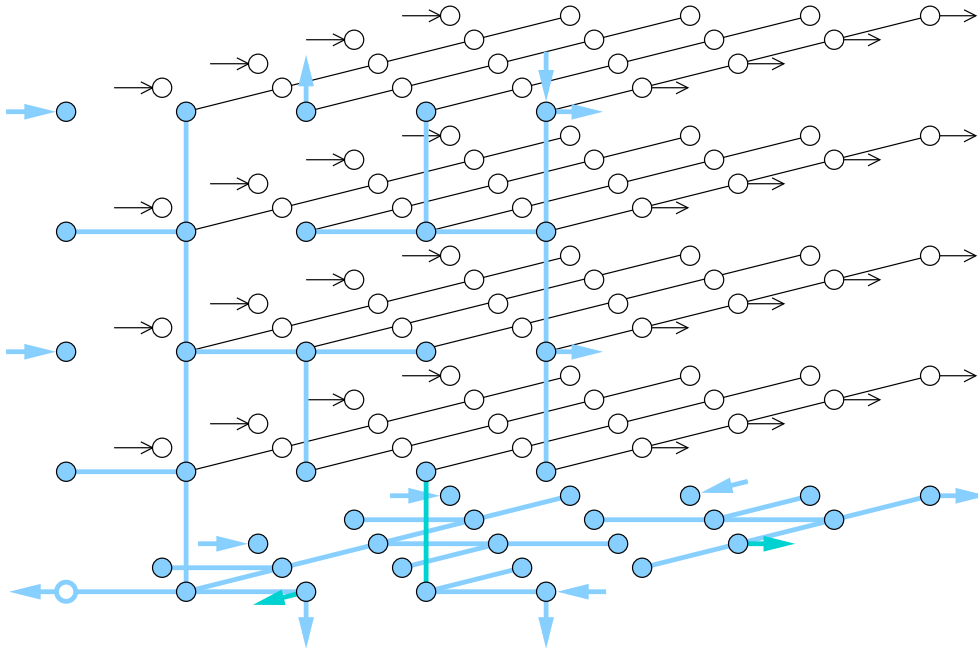


FIG. 11 – Arbre $T_{c=0}$ de profondeur $2n = 10$ dans $TM(n = 5)^3$

REMARQUE. De façon à améliorer la lisibilité de la figure, les chaînes de diffusion dans la sous-grille SM et les arêtes de raccord sont en traits fins, par opposition aux arêtes des sous-arbres A et B .

Montrons que T_c est bien un arbre couvrant : par construction, on atteint tous les sommets de $STM(c, c + 1)$ par le sous-arbre A_c et ceux de $STM(c - 1, c)$ par le sous-arbre B_c ; ensuite, on atteint tous les sommets de la sous-grille SM par les chaînes de la dimension $c - 1$ issues de $STM(c, c + 1)$ et enfin les sommets non encore atteints de $STM(c + 1, c - 1)$ par les arêtes de raccord de la dimension c . Tous les sommets du tore sont atteints et on a bien un arbre (on n'a pas créé de cycle ; on peut aussi compter les arêtes).

De même, il est facile de montrer que T_0, T_1 et T_2 sont bien deux à deux arête-disjoints. En effet, on a imposé que seul l'arbre de couleur c pouvait utiliser des arêtes de l'axe c , et que c'était toujours les mêmes. Dans $STM(c, c + 1)$, les sous-arbres A_c et B_{c+1} sont arête-disjoints par définition. Enfin, pour T_c , on utilise exclusivement des arêtes de la dimension $c - 1$ dans SM et des arêtes de la dimension c pour les raccords.

La profondeur de l'arbre T_c est donnée par $h(T_c) = \max(h(B_c), h(A_c) + n_{c-1})$. En effet, les feuilles de T_c dont la profondeur est maximale sont soit les feuilles du sous-arbre B_c , soit extrémités des chaînes de diffusion dans SM , c'est-à-dire à distance au plus $h(A_c) + n_{c-1} - 1$, soit les sommets de $STM(c + 1, c - 1)$ atteints par les arêtes de raccord, donc à distance au plus $h(A_c) + n_{c-1}$. Or, quelle que soit la construction choisie parmi celles de la section 4 pour les sous-arbres de type A et B (hormis les arbres de [27]), on a toujours : $h(B_c) \leq h(A_c) + n_{c-1}$, $\forall c \in [0, 2]$, et $h(TM(n_1, n_2)) + n_0 \leq h(TM(n_0, n_2)) + n_1 \leq h(TM(n_0, n_1)) + n_2$ si $n_0 \leq n_1 \leq n_2$. \square

Proposition 6.2 $TM(n)^3$ admet trois arbres couvrants arête-disjoints de profondeur $2n - 1$.

PREUVE. Notons $\mathcal{T}_{\text{BMT}}^3$, l'ensemble des arbres T_0, T_1 et T_2 obtenus dans le tore tridimensionnel $TM(n_0, n_1, n_2)$, en prenant les arbres de [4] pour A et B . En première approximation, si $n_0 \leq n_1 \leq n_2$, on a : $h(\mathcal{T}_{\text{BMT}}^3) = \lfloor \frac{n_0}{2} \rfloor + \lfloor \frac{n_1}{2} \rfloor + n_2$ (+1 si au moins n_0 ou n_1 est impair). Or, on peut raffiner le calcul de la profondeur de l'arbre T_c en remarquant que ce n'est pas la profondeur du sous-arbre A_c qui est déterminante, mais la distance de certains de ses sommets par rapport à l'origine. En effet, considérons les sommets origines des chaînes de

diffusion dans la sous-grille SM (on traite à part ceux dont les chaînes engendrent les raccords). La profondeur de l'arbre T_c peut être calculée à partir de la distance de ces sommets par rapport à l'origine O :

Soient $m_c = \max_{x \in X} (d(O, x))$ et $m'_c = \max_{x' \in X'} (d(O, x'))$, avec :

$$X = \{ (x_0, x_1, x_2) \mid x_{c-1} = 0, x_{c+1} \neq 0, x_c = n_c - 1 \}$$

$$X' = \{ (x_0, x_1, x_2) \mid x_{c-1} = 0, x_{c+1} \neq 0, 0 < x_c < n_c - 1 \}$$

alors : $h(T_c) = \max(h(B_c), \max(m_c + n_c - 1, m'_c + n_{c-1} - 1))$.

En prenant les arbres de [4] pour A_c et B_c dans $TM(n)^3$, on trouve alors une profondeur de $2n$ pour $n \leq 10$ et de $2n - 1$ pour $n > 10$. En effet, pour $n \leq 10$, il existe des sommets dans X et dans X' à distance n de l'origine. On a :

$$\forall c \in [0, d - 1], \quad h(T_c) = \max(n, \max(2n, 2n - 1)) = 2n$$

Pour les autres valeurs de n , les sommets de X sont au plus à distance $n - 2$ de l'origine, mais il existe toujours des sommets à distance n dans X' . On a alors :

$$\forall c \in [0, d - 1], \quad h(T_c) = \max(n, \max(2n - 2, 2n - 1)) = 2n - 1$$

Or, on peut construire dans $STM(c, c + 1)$ un sous-arbre de couleur c de profondeur plus grande que celui obtenu avec [4], mais donnant un meilleur résultat en profondeur globale (voir figure 12).

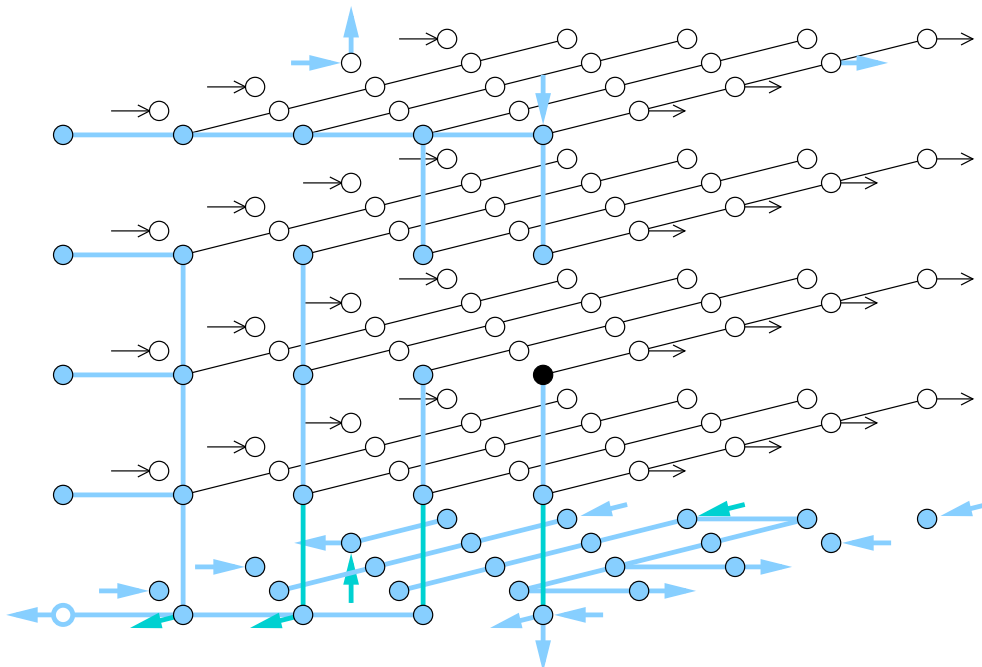
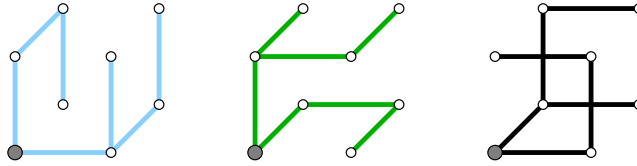


FIG. 12 - Arbre $T_{c=0}$ de profondeur $2n - 1 = 9$ dans $TM(n = 5)^3$

REMARQUE. Cette construction est particulière au tore tridimensionnel : les arêtes de $STM(c - 1, c)$ utilisée par l'arbre T_c ne constituent pas un sous-arbre valable en 2 dimensions.

La profondeur du sous-arbre de T_c dans $STM(c, c + 1)$ est $n + 1$ et il existe un sommet de T_c dans $STM(c - 1, c)$ à distance $2n - 1$! Mais pour tout $n > 3$, le sommet de X le plus éloigné de l'origine dans $STM(c, c + 1)$ est à distance $n - 2$ (il est indiqué en noir sur la figure 12). Pour $n = 3$ ou $n = 2$, il est à distance 2. Il existe bien sûr des sommets de X' à distance n de l'origine, ce qui donne une profondeur globale de $2n - 1$, pour $n > 2$. Pour le cas $n = 2$, la figure 13 donne les trois arbres de profondeur optimale $2n - 1 = d = 3$. \square


 FIG. 13 – T_0 , T_1 et T_2 de profondeur optimale 3 dans $TM(2)^3$

Proposition 6.3 Si $n_0 \leq n_1 \leq n_2$, alors $TM(n_0, n_1, n_2)$ admet trois arbres couvrants arête-disjoints de profondeur : $\frac{n_0}{2} + \frac{n_1}{2} + n_2 - 1$, si n_0 et n_1 sont pairs et supérieurs à 2, et $\lfloor \frac{n_0}{2} \rfloor + \lfloor \frac{n_1}{2} \rfloor + n_2$, sinon.

PREUVE. Dans $TM(n)^2$, on peut modifier les arbres de [4] de façon à minimiser la distance des sommets de X par rapport à l'origine, tout en gardant les sommets de X' à distance inférieure ou égale à n , quel que soit $n \leq 10$. Ces modifications se font au cas par cas et il est sans intérêt de les énumérer ici. Une adaptation de ces modifications est possible dans $TM(n_0, n_1)$. En reprenant les considérations sur la profondeur des arbres en fonction de la distance des sommets de X par rapport à l'origine et celles de la preuve du théorème 6.1, on trouve : si $n_0 \leq n_1 \leq n_2$, alors $h(\mathcal{T}_{\text{BMT}}^3) = \lfloor \frac{n_0}{2} \rfloor + \lfloor \frac{n_1}{2} \rfloor + n_2$ (-1 si n_0 et n_1 sont pairs). \square

7 Généralisation

Théorème 7.1 Pour $d > 2$, si on suppose $n_0 \leq n_1 \leq \dots \leq n_{d-1}$, alors :

$$h(TM(n_0, n_1, \dots, n_{d-1})) \leq h(TM(n_0, n_1)) + \sum_{i=2}^{d-1} (n_i - 1) + 1$$

PREUVE. Les arbres sont construits par l'algorithme \mathcal{A} suivant :

- ◇ **Étape 1 :** On construit directement les sous-arbres dans les $\frac{d(d-1)}{2}$ sous-tore bidimensionnels, comme expliqué dans l'approche générale (section 5). Plus exactement, dans $STM(d_0, d_1)$, avec $d_0 < d_1$, on plonge :
 - un sous-arbre $A_{d_0}^{d_0, d_1}$ et un sous-arbre $B_{d_1}^{d_0, d_1}$ si $(d_1 - d_0) \leq \lfloor \frac{d}{2} \rfloor$;
 - un sous-arbre $A_{d_1}^{d_0, d_1}$ et un sous-arbre $B_{d_0}^{d_0, d_1}$ sinon.

EXEMPLES.

- Dans le sous-tore $STM(0, 3)$, on aura les sous-arbres A_0B_3 , si $d \geq 6$, mais A_3B_0 , si $d = 4$ ou 5 .
- Dans le cas tridimensionnel, on retrouve bien A_0B_1 dans $STM(0, 1)$, A_1B_2 dans $STM(1, 2)$ et A_2B_0 dans $STM(0, 2)$ (voir figure 8).
- Dans le cas quadridimensionnel (voir figure 14 ou figure 15), on obtient A_0B_1 dans $STM(0, 1)$, A_0B_2 dans $STM(0, 2)$, A_3B_0 dans $STM(0, 3)$, A_1B_2 dans $STM(1, 2)$, A_1B_3 dans $STM(1, 3)$ et A_2B_3 dans $STM(2, 3)$.

REMARQUE. On omet d'indiquer les dimensions qui sont en exposant de A et B dans la notation formelle des sous-arbres, car elles sont redondantes dans la notation condensée A_aB_b , qui désigne implicitement le sous-tore $STM(a, b) = STM(b, a)$.

- ◇ **Étape i :** A l'étape i , $i \in [2, d-1]$, pour tout $k \in [0, i]$, on poursuit la construction des arbres dans le sous-tore $(i+1)$ -dimensionnel $STM(d_0, d_1, \dots, d_k, \dots, d_i)$, dans lequel on va construire des sous-arbres de T_{d_k} (rappelons que les autres arbres ne possèdent aucune arête dans ce sous-tore). Par hypothèse, pour tout $k \in [0, i]$, on a construit à l'étape $i-1$ les sous-arbres dans les sous-tore i -dimensionnels $STM(d_0, d_1, \dots, d_{k-1}, d_{k+1}, \dots, d_i)$. A l'étape i , on effectue alors les deux opérations suivantes, pour tout $k \in [0, i]$:
 - On atteint les sommets de la sous-grille $SM(d_0, d_1, \dots, d_k, \dots, d_i)$ avec la couleur $d_{(k+1) \bmod i+1}$, en coloriant les chaînes de la dimensions d_k issues des sommets de $STM(d_0, d_1, \dots, d_{k-1}, d_{k+1}, \dots, d_i)$ qui ne sont pas dans un sous-tore $(i-2)$ -dimensionnel, c'est-à-dire les sommets $(x_0, x_1, \dots, x_{d-1})$, tels que $x_{d_j} \neq 0$, si $j \in [0, i] \setminus \{k\}$, et $x_{d_j} = 0$ sinon.

- Ensuite, on atteint les sommets $(x_0, x_1, \dots, x_{d-1})$ de $STM(d_0, d_1, \dots, d_{k-1}, d_{k+1}, \dots, d_i)$ définis ci-dessus avec la couleur d_k , en utilisant les arêtes de raccord de la dimension d_k issues des sommets $(y_0, y_1, \dots, y_{d-1})$ de $SM(d_0, d_1, \dots, d_k, \dots, d_i)$, tels que $y_{d_k} = n_{d_k} - 1$.

EXEMPLES.

- Pour $d = 4$, on construit à l'étape 2 les arbres de couleur d_0, d_1 et d_2 dans $STM(d_0, d_1, d_2)$. Ainsi, si $d_0 = 0, d_1 = 2$ et $d_2 = 3$, on travaille dans $STM(0, 2, 3)$:
 - pour $k = 0$: à partir des sommets $(0, 0, x_2, x_3)$ de $STM(2, 3)$ vérifiant $x_2 \neq 0$ et $x_3 \neq 0$, on colorie les chaînes de la dimension 0 avec la couleur 2 ;
 - pour $k = 1$: à partir des sommets $(x_0, 0, 0, x_3)$ de $STM(0, 3)$ vérifiant $x_0 \neq 0$ et $x_3 \neq 0$, on colorie les chaînes de la dimension 2 avec la couleur 3 ;
 - pour $k = 2$: à partir des sommets $(x_0, 0, x_2, 0)$ de $STM(0, 2)$ vérifiant $x_0 \neq 0$ et $x_2 \neq 0$, on colorie les chaînes de la dimension 3 avec la couleur 0.

On a donc colorié la sous-grille $SM(0, 2, 3)$. On termine le coloriage de $STM(0, 2, 3)$ en rajoutant les arêtes de raccord :

- les arêtes reliant les sommets $(n_0 - 1, 0, x_2 \neq 0, x_3 \neq 0)$ et les sommets $(0, 0, x_2, x_3)$ dans la dimension 0 seront de couleur 0 ;
- les arêtes reliant les sommets $(x_0 \neq 0, 0, n_2 - 1, x_3 \neq 0)$ et les sommets $(x_0, 0, 0, x_3)$ dans la dimension 2 seront de couleur 2 ;
- les arêtes reliant les sommets $(x_0 \neq 0, 0, x_2 \neq 0, n_3 - 1)$ et les sommets $(x_0, 0, x_2, 0)$ dans la dimension 3 seront de couleur 3.
- Observons comment, pour $d = 4$, tous les sommets du tore sont bien atteints par l'arbre de couleur 0 :
 - Les sommets de l'axe 0 sont atteints par les $n_0 - 1$ arêtes de l'axe 0 fixées par la construction des sous-arbres de type A et B de couleur 0 dans les sous-tores bidimensionnels $STM(0, k)$.
 - Les sommets des sous-tores bidimensionnels $STM(0, k)$ sont atteints par le sous-arbre de couleur 0 de l'étape 1, à savoir A_0 , pour $STM(0, 1)$ et $STM(0, 2)$, et B_0 pour $STM(0, 3)$.
 - Les sommets des sous-tores $STM(k, k')$, avec $k \neq k'$, sont atteints à l'étape 2 par les arêtes de raccord venant de sommets de $SM(0, k, k')$, qui eux aussi sont atteints à l'étape 2.
 - Les sommets de la sous-grille $SM(0, 1, 2, 3)$ et ceux du sous-tore $STM(1, 2, 3)$ sont atteints lors de l'étape 3.

On peut aussi exprimer la construction des arbres de façon récursive, en considérant une procédure récursive `Construire_Arbres(Ordres, Indices)`, où `Ordres` est la liste des $n_i, \forall i \in [0, d - 1]$, de $TM(n_0, n_1, \dots, n_{d-1})$ et `Indices` est une liste de i indices distincts ($2 \leq i \leq d - 1$) des éléments de `Ordres`, qui construit i arbres couvrants arête-disjoints dans $STM(d_0, d_1, \dots, d_{i-1})$ si `Indices`=(d_0, d_1, \dots, d_{i-1}).

- Si `Indices` ne contient que deux éléments $d_0 < d_1$, alors on plonge A_{d_0} et B_{d_1} dans $STM(d_0, d_1)$, si $(d_1 - d_0) \leq \lfloor \frac{d}{2} \rfloor$, et A_{d_1} et B_{d_0} sinon.
- Si `Indices` possède plus de deux éléments, alors, pour chaque élément $d_k, \forall k \in [0, i - 1]$:
 - appeler `Construire_Arbres(Ordres, Indices_2)` où `Indices_2` est la liste `Indices` privée de l'élément d_k ;
 - ajouter les chaînes de diffusion de la dimension d_k à $T_{(k+1) \bmod i+1}$;
 - ajouter les arêtes de raccord de la dimension d_k à T_{d_k} .

Mais cet algorithme n'est pas optimal : pour chaque liste `Indices` de i indices, on appelle $1 \times 2 \times 3 \times \dots \times (d - i + 2) \times (d - i + 1) \times (d - i)$ fois la procédure `Construire_Arbres(Ordres, Indices)`, à partir de l'appel où $i = d$.

Au delà de trois dimensions, une représentation graphique des arbres est pratiquement impossible. La figure 14 tente cependant de mettre en évidence le cas quadridimensionnel.

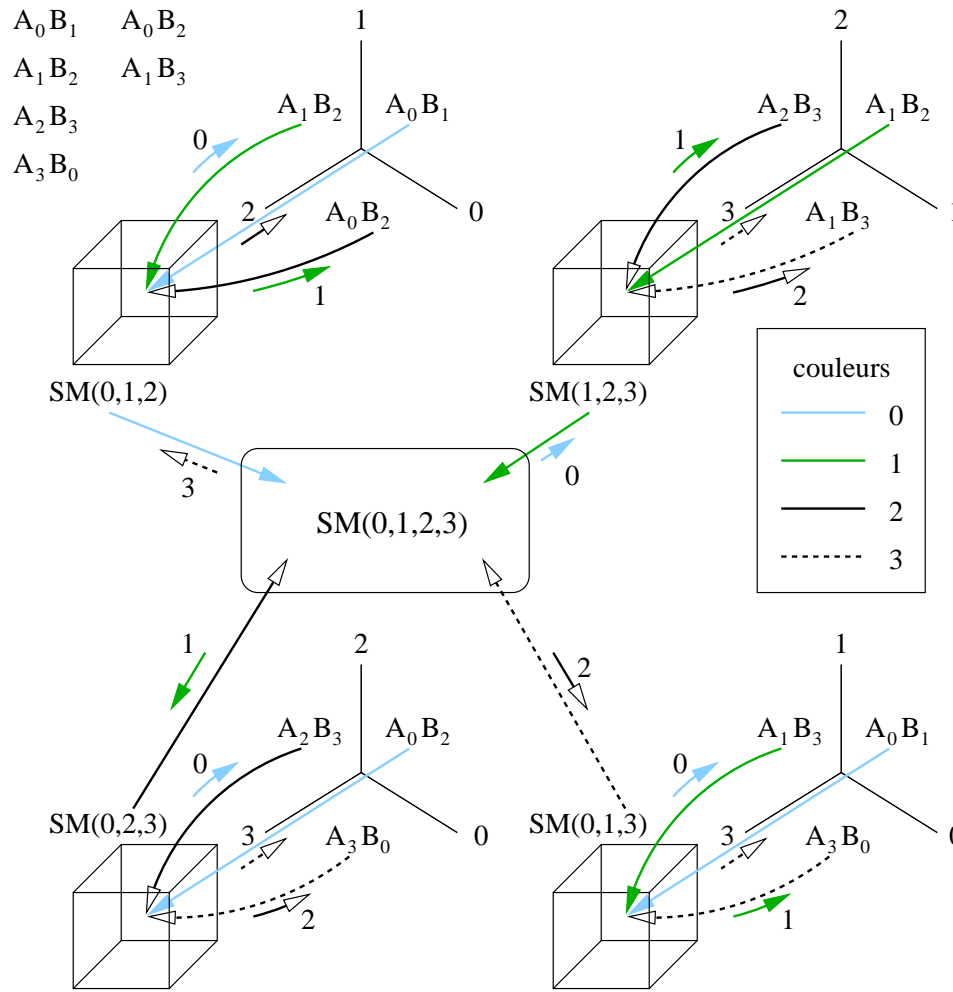


FIG. 14 – Représentation schématique de l’algorithme A pour $d = 4$

Les quatre sous-tores tridimensionnels sont représentés schématiquement avec leur répartition des sous-arbres A et B , dont la liste est donnée en haut à gauche de la figure. Les chaînes de diffusion dans les sous-grilles sont matérialisées par de longues flèches issues du numéro de couleur auquel elles correspondent. Les arêtes de raccord sont représentées parallèlement par des flèches plus courtes et sont étiquetées par la dimension suivie, qui est aussi le numéro de couleur qui les utilise et le numéro de dimension suivie par les chaînes de diffusion.

Nous allons montrer que chaque couleur c couvre bien tous les sommets du tore. Lors de la première étape, tous les sommets des sous-tores bidimensionnels $STM(d_0, d_1)$ sont coloriés par les couleurs d_0 et d_1 , dont les sommets des axes, qui sont atteints par toutes les couleurs. En effet, chaque axe c est partagé par $d - 1$ sous-tores $STM(c, k)$, $\forall k \in [0, d - 1] \setminus \{c\}$. Donc, ses sommets sont atteints par les $d - 1$ couleurs k , en plus de la couleur c qui utilise $n_c - 1$ de ses arêtes. Puis, à chaque étape i , $\forall i \in [2, d - 1]$, on complète le coloriage des sous-tores $(i+1)$ -dimensionnels $STM(d_0, d_1, \dots, d_i)$ qui, après cette étape, sont atteints par toutes les couleurs d_j , $\forall j \in [0, i]$. En particulier, à l’étape $d - 1$, le sous-tore complété est $STM(d_0, d_1, \dots, d_{d-1})$, qui peut s’écrire $STM(0, 1, \dots, d - 1)$, et qui n’est autre que $TM(n_0, n_1, \dots, n_{d-1})$.

Pour montrer que toutes les couleurs utilisent des arêtes distinctes, on peut partitionner l’ensemble des arêtes empruntées par la couleur c lors des $d - 1$ étapes en trois sous-ensembles :

- les arêtes des sous-arbres de type A et B ;
- les arêtes des chaînes de diffusion dans les sous-grilles ;
- les arêtes de raccord.

Or, pour chacun de ces sous-ensembles, la démonstration de l'absence de recouvrement de couleurs différentes est simple :

- Par hypothèse, les arêtes de tous les axes ne sont utilisées que par une et une seule couleur (la couleur c pour l'axe c) et toujours de la même façon. Quant aux autres arêtes des sous-tores bidimensionnels, elles appartiennent soit à un sous-arbre de type A , soit à un sous-arbre de type B de couleur différente, qui sont arête-disjoints par définition. Donc, il n'y a pas de conflit de couleur.
- Les sous-grilles $(i + 1)$ -dimensionnelles $SM(d_0, d_1, \dots, d_i)$ ne sont coloriées qu'à l'étape i , en utilisant exclusivement la couleur $c = d_{(k+1) \bmod i+1}$ pour les chaînes de la dimension d_k , $\forall k \in [0, i]$.
- Quant aux arêtes de raccord, elles suivent aussi une dimension différente pour chaque couleur, à chaque étape i : la dimension c pour la couleur c .

On peut facilement donner une borne supérieure sur la profondeur des arbres. A l'étape 1, on a construit des arbres dans les sous-tores bidimensionnels ; puis, à chaque étape i allant de 2 à $d - 1$, on les a prolongés par des chaînes de diffusion (de longueur $n_{d_k} - 1$ dans la dimension d_k) et des arêtes de raccord. Mais les sommets atteints par les arêtes de raccord d'une étape $i < d - 1$ sont des feuilles et ne sont donc pas à l'origine des chaînes de diffusion de l'étape $i + 1$. Quand $i < d - 1$, on ne doit donc pas compter les arêtes de raccord dans le calcul de la profondeur des arbres. La profondeur de T_c est donc donnée par la somme des longueurs des $(d - 2)$ chaînes de diffusion dont la succession constituent les plus longues branches de l'arbre, à laquelle s'ajoutent la profondeur du sous-arbre bidimensionnel de couleur c dont certains des sommets sont à l'origine de ces chaînes et 1 pour le dernier raccord. Or, dans l'algorithme \mathcal{A} , le sous-arbre de couleur c dont la profondeur entre en ligne de compte est toujours un sous-arbre de type A . En effet, parcourons l'arbre en partant des feuilles les plus éloignées de la racine (les raccords en dimension d) : à l'étape $d - 1$, l'arbre T_c utilise des chaînes de la dimension $(c - 1) \bmod d$; à l'étape $d - 2$, ce sont des chaînes de la dimension $(c - 2) \bmod d$; et ainsi de suite, jusqu'à l'étape 2, où la dimension utilisée est $(c + 2) \bmod d$. On est donc parti de $STM(0, 1, \dots, c - 2, c - 1, c, c + 1, c + 2, \dots, d - 1)$ pour arriver à $STM(c, (c + 1) \bmod d)$ qui contient le sous-arbre qui est à l'origine des chaînes que nous avons suivies. Or, dans $STM(c, (c + 1) \bmod d)$, c'est bien un sous-arbre A_c que nous avons plongé lors de la première étape. Donc, la profondeur de l'arbre T_c construit par l'algorithme \mathcal{A} est :

$$h(T_c) = h(A_c) + \sum_{i=(c+2) \bmod d}^{(c-1) \bmod d} (n_i - 1) + 1$$

Or, comme on a $\lfloor \frac{n_0}{2} \rfloor + \lfloor \frac{n_1}{2} \rfloor \leq h(TM(n_0, n_1)) \leq \lfloor \frac{n_0}{2} \rfloor + \lfloor \frac{n_1}{2} \rfloor + 1$, la profondeur $h(T_c)$ est bien majorée par :

$$h(TM(n_0, n_1)) + \sum_{i=2}^{d-1} (n_i - 1) + 1$$

si $n_0 \leq n_1 \leq \dots \leq n_{d-1}$. □

Corollaire 7.2 *Pour tout $d > 3$, $TM(n)^d$ admet d arbres couvrants deux à deux arête-disjoints de profondeur $2n + (d - 3)(n - 1)$.*

PREUVE. Ce résultat s'obtient en utilisant les arbres de [4] pour A et B dans l'algorithme \mathcal{A} . En effet, dans ce cas on a $h(A_c) = h(B_c) = n$. Donc, $\forall c \in [0, d - 1]$, $h(T_c) = n + (d - 2)(n - 1) + 1$. La condition $d > 3$ vient du fait que dans le cas tridimensionnel, il est possible de diminuer la profondeur des arbres en minimisant la distance par rapport à l'origine des sommets qui sont à l'origine des chaînes de diffusion qui engendrent les raccords (voir la preuve de la proposition 6.2). L'ensemble de ces sommets est appelé X et l'ensemble des sommets origines des autres chaînes est appelé X' . Quand $d > 3$, les sommets origines des chaînes de diffusion à l'étape 3 sont les sommets de toutes les chaînes construites à l'étape 2, c'est-à-dire issues à la fois des sommets de X et de X' . Or, dans $X \cup X'$, il existe forcément des sommets à distance n de l'origine. □

Le tableau suivant exprime la borne supérieure sur la profondeur des arbres en fonction du diamètre de $TM(n)^d$ quand n tend vers l'infini, pour des valeurs significatives de d .

d	h/D
2	1
3	1.333...
4	1.5
5	1.6
6	1.666...
7	1.714285...
8	1.75
9	1.777...
10	1.8

(suite)	
20	1.9
25	1.92
40	1.95
50	1.96
100	1.98
200	1.99
⋮	⋮
∞	2

Proposition 7.3 *Si $n_0 \leq n_1 \leq \dots \leq n_{d-1}$ et $d > 3$:*

$$h(TM(n_0, n_1, \dots, n_{d-1})) \leq \left\lfloor \frac{n_0}{2} \right\rfloor + \left\lfloor \frac{n_1}{2} \right\rfloor + \sum_{i=2}^{d-1} (n_i - 1) + \varepsilon$$

où ε vaut 1 si n_0 et n_1 sont pairs, et 2 sinon.

PREUVE. Une fois de plus, le résultat s'obtient en prenant les arbres de [4] pour A et B dans l'algorithme \mathcal{A} . La valeur de ε résulte du fait que $d > 3$ et qu'il existe des sommets dans X' qui sont à distance $h(TM(n_0, n_1))$ de l'origine. Elle comprend aussi le coût de 1, induit par le raccord. Dans le cas quadridimensionnel, on déduit facilement la profondeur des arbres de la figure 14 :

$$\begin{aligned} h(T_0) &= \left\lfloor \frac{n_0}{2} \right\rfloor + \left\lfloor \frac{n_1}{2} \right\rfloor + \varepsilon(n_0, n_1) + n_2 - 1 + n_3 - 1 \\ h(T_1) &= \left\lfloor \frac{n_1}{2} \right\rfloor + \left\lfloor \frac{n_2}{2} \right\rfloor + \varepsilon(n_1, n_2) + n_3 - 1 + n_0 - 1 \\ h(T_2) &= \left\lfloor \frac{n_2}{2} \right\rfloor + \left\lfloor \frac{n_3}{2} \right\rfloor + \varepsilon(n_2, n_3) + n_0 - 1 + n_1 - 1 \\ h(T_3) &= \left\lfloor \frac{n_3}{2} \right\rfloor + \left\lfloor \frac{n_0}{2} \right\rfloor + \varepsilon(n_3, n_0) + n_1 - 1 + n_2 - 1 \end{aligned}$$

Le terme $\varepsilon(a, b)$ vient du fait que d'une manière générale, on peut exprimer la profondeur des arbres dans un tore bidimensionnel $TM(a, b)$ de la façon suivante :

$$h(T) = \alpha a + \alpha' b + \varepsilon(a, b)$$

où α et α' sont des coefficients inférieurs ou égaux à 1 et où $\varepsilon(a, b)$ est en fait un terme constant qui peut dépendre de la parité de a et/ou de b . Or, quand $\alpha_c < 1$ et/ou $\alpha'_c < 1$ (c'est le cas des arbres de [4], où $\alpha = \alpha' = \frac{1}{2}$, ou encore des arbres de Simmen modifiés, où $\alpha = \frac{1}{2}$ et $\alpha' = 1$ ou inversement), on peut diminuer la profondeur des arbres obtenus par l'algorithme \mathcal{A} , en modifiant les choix qui ont été faits pour associer couleurs et dimensions lors des diffusions dans les sous-grilles aux étapes 2 à $d-1$. En effet, les chaînes de diffusion sont attribuées aux arbres en mettant en bijection les couleurs et les dimensions. Or, il n'est pas nécessaire de fixer la même bijection pour toutes les sous-grilles à une étape i donnée. La seule contrainte à respecter est évidemment d'imposer que la couleur attribuée à une chaîne de la dimension d_k dans $SM(d_0, d_1, \dots, d_i)$ soit dans $[d_0, d_i] \setminus d_k$. Les arbres de la figure 14 sont obtenus directement de l'algorithme \mathcal{A} qui utilise la même bijection, afin de construire des arbres très réguliers (ils peuvent être obtenus les uns des autres par translation sur les numéros de dimension). La figure 15 montre un exemple de gain de profondeur par rapport à la figure 14 dans le cas quadridimensionnel, obtenu uniquement en changeant les bijections qui associent les couleurs aux dimensions aux étapes 2 et 3 de l'algorithme \mathcal{A} .

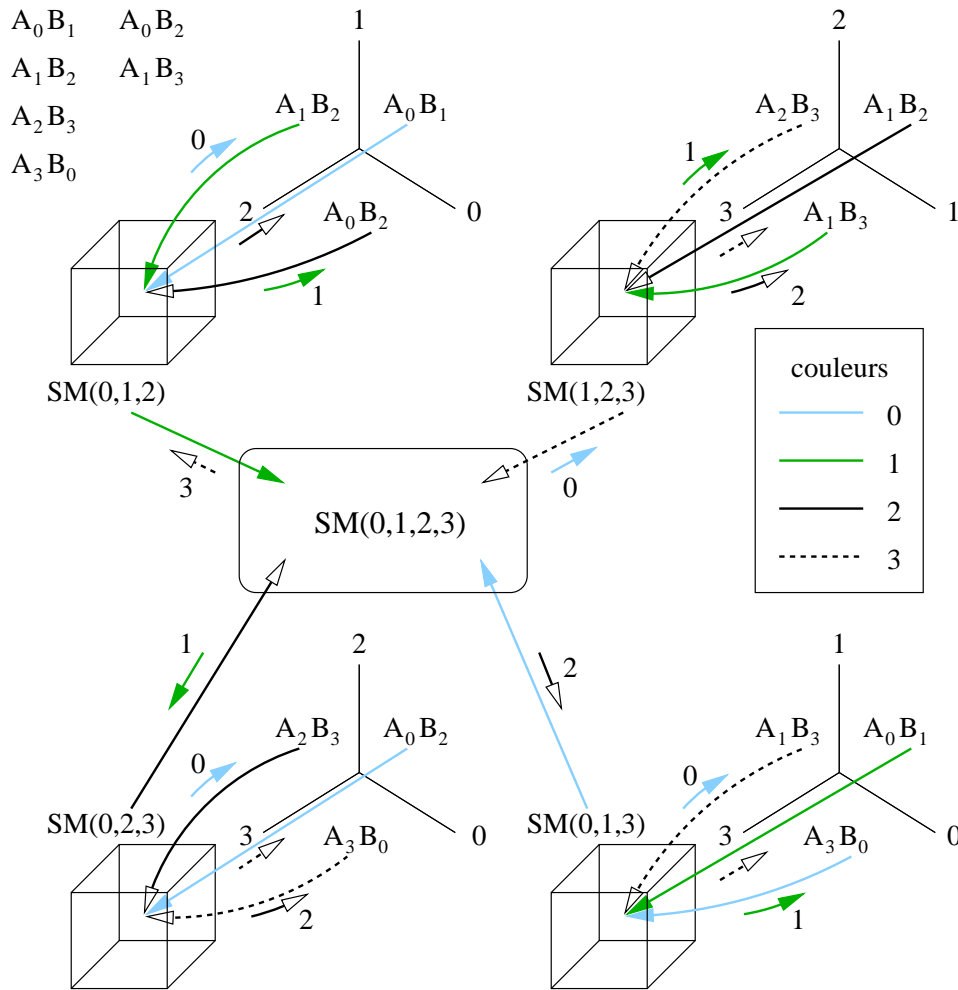


FIG. 15 – Exemple de modification (diminution de la profondeur) des arbres pour $d = 4$

Les profondeurs obtenues sont alors :

$$\begin{aligned}
 h(T_0) &= \left\lfloor \frac{n_0}{2} \right\rfloor + \left\lfloor \frac{n_3}{2} \right\rfloor + \varepsilon(n_0, n_3) + n_1 - 1 + n_2 - 1 \\
 h(T_1) &= \left\lfloor \frac{n_1}{2} \right\rfloor + \left\lfloor \frac{n_2}{2} \right\rfloor + \varepsilon(n_1, n_2) + n_0 - 1 + n_3 - 1 \\
 h(T_2) &= \left\lfloor \frac{n_2}{2} \right\rfloor + \left\lfloor \frac{n_3}{2} \right\rfloor + \varepsilon(n_2, n_3) + n_0 - 1 + n_1 - 1 \\
 h(T_3) &= \left\lfloor \frac{n_3}{2} \right\rfloor + \left\lfloor \frac{n_2}{2} \right\rfloor + \varepsilon(n_3, n_2) + n_1 - 1 + n_0 - 1
 \end{aligned}$$

Il est facile de vérifier que dans la construction de la figure 14, la profondeur majorante est celle de l'arbre T_0 . Ici, cette profondeur est clairement diminuée. De plus, la profondeur des arbres T_1 et T_2 reste inchangée et celle de T_3 est aussi diminuée. Ces modifications ne sont possibles que pour $d > 3$ (si $d = 3$, il est facile de voir que le choix de l'association d'une couleur à une dimension impose toutes les autres associations) et le nombre de permutations possibles des couleurs augmente avec d . Dans le cas général, c'est-à-dire si $n_0 \leq n_1 \leq \dots \leq n_{d-1}$, le calcul de la profondeur globale dépend donc à la fois de d et de toutes les valeurs de n_i , $\forall i \in [0, d-1]$. Cependant, on peut toujours donner une valeur majorante : celle de l'arbre T_0 obtenu avec l'algorithme \mathcal{A} sans changement. \square

REMARQUE. Dans [15], Fraigniaud donne une construction très simple de 4 arbres couvrants arc-disjoints et de même racine dans $TM^*(n_0, n_1)$, qui correspond en fait aux arbres de Simmen, adaptés au cas orienté symétrique (voir section 4). Cette construction est généralisée aux tores $TM^*(n_0, n_1, \dots, n_{d-1})$ dans [17], où l'on peut aussi en trouver une adaptation pour le cas non orienté. Or, cette construction peut aussi être obtenue

directement de l'algorithme \mathcal{A} , en prenant les arbres de Simmen pour A et B . Les arbres obtenus sont alors de profondeur $\sum_{i=0}^{d-1} (n_i - 1) + 1 = 2D - d + 1$, ce qui est toujours moins bon qu'en prenant les arbres de [4], dès que $n_0 > 2$. Par exemple, dans $TM(n)^3$, on obtient des arbres de profondeur $2n - 1$, ce qui est bien meilleur que $3n - 2$. La correspondance entre ces deux méthodes permet cependant d'imaginer une adaptation de l'algorithme \mathcal{A} au cas de la grille torique orientée symétrique, en précisant des propriétés d'orientation sur les arcs de A et B , notamment au niveau des axes (les arbres de Simmen [29] ayant donc ces propriétés, une fois orientés).

On peut aussi déduire une construction valable pour $H^*(d)$, à partir des arbres obtenus dans $TM(2)^d$. On retrouve alors le résultat de Ho et Johnson dans [20], comme corollaire du théorème 7.1 :

Corollaire 7.4 *L'hypercube orienté symétrique de dimension d , $H^*(d)$, admet d arbres couvrants deux à deux arc-disjoints et de même racine, de profondeur optimale $d + 1$.*

PREUVE. Par définition, on peut écrire $TM(2)^d = C_2 \square C_2 \square \dots \square C_2$ (où C_2 est répété d fois) et $H^*(d) = P_2^* \square P_2^* \square \dots \square P_2^* = \vec{C}_2 \square \vec{C}_2 \square \dots \square \vec{C}_2$ (où P_2^* et \vec{C}_2 sont répétés d fois). Donc, $TM(2)^d$ peut être vu comme un hypercube orienté symétrique de dimension d dont on aurait omis l'orientation. Or, si on prend les arbres de [29] pour A et B dans l'algorithme \mathcal{A} et que l'on oriente dans le sens de la diffusion les arêtes de $TM(2)^d$ utilisées par les arbres obtenus, on obtient une construction qui est aussi valide pour $H^*(d)$. La figure 16 montre les cas $d = 3$ et $d = 4$.

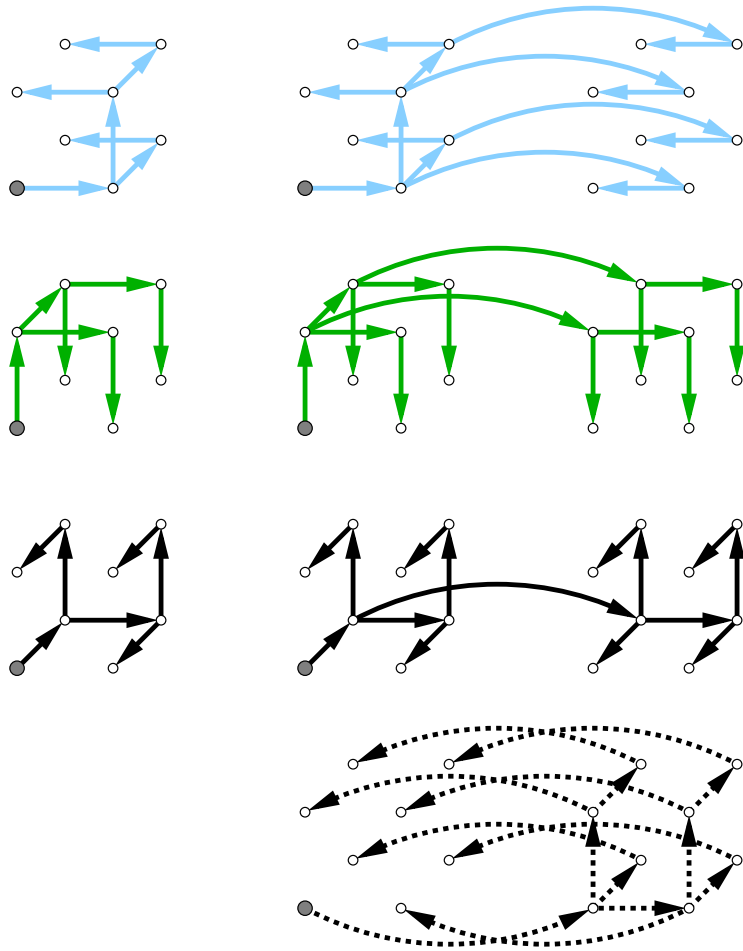


FIG. 16 – Arbres de profondeur optimale $d + 1$ dans $H^*(d = 3)$ et $H^*(d = 4)$

Nous allons montrer qu'il n'y a jamais deux arêtes orientées dans le même sens. Lors de la première étape, le plongement des arbres de [29] dans $STM(d_0, d_1)$ revient à orienter certaines arêtes en un chemin reliant les sommets $(0 \dots 0)$, $(0 \dots 0, x_{d_0} = 1, 0 \dots 0)$, $(0 \dots 0, x_{d_0} = 1, 0 \dots 0, x_{d_1} = 1, 0 \dots 0)$ et $(0 \dots 0, x_{d_1} = 1, 0 \dots 0)$, dans cet

ordre pour T_{d_0} , et un autre reliant les sommets $(0 \dots 0)$, $(0 \dots 0, x_{d_1} = 1, 0 \dots 0)$, $(0 \dots 0, x_{d_0} = 1, 0 \dots 0, x_{d_1} = 1, 0 \dots 0)$ et $(0 \dots 0, x_{d_0} = 1, 0 \dots 0)$, dans cet ordre pour T_{d_1} , où $0 \dots 0$ représente une suite de coordonnées nulles. Les deux arêtes reliant les sommets $(0 \dots 0, x_{d_0} = 1, 0 \dots 0)$ et $(0 \dots 0, x_{d_0} = 1, 0 \dots 0, x_{d_1} = 1, 0 \dots 0)$ (resp. $(0 \dots 0, x_{d_1} = 1, 0 \dots 0)$ et $(0 \dots 0, x_{d_0} = 1, 0 \dots 0, x_{d_1} = 1, 0 \dots 0)$) ont bien des orientations opposées. A l'étape i , $\forall i \in [0, d-1]$, on n'utilise que des arêtes appartenant à des sous-tore de dimension $i+1$. S'il y a un conflit d'orientation sur une arête double lors des étapes 2 à $d-1$, il est clair qu'il ne peut survenir qu'entre l'arête de diffusion dans la sous-grille SM (réduite à un seul sommet) et l'arête de raccord d'un même sous-tore de dimension $i+1$. Or, lors d'une étape i , $\forall i \in [2, d-1]$, une arête de diffusion dans SM est toujours orientée depuis un sous-tore de dimension i vers SM et une arête de raccord est toujours orientée depuis SM vers un sous-tore de dimension i . Toutes les arêtes doubles de $TM(2)^d$ utilisées par les arbres construits ont donc reçu une orientation conforme avec celle de $H^*(d)$. D'après le corollaire 7.2, la profondeur des arbres est $2 + (d-2)1 + 1 = d+1$. \square

8 Etude comparative de performances

Dans [9], je donne un algorithme local à chaque nœud qui oriente tout paquet reçu vers ses prochains destinataires dans l'arbre concerné, en fonction de sa position relative à l'origine de la diffusion. On se situe dans un cas classique de symétrie de code : tous les nœuds doivent avoir le même algorithme car tous sont susceptibles de jouer le rôle de n'importe quel autre, en fonction de l'origine de la diffusion (l'algorithme qui tourne sur l'origine n'est pas intéressant en lui-même, car il s'agit d'une boucle de découpage du message initial en paquets de taille optimale et transmis sur tous ses liens de sortie en fonction de leur couleur).

Le temps logiciel ajouté au temps de communication proprement-dit peut être considéré comme négligeable et incorporé à β . En effet, l'algorithme de calcul des destinataires de chaque paquet, pour un nœud émetteur donné, n'est exécuté qu'une seule fois, au moment de la réception du premier paquet de la diffusion par ce nœud.

Même si je n'ai pas encore implémenté cet algorithme, il me semblait intéressant d'en évaluer la performance théorique avec des valeurs réelles de paramètres d'une machine récente. Pour cela, j'ai pris pour référence les travaux de Calvin et Colombet sur le Cray T3D (dont on peut extrapoler les résultats au plus récent T3E), qui est une machine *wormhole* (voir [5] ou [6]).

Dans ces travaux, ils arrivent à la conclusion que la valeur de δ est négligeable, quelle que soit la couche logicielle utilisée pour les primitives de communication d'envoi et de réception de messages. On peut donc estimer que :

$$\forall d(x, y) \quad \alpha + d(x, y) \delta = \beta$$

Bien entendu, les évaluations de β et τ sont différentes suivant la couche logicielle utilisée. En effet, elles dépendent directement des choix d'implémentation : transparence, encapsulages, etc. Deux environnements ont été testés : les bibliothèques PVM et ShMem.

- La première se doit d'être portable, donc d'offrir une certaine transparence, et garantit la fiabilité des données transmises. Cela n'est possible qu'au prix d'une baisse de performance par rapport à la puissance nominale de la machine : augmentation du coût logiciel (incorporé dans le paramètre β) et perte de bande passante pour les données utiles, due à l'ajout d'informations de contrôle aux messages. Enfin, la pertinence du placement (le plus souvent transparent) des processeurs virtuels sur les ressources physiques est capitale.
- La seconde est la couche de communication de plus bas niveau disponible. Son objectif est d'offrir les services d'une mémoire distribuée (virtuelle) sur une mémoire physiquement distribuée. Ses points faibles résident dans les conflits d'accès concurrents et l'absence de garantie sur la fiabilité des données transférées.

La solution la plus satisfaisante dans le cas d'un algorithme de communication intensif comme celui dont on traite ici est d'utiliser la primitive `pvm_channel`, qui est un compromis entre ces deux extrêmes. Elle permet de s'affranchir de la trop grande transparence de l'environnement PVM, tout en garantissant la fiabilité des données. Ce système de canaux est une extension de PVM par Cray Research. Il permet d'établir des connexions unidirectionnelles explicites entre les processeurs, et ainsi de «coller» au mieux à l'architecture physique de la machine. Les valeurs obtenues pour β et τ sont d'ailleurs proches des meilleures performances observées (petit β et une bande passante d'environ 103 Mo/s) : $\beta = 10.23 \mu s$ et $\tau = 0.0097 \mu s/octet$. De plus, ce comportement reste constant, quelle que soit la taille du message.

Tous les graphiques de la figure 17 représentent des temps de diffusion, exprimés en μs , en fonction de la taille L du message, exprimée en octets. Ces temps sont des estimations à partir des équations suivantes, en remplaçant les paramètres β et τ par les valeurs expérimentales observées avec `pvm_channel` :

$$b_{wormhole} \left(TM \left((2k+1)^i \right)^k \right) = \log_{2k+1} \left((2k+1)^{i^k} (\beta + L\tau) \right) = i^k (\beta + L\tau)$$

$$b_{store-and-forward} (L, p, t) = \left(\sqrt{(p-1)\beta} + \sqrt{\frac{L\tau}{t}} \right)^2$$

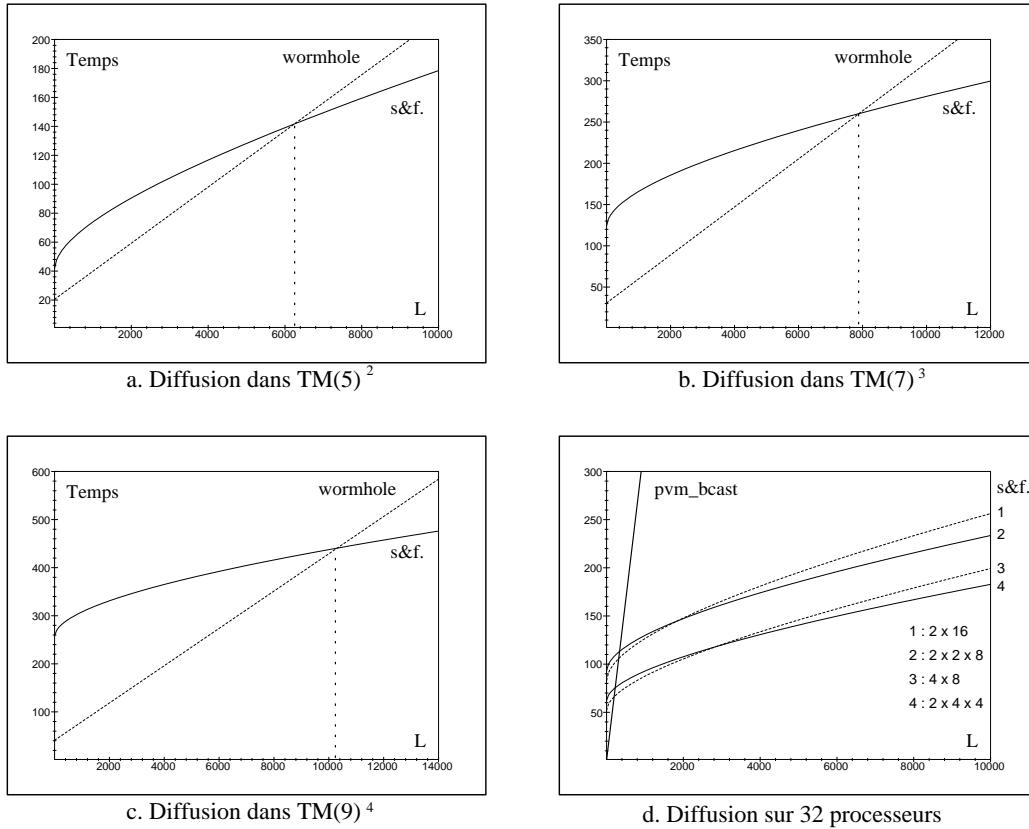


FIG. 17 – Performances théoriques sur Cray T3D

Les graphiques *a*, *b* et *c* mettent en évidence les tailles de message critiques à partir desquelles l'algorithme de diffusion pipe-linéé sur arbres couvrants devient plus efficace que l'algorithme *wormhole* optimal en nombre d'étapes de Delmas et Perennes [12]. Les tailles de message critiques sont respectivement d'environ 6, 8 et 10 Kilo-octets pour les tores $TM(5)^2$, $TM(7)^3$ et $TM(9)^4$, ce qui est très raisonnable. Bien entendu, seuls les deux premiers cas sont réalisables dans une machine comme le T3D. Le troisième exemple montre que la progression est assez lente pour assurer cette supériorité, même dans le cas d'éventuelles machines à plus de dimensions.

Quant au dernier graphique, il compare le temps de diffusion moyen obtenu sur 32 processeurs par la primitive `pvm_bcast` aux temps estimés de l'algorithme *store-and-forward*, pour toutes les combinaisons possibles de tores bidimensionnels ou tridimensionnels d'ordre 32 (du fait de la trop grande transparence offerte par PVM, il n'est pas possible d'avoir une vision de la topologie réelle testée). On voit que dans tous les cas, les résultats obtenus par l'algorithme *store-and-forward* sont de loin les meilleurs. A titre d'exemple, sur un tore $2 \times 4 \times 4$, il est 32 fois plus rapide pour $L = 30\,000$ et plus de 42 fois pour $L = 60\,000$. L'implémentation de ce genre d'algorithme dans les bibliothèques de communication du T3D serait donc un bénéfice considérable par rapport aux outils logiciels trop généraux existants.

9 Conclusion

La construction proposée profite au maximum de la définition récursive de la grille torique, apportant de multiples avantages au niveau de la mise en œuvre de l'algorithme de diffusion. Elle permet une réutilisabilité maximale des résultats connus du cas bidimensionnel et reste valide, quels que soient les paramètres du tore.

Une amélioration de la profondeur de ces arbres pourrait être obtenue en cassant les chaînes de diffusion dans les sous-grilles, mais cela paraît très difficile (et peut-être impossible sans casser les sous-arbres des sous-tores), même dans le cas tridimensionnel.

L'optimalité réside probablement dans des solutions spécifiques (pour un nombre de dimensions donné), mais ce serait certainement au prix de la simplicité, donc de l'efficacité de l'algorithme de routage des paquets, à moins de découvrir une solution algébrique (à base de théorie des codes, par exemple), comme cela a été fait en *wormhole* dans [11] et [12] pour le tore et dans [22] pour l'hypercube. Cependant, il est à remarquer que ces solutions algébriques optimales sont limitées à un nombre de cas très restreints. Par exemple, les résultats de [12] ne sont valables que pour des tores $TM((2k+1)^i)^k$ et ceux de [11] qu'en dimension 3 pour des tores $TM(7^i)^3$. Dans [7], Calvin, Perennes et Trystram proposent des techniques d'extension de ces cas réguliers pour l'échange total, mais en multipliant les algorithmes.

Or, comme on l'a vu dans la section 8, c'est plus le nombre d'arbres que leur profondeur qui influe sur le temps global de diffusion et les temps théoriques obtenus sont très satisfaisants.

Références

- [1] J-C. Bermond et P. Fraigniaud. Communications in Interconnection Networks. In *Proceedings of Combinatorial Optimization in Science and Technology'91*, 1991.
- [2] J-C. Bermond et P. Fraigniaud. Broadcasting and NP-Completeness. In *Graph Theory Notes of New York*, vol. 23, pp. 8–14, 1992.
- [3] J-C. Bermond et P. Fraigniaud. Broadcasting and Gossiping in de Bruijn Networks. *SIAM Journal on Computing*, 23(1):212–225, 1994.
- [4] J-C. Bermond, P. Michallon et D. Trystram. Broadcasting in wraparound meshes with parallel monodirectional links. *Parallel Computing*, 18:639–648, 1992.
- [5] C. Calvin. *Minimisation du sur-coût des communications dans parallélisation des algorithmes numériques*. Thèse de Doctorat, Institut National Polytechnique de Grenoble, juillet 1995.
- [6] C. Calvin et L. Colombet. Performance Evaluation and Modeling of Collective Communications on Cray T3D. *Parallel Computing*, 22:1413–1427, 1996.
- [7] C. Calvin, S. Perennes et D. Trystram. Gossiping in torus with wormhole-like routing. In *Proceedings of the 7th IEEE Symposium on Parallel and Distributed Processing*, San Antonio, pp. ???–???, octobre 1995.
- [8] E. Darrot. Diffusion *store-and-forward* dans les grilles toriques tridimensionnelles. In *Actes RenPar'8, 8èmes Rencontres francophones sur le Parallélisme*, Bordeaux (France), p. 208, mai 1996.
- [9] E. Darrot. *Diffusion de messages longs et propriétés structurelles dans les réseaux d'interconnexion*. Thèse de Doctorat, Université de Nice - Sophia Antipolis, juin 1997.
- [10] O. Delmas. *Communications par commutation de circuits dans les réseaux d'interconnexion*. Thèse de Doctorat, Université de Nice - Sophia Antipolis, janvier 1997.
- [11] O. Delmas et S. Perennes. Circuit-switched gossiping in 3-dimensional torus networks. Rapport de Recherche n° 2930 — Thème 1, INRIA Sophia Antipolis, juillet 1996.
- [12] O. Delmas et S. Perennes. Diffusion en mode commutation de circuits dans les tores de dimension k . *Technique et Science Informatiques*, 16(5):563–581, 1997.
- [13] J. Edmonds. Edge-disjoint branchings. In R. Rustin, editor, *Combinatorial Algorithms*, pp. 91–96. Algorithmics press, New York, 1972.
- [14] E. Fleury. *Communications, routage et architectures des machines à mémoire distribuée – Autour du routage wormhole*. Thèse de Doctorat, Université de Lyon/Ecole Normale Supérieure de Lyon, octobre 1996.
- [15] P. Fraigniaud. *Communications intensives dans les architectures à mémoire distribuée et algorithmes parallèles pour la recherche de racines de polynômes*. Thèse de Doctorat, Ecole Normale Supérieure de Lyon, décembre 1990.
- [16] P. Fraigniaud et C. Laforest. Disjoint spanning trees of small depth. In *Parallel Computing: trends and applications*, pp. 105–112. North-Holland, 1993. Actes de la conférence ParCo'93, Grenoble (France), 7-10 septembre 1993.
- [17] P. Fraigniaud et E. Lazard. Methods and problems of communication in usual networks. *Discrete Applied Mathematics*, 53:79–133, 1994.
- [18] Zhengyu Ge et S. Louis Hakimi. Disjoint rooted spanning trees with small depths in de Bruijn and Kautz graphs. *SIAM Journal on Computing*, 26(1):79–92, 1997.
- [19] M-C. Heydemann, J. Opatrny et D. Sotteau. Broadcasting and spanning trees in de Bruijn and Kautz networks. *Discrete Applied Mathematics*, 37/38:297–317, 1992.
- [20] C. T. Ho et S. L. Johnson. Optimum broadcasting and personalized communication in hypercubes. *IEEE Transactions on Computers*, 38(9):1249–1268, 1989.

- [21] C. T. Ho et M. T. Raghunath. Efficient communication primitives on hypercubes. Rapport Technique n° RJ-72915, IBM San Jose, 1991.
- [22] T. Kodate. *Communications structurées dans les réseaux d'interconnexion*. Thèse de Doctorat, Université de Nice - Sophia Antipolis, octobre 1996.
- [23] S. Kundu. Bounds on the number of disjoint spanning trees. *Journal of Combinatorial Theory*, 17:199–203, 1974.
- [24] P. Michallon et D. Trystram. Minimum depth Arc-Disjoint Spanning Trees for broadcasting on wrap-around meshes. In *Proceedings of ICPP'95*, Oconomowoc, USA, pp. 80–83, août 1995.
- [25] P. Michallon, D. Trystram et G. Villard. Optimal broadcasting algorithms on torus. Rapport de Recherche n° RR-872 -I-, LMC-IMAG, janvier 1992.
- [26] Jean de Rumeur. *Communications dans les réseaux de processeurs*. Collection études et recherches en informatique. Masson, Paris, 1994. Version anglaise à paraître.
- [27] Y. Saad et M. Schultz. Data communication in parallel architectures. *Parallel Computing*, 11:131–150, 1989.
- [28] S. R. Seidel, M. H. Lee et S. Fotedar. Concurrent bidirectional communication on the Intel iPSC860 and iPSC2. Rapport Technique n° CS-TR 90-06, Michigan Technological University, Houghton, MI, 1990.
- [29] M. Simmen. Comments on broadcast algorithms for two-dimensional grids. *Parallel Computing*, 17:109–112, 1991.

Table des matières

1	Introduction	3
2	Définitions et notations	5
3	Bornes inférieures	6
4	Cas bidimensionnel	7
5	Approche d-dimensionnelle	9
6	Cas tridimensionnel	11
7	Généralisation	15
8	Etude comparative de performances	22
9	Conclusion	24



Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Lorraine : Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399