ELSEVIER

Theoretical Computer Science 263 (2001) 173–189

Theoretical Computer Science

www.elsevier.com/locate/tcs

Efficient reconfiguration algorithms of de Bruijn and Kautz networks into linear arrays

Rabah Harbane, Marie-Claude Heydemann *

LRI, UMR CNRS, bât 490, Université Paris-Sud, 91405 Orsay, Cedex, France

Accepted April 2000

Abstract

In this paper, we prove the existence of *ranking* and *unranking* algorithms on d-ary de Bruijn and Kautz graphs. A *ranking* algorithm takes as input the label of a node and returns the rank r of that node in a hamiltonian path ($0 \leqslant r \leqslant N-1$, where N is the order of the considered graph). An *unranking* algorithm takes as input an integer r ($0 \leqslant r \leqslant N-1$) and returns the label of the rth ranked node in a hamiltonian path. Our results generalize results given by Annexstein for binary de Bruijn graphs. The key of our framework is based on a recursive construction of hamiltonian paths in de Bruijn and Kautz graphs. The construction uses suitable uniform homomorphisms of de Bruijn and Kautz graphs of diameter D on de Bruijn graphs of diameter $D-1$. Our ranking and unranking algorithms have sequential time complexity in $O(D^2)$, where D is the length of node labels. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: de Bruijn graph; Kautz graph; Uniform homomorphism; Hamiltonian path; Ranking/unranking algorithms

1. Introduction

In implementation of parallel algorithms in parallel machine, we often need to emulate the physical topology by another one because certain parallel algorithms are very efficient when they are implemented in a suitable parallel machine. This emulation can be done by generating the software that reconfigures the physical architecture to the logical one.

A popular approach to model this problem is to consider graph embeddings which embed the underlying logical topology into the physical one. Unfortunately, the results obtained by this method may induce algorithms that run in time linear in the size of the topologies, which is not useful in practice.

* Corresponding author.
E-mail address: mch@lri.fr (Marie-Claude Heydemann).

PII: S0304-3975(00)00240-1

In [1], the author introduced the notion of parallel implementation of graph embeddings and gave results concerning parallel embeddings of trees, X-trees on the hypercube. These results demonstrate the way in which graph embeddings can be implemented efficiently by using parallel algorithms.

In this paper, we will focus on the problem of reconfiguration involving the use of hamiltonian paths in de Bruijn and Kautz graphs. The existence of hamiltonian paths in these graphs is well known. However, the existence of such paths does not imply that they may be used effectively in parallel computations on networks. Paths can be embedded efficiently in parallel by developing fast algorithms.

Specifically, the problem we consider is the following: given an N-node labelled graph, find a hamiltonian path along with associated *ranking* and *unranking* algorithms. The *ranking* algorithm takes as input the label of a node and returns the rank r $(0 \leqslant r \leqslant N-1)$ of that node in the hamiltonian path. The *unranking* algorithm takes as input an integer r $(0 \leqslant r \leqslant N-1)$ and returns the label of the rth ranked node in the hamiltonian path.

In a previous paper, Annexstein proved the existence of such algorithms in the hypercube, the binary de Bruijn graphs and the butterfly graphs [2]. In this paper, we prove the existence of *ranking* and *unranking* algorithms in d-ary de Bruijn and Kautz graphs and generalize the results given by Annexstein for binary de Bruijn graphs.

The key of our framework is based on a recursive construction of hamiltonian paths in de Bruijn and Kautz graphs. The construction uses suitable uniform homomorphisms of de Bruijn and Kautz graphs of diameter D on de Bruijn graphs of diameter $D-1$.

Our ranking and unranking algorithms have sequential time complexity in $O(D^2)$, where D is the length of node labels. Thus, when such algorithms are applied to reconfigure de Bruijn or Kautz networks into linear arrays, they run as SIMD-style programs in parallel time poly-logarithmic in the size of the networks. This time bound is exponentially faster than any implementation that relies on listing all the nodes in the path.

2. Definitions and notation

Let G and H be undirected graphs. An *embedding* of G into H is a one-to-one mapping $\varphi : V(G) \to V(H)$ of the nodes of G to the nodes of H. If G and H are node-labeled graphs, we say that there is a *parallel embedding* of G into H (realizing φ) if there is a pair of algorithms that effect the mapping φ, specified as follows:

- a *labelling algorithm* $\mathscr{A}$ which inputs a node-label g of $V(G)$ and outputs the node-label of $\varphi(g)$ of $V(H)$.
- an *unlabelling algorithm* $\mathscr{A}'$ which inputs a node-label h of $V(H)$ and outputs the node-label of $\varphi^{-1}(h)$ of $V(G)$.

The quality of parallel embedding depends on two cost measures: the dilation and the run-time.

The *dilation* of parallel embedding φ is equal to $\max_{(x,y)\in E(G)} d(\varphi(x),\varphi(y))$ (where $d(x',y')$ denotes the distance between the two vertices x', y', i.e. the minimum number of edges of a path with end vertices x', y').

The *run-time* of the parallel embedding is the number of parallel bit-operations necessary to implement the algorithms $\mathscr{A}$ and $\mathscr{A}'$. In the case where H is a path, we call algorithms $\mathscr{A}$ and $\mathscr{A}'$, *ranking* and *unranking* algorithms.

Let $G_1=(V_1,A_1)$ and $G_2=(V_2,A_2)$ be two digraphs and $f: V_1\to V_2$ be a vertex mapping. f is a *homomorphism* of G_1 into G_2 if it is arc-preserving ($\forall (x_1,y_1)\in A_1$; $(f(x_1),f(y_1))\in A_2$). Notice that such a homomorphism f induces an arc mapping f' from A_1 into A_2 defined by $f'(a_1)=a_2$, where $a_1=(x_1,y_1)$ and $a_2=(f(x_1),f(y_1))$.

The *load* of vertex $x_2\in V_2$ under f, $load_f(x_2)$, is equal to $|f^{-1}(x_2)|$. The *vertex load* of f is the maximum load over all vertices in V_2. Homomorphism f is *vertex-uniform* iff

$$\forall x_2\in V_2;\quad \left\lfloor \frac{|V_1|}{|V_2|}\right\rfloor \leqslant load_f(x_2) \leqslant \left\lceil \frac{|V_1|}{|V_2|}\right\rceil .$$

The *load* of arc $a_2\in A_2$ under f, $load_f(a_2)$, is equal to $|f'^{-1}(a_2)|$. The *arc load* of f is the maximum load over all arcs in A_2. Homomorphism f is *arc-uniform* iff

$$\forall a_2\in A_2;\quad \left\lfloor \frac{|A_1|}{|A_2|}\right\rfloor \leqslant load_f(a_2) \leqslant \left\lceil \frac{|A_1|}{|A_2|}\right\rceil .$$

Let G be a digraph (directed graph). We denote by $\delta^+(u)$ (respectively, $\delta^-(u)$) the *out-degree* (respectively, *in-degree*) of a vertex u of G, i.e. the number of arcs (u,v) (respectively (v,u)) of G.

The (di)graphs we will consider here are labelled by strings which are d-ary words. For simplicity, we often confuse a vertex and its label. For convenience of notation, strings will be indexed here from right to left, for example a D-letter string will be written $\mathbf{x}=x_{D-1}x_{D-2}\dots x_0$.

Throughout this paper, the word of length D containing the letter a D times will be denoted by a^D.

Let $\mathbb{Z}_d=\{0,1,\dots,d-1\}$ and $\mathbb{Z}_d^n=\{x_nx_{n-1}\dots x_2x_1 | x_i\in\mathbb{Z}_d\}$ the set of all d-ary words of length n on $\mathbb{Z}_d$. Let $B(d,D)$ and $K(d,D)$ denote the de Bruijn and Kautz digraphs respectively, with in- and out-degree d and diameter D [4, 5, 7], defined as follows:

$$V(B(d,D))=\mathbb{Z}_d^D,$$

$$A(B(d,D))=\{(x_{D+1}x_D\dots x_2, x_D\dots x_2x_1)\,|\,x_i\in\mathbb{Z}_d, 1\leqslant i\leqslant D+1\},$$

$$V(K(d,D))=\{x_Dx_{D-1}\dots x_1\,|\,\forall i: x_i\in\mathbb{Z}_{d+1}, \forall i, 1\leqslant i\leqslant D-1: x_i\neq x_{i+1}\},$$

$$A(K(d,D))=\{(x_{D+1}x_D\dots x_2, x_D\dots x_2x_1)\,|\,x_i\in\mathbb{Z}_{d+1}, 1\leqslant i\leqslant D+1, x_i\neq x_{i+1}, 1\leqslant i\leqslant D\}.$$

Arcs $(x_{D+1}x_D\dots x_2, x_D\dots x_2x_1)$ of $B(d,D)$ or $K(d,D)$ represent operation called *left shifting*. Figs. 1 and 2 show de Bruijn digraph $B(2,4)$ and Kautz digraph $K(2,4)$, respectively.

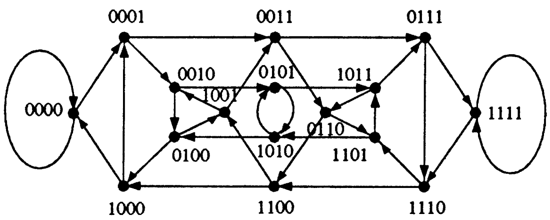

Fig. 1. De Bruijn digraph $B(2,4)$.

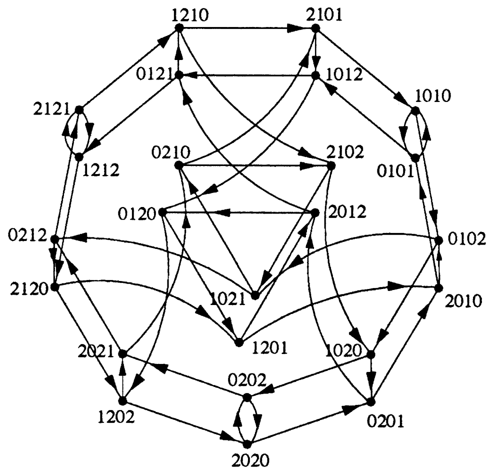

Fig. 2. Kautz digraph $K(2,4)$.

We will call de Bruijn and Kautz digraphs jointly *B/K-digraphs*. B/K-digraphs form a family of digraphs that have been extensively studied as they have many useful properties for designing interconnection networks (see [3]). The B/K-graphs are obtained, respectively, from the B/K-digraphs by deleting the orientation of the edges, loops and parallel edges.

Let us recall that $B(d,D)$, $K(d,D)$ and their associated graphs have all diameter D.

Let G be a digraph and P be a (non-directed) path in G. Let $\mathbf{u}$ and $\mathbf{v}$ be the end vertices of P. Consider the directed path P^* obtained from P which begins at vertex $\mathbf{u}$, ends at vertex $\mathbf{v}$ and is oriented from $\mathbf{u}$ to $\mathbf{v}$. The arc (i,j) of G will be said a *positive arc* (of P^*) if (i,j) is also an arc of P^* (in other words, its orientation coincides with that chosen for traversing P). Otherwise, it will be said a *negative arc*. For any vertex $\mathbf{y}$ in P, we denote by $S(y)$ the difference between the number of positive arcs and the number of negative ones, needed to reach the node $\mathbf{y}$ on P from the first node $\mathbf{x}$. For example, if P is the path in $B(2,4)$: $0001, 0011, 1001, 0010$, then $(0001, 0011)$ and $(1001, 0010)$ are positive arcs and $(0011, 1001)$ is a negative arc, thus, $S(0001) = 0$, $S(0011) = 1$, $S(1001) = 0$, $S(0010) = 1$.

We will often use the following property of B/K-digraphs (which is just a generalization of the property used in the binary case in [2]).

Lemma 1. *Let P be a path in a B/K-digraph of diameter D, beginning at vertex $\mathbf{x} = x_{D-1}x_{D-2}\ldots x_0$. If for any vertex $\mathbf{z}$ on P, $0 \leqslant S(z) \leqslant D-1$, then, for any vertex $\mathbf{y} = y_{D-1}y_{D-2}\ldots y_0$ on P, $y_{S(y)} = x_0$.*

Proof. Let us recall that in B/K-digraphs a positive (negative) arc corresponds to a left (right) shifting of the letters in the word labelling the vertex. Since, for any vertex $\mathbf{z}$ on P, $0 \leqslant S(z) \leqslant D-1$, the cyclic shifting never moves the original least significant letter x_0 past the most significant letter position, nor past the least significant letter position, and it is therefore left unaltered. Furthermore, the position of x_0 in the word $\mathbf{y}$ is given by $S(y)$, thus $y_{S(y)} = x_0$. $\square$

3. Background on uniform homomorphisms of de Bruijn and Kautz digraphs

We recall here some results related to uniform homomorphisms of de Bruijn and Kautz digraphs that we need in next sections and which can be found in [8].

Proposition 1. *Let $\diamond$ be a binary operation on $\mathbb{Z}_d$. Then the mapping $f^D_{(\diamond)}$ defined by*

$$f^D_{(\diamond)}(x_1x_2\ldots x_D) = (x_1 \diamond x_2)(x_2 \diamond x_3)\ldots(x_{D-1} \diamond x_D)$$

is a homomorphism of $B(d,D)$ onto $B(d,D-1)$.

Proposition 2. *If operation $\diamond$ satisfies at least one of the following two properties (B) and (B'):*
$\forall\, \alpha \in \mathbb{Z}_d$, mapping $x \mapsto \alpha \diamond x$ is a bijection on $\mathbb{Z}_d$ $\quad (B)$
$\forall\, \alpha \in \mathbb{Z}_d$, mapping $x \mapsto x \diamond \alpha$ is a bijection on $\mathbb{Z}_d$ $\quad (B')$.
Then for any $D > 1$, $f^D_{(\diamond)}$ is a vertex- and arc-uniform surjective homomorphism.

Obviously, any operation $\diamond$ such that $(\mathbb{Z}_d, \diamond)$ is a group, induces a vertex- and arc-uniform homomorphism of $B(d,D)$ onto $B(d,D-1)$. By applying Proposition 2, we get homomorphisms already known in the literature.

Example 1. The following four mappings $f_i : V(B(d,D)) \to V(B(d,D-1))$, $i = 1,2,3,4$, are vertex- and arc-uniform homomorphisms of $B(d,D)$ onto $B(d,D-1)$:
- $f_1(x_{D-1}x_{D-2}\ldots x_0) = x_{D-2}x_{D-3}\ldots x_0$,
- $f_2(x_{D-1}x_{D-2}\ldots x_0) = x_{D-1}x_{D-2}\ldots x_1$,
- $f_3(x_{D-1}x_{D-2}\ldots x_0) = (x_{D-1} \oplus_d x_{D-2})(x_{D-2} \oplus_d x_{D-3})\ldots(x_1 \oplus_d x_0)$,
- $f_4(x_{D-1}x_{D-2}\ldots x_0) = (x_{D-1} \ominus_d x_{D-2})(x_{D-2} \ominus_d x_{D-3})\ldots(x_1 \ominus_d x_0)$,

where $\oplus_d$ and $\ominus_d$ are operations $+$ and $-$ modulo d, respectively.

In fact, it is proved in [8] that every surjective homomorphism of $B(d,D)$ onto $B(d,D-1)$ is of the form $f^D_{(\diamond)}$, but we do not need here such a characterization. The same results are true for Kautz digraphs as follows.

Proposition 3. *Let* $\diamond$ *be a binary operation on* $\mathbb{Z}_{d+1}$ *such that*
- $\forall i \neq j \in \mathbb{Z}_{d+1};\ i \diamond j \in \mathbb{Z}_d$,
- $\forall i \in \mathbb{Z}_{d+1};\ i \diamond i = d$,

then the mapping $f^D_{(\diamond)}$ *defined by*

$$f^D_{(\diamond)}(x_1 x_2 \dots x_D) = (x_1 \diamond x_2)(x_2 \diamond x_3)\dots(x_{D-1} \diamond x_D)$$

is a homomorphism f *of* $K(d,D)$ *onto* $B(d,D-1)$.

Proposition 4. *If operation* $\diamond$ *satisfies at least one of the following two properties* (KB) *and* (KB'):

$\forall\, \alpha \in \mathbb{Z}_d$, *mapping* $x \mapsto \alpha \diamond x$ *is a bijection on* $\mathbb{Z}_{d+1}$ (KB),

$\forall\, \alpha \in \mathbb{Z}_d$, *mapping* $x \mapsto x \diamond \alpha$ *is a bijection on* $\mathbb{Z}_{d+1}$ (KB').

Then for any $D > 1$, $f^D_{(\diamond)}$ *from* $K(d,D)$ *onto* $B(d,D-1)$ *is a vertex- and arc-uniform surjective homomorphism.*

We define a $d \times d$ matrix $M_\diamond = (m_{i,j})^{d-1,d-1}_{i,j=0,0}$, with $m_{i,j} = i \diamond j$, representing the *multiplication table* of $\diamond$. By Proposition 1, each table of this kind represents a homomorphism associated to the operation $\diamond$. Notice that an operation $\diamond$ satisfies properties (B), (B'), iff all rows, all columns, respectively, of $M_\diamond$ are permutations of $\mathbb{Z}_d$. Similarly, we can consider a $(d+1) \times (d+1)$ matrix for defining homomorphisms of $K(d,D)$ into $B(d,D-1)$.

4. Reconfiguration algorithms for de Bruijn graphs

In this section, we will explicitly give a parallel embedding of the path of order d^D in the de Bruijn graph $UB(d,D)$, by specifying the ranking and unranking algorithms. We first give an idea of the construction which generalizes the one given by Annexstein [2].

4.1. Sketch of construction

It is based on the use of uniform homomorphisms $f^D_{(\diamond)}$ of the de Bruijn digraphs $B(d,D)$ into $B(d,D-1)$ which satisfy the properties (B) and (B') as they are defined in Proposition 2. In [8], the existence of such homomorphisms which are d vertex- and arc-uniform from $B(d,D)$ into $B(d,D-1)$ is proved. Let us consider such a homomorphism $f^D_{(\diamond)}$, we write f for short. Each arc of $B(d,D-1)$ is the image under f of d node disjoint arcs of $B(d,D)$. It follows by induction that any path P of the undirected graph $UB(d,D-1)$ is the image of d node disjoint paths of $UB(d,D)$, which are isomorphic to P as directed graphs.

Thus, the pre-image by f of any hamiltonian path P_{D-1} of $UB(d,D-1)$ consists of d pairwise node disjoint paths, each one being a digraph isomorphic to P_{D-1}. We denote these paths of $B(d,D)$ by P^i, $0 \leqslant i \leqslant d-1$ (omiting the subscript D for sake of simplicity). Such a homomorphism f allows to construct recursively a hamiltonian path P_D in $UB(d,D)$ by using the hamiltonian P_{D-1} of $UB(d,D-1)$ only if there are suitable arcs of $B(d,D)$ to join the paths P^i. Let $\mathbf{p}^i$ be the beginning vertex of the path P^i and $\mathbf{t}^i$ its end vertex. We will assume that f is chosen such that it fulfils the following condition:

$$\text{(C):}\quad (\mathbf{t}^{2j}, \mathbf{t}^{2j+1}),\ 0 \leqslant j \leqslant \left\lfloor \frac{d-2}{2} \right\rfloor, \qquad \text{and} \quad (\mathbf{p}^{2j+2}, \mathbf{p}^{2j+1}),$$
$$0 \leqslant j \leqslant \left\lfloor \frac{d-3}{2} \right\rfloor, \text{ are arcs of } B(d,D).$$

The choice of the homomorphism f is then determined by this condition (see Section 4.3).

Under this hypothesis, let us explain the recursive construction of the hamiltonian path P_D.

We construct the hamiltonian path P_D by joining the paths P^i, with the arcs $(\mathbf{t}^{2j}, \mathbf{t}^{2j+1})$ and $(\mathbf{p}^{2j+2}, \mathbf{p}^{2j+1})$. More precisely, the path P^{2j} $(j \geqslant 0)$ is joined to the path P^{2j+1} by the positive arc $(\mathbf{t}^{2j}, \mathbf{t}^{2j+1})$, and the path P^{2j+1} is joined to the path P^{2j+2} by the negative arc $(\mathbf{p}^{2j+2}, \mathbf{p}^{2j+1})$.

Then, if $\overleftarrow{P}^i$ denotes the path P^i traversed from $\mathbf{t}^i$ to $\mathbf{p}^i$,

$$P_D = P^0 + \overleftarrow{P}^1 + P^2 + \overleftarrow{P}^3 + \cdots + \overleftarrow{P}^{d-2} + P^{d-1}$$

if d is odd and

$$P_D = P^0 + \overleftarrow{P}^1 + P^2 + \overleftarrow{P}^3 + \cdots + P^{d-2} + \overleftarrow{P}^{d-1}$$

if d is even.

The idea of the ranking algorithm is then the following. Given a vertex $\mathbf{x}$ of $B(d,D)$, compute the vertex $f(\mathbf{x}) = \mathbf{y}$ of $B(d,D-1)$ and recursively the rank r of $\mathbf{y}$. By construction of the path P_D, if we know to which path P^i the vertex $\mathbf{x}$ belongs, we can easily compute its rank r':

$$r' = \begin{cases} i d^{D-1} + r & \text{if } i \text{ is even,} \\ (i+1) d^{D-1} + 1 - r & \text{if } i \text{ is odd.} \end{cases}$$

We now introduce tools which allow to determine P^i.

4.2. Tools based on total shifting

Let $S(D,i)$ be the difference between the number of positive arcs and the number of negative ones, needed to reach the node of rank i on P_D from the first node 0^D. The following results indicate how to compute $S(D,i)$. Notice that, if d is odd, the number of positive arcs $(\mathbf{t}^{2j},\mathbf{t}^{2j+1})$, and the number of negative arcs $(\mathbf{p}^{2j+2},\mathbf{p}^{2j+1})$ are equal and have the same value equal to $\lfloor d/2 \rfloor$. On the contrary, if d is even, then there are $d/2$ positive arcs $(\mathbf{t}^{2j},\mathbf{t}^{2j+1})$ and $d/2-1$ negative arcs $(\mathbf{p}^{2j+2},\mathbf{p}^{2j+1})$.

Lemma 2. *For any* $i=0,1,\ldots,d-1$,

$$S(1,i)=\begin{cases} 0 & \textit{if } i \textit{ is even},\\ 1 & \textit{if } i \textit{ is odd}.\end{cases}$$

Proof. For $D=1$, the hamiltonian P_1 of $UB(d,1)$ is the alternate hamiltonian di-path of $B(d,1)$ which begins at vertex 0 with a positive arc and ends at vertex $d-1$:

$$P_1: 0\rightarrow 1 \leftarrow 2 \rightarrow 3 \leftarrow \cdots \leftarrow 2j \rightarrow 2j+1 \leftarrow \cdots d-1$$

with $\delta^-(0)=0$, $\delta^+(d-1)+\delta^-(d-1)=1$, $\delta^+(2j)=2$, $\delta^-(2j)=0$, for $1\leqslant j\leqslant \lfloor (d-2)/2 \rfloor$, and $\delta^+(2j+1)=0$, $\delta^-(2j+1)=2$ for $0\leqslant j \leqslant \lfloor (d-3)/2 \rfloor$.

Hence, the difference between the number of positive arcs and the negative ones needed to reach the vertex i from 0 is equal to 0 if i is even and it is equal to 1 if i is odd.

Furthermore, by construction of P_1, we have

$$S(1,i)=\begin{cases} 0 & \text{if } i \text{ is even},\\ 1 & \text{if } i \text{ is odd}\end{cases}$$

and, in particular

$$S(1,d-1)=\begin{cases} 0 & \text{if } d \text{ is odd},\\ 1 & \text{if } d \text{ is even}. \qquad \Box\end{cases}$$

A recursive formula for computing $S(D,i)$ is given by the following proposition.

Proposition 5. *For any* $i=0,1,\ldots,d-1$,

$$S(1,i)=\begin{cases} 0 & \text{if } i \text{ is even},\\ 1 & \text{if } i \text{ is odd}.\end{cases}$$

For any $D>1$, *if* $\omega^i, 0\leqslant i \leqslant d^D-1$, *is the vertex of rank* i *on* P_D,

$$S(D,i)=\begin{cases} S(D-1, i \bmod d^{D-1}) & \text{if } \omega^i\in P^{2j},\ 0\leqslant j\leqslant \lfloor \frac{d-2}{2} \rfloor\\ S(D-1, d^{D-1}-i \bmod d^{D-1}-1)+1 & \text{if} \omega^i\in P^{2j+1},\ 0\leqslant j\leqslant \lceil \frac{d-4}{2} \rceil.\end{cases}$$

Proof. By Lemma 2, the formula is true for $D=1$. Assume $D\geqslant 2$. Recall that the construction of the hamiltonian path P_D of $B(d,D)$ verifies the following arguments:

(a) Each path P^i has the same orientation (of arcs) as P_{D-1}.

(b) The construction of P_D from P_{D-1} uses the paths P^i alternatively in the orientation of P_{D-1} and the opposite direction.
(c) The arcs needed for connecting the end vertex of P^{2j} to the begin vertex of P^{2j+1} are positive arcs and that used for connecting P^{2j+1} to P^{2j+2} are negative arcs.

In the construction of the hamiltonian path P_D, the cyclic shifting in the paths P^{2j}, $0 \leqslant j \leqslant \lfloor (d-1)/2 \rfloor$, are the same as that accomplished in P_{D-1} and the cyclic shifting in the paths P^{2j+1}, $0 \leqslant j \leqslant \lfloor d/2-1 \rfloor$, are the reverse of that done in P_{D-1}. Hence, knowing the sense of the arcs joining the paths P^i, we get that the total cyclic shifting $S(D,i)$ required after i steps, $i \in \{2jd^{D-1}, 2jd^{D-1}+1, \ldots, (2j+1)d^{D-1}-1\}$ $(0 \leqslant j \leqslant \lfloor (d-1)/2 \rfloor)$ is equal to the total cyclic shifting in P_{D-1}, which is equal to $S(D-1, i \bmod d^{D-1})$.

The cyclic shiftings done in the paths P^{2j+1}, $0 \leqslant j \leqslant d/2-1$, are the reverse of that done in P_{D-1}. These cyclic shiftings undo the shiftings accomplished in the even paths P^{2j}, $0 \leqslant j \leqslant d/2-1$. Thus, the total cyclic shifting $S(D,i)$ required after i steps, $i \in \{(2j+1)d^{D-1}, \ldots, (2j+2)d^{D-1}-1\}$ (with $0 \leqslant j \leqslant \lfloor (d-2)/2 \rfloor$), is equal to the value $S(D-1, d^D - i \bmod d^{D-1}-1)$ plus the amount of cyclic shifting due to the arc (t_{2j}, t_{2j+1}) used to join the path P^{2j} to the path P^{2j+1}. This value is equal to $+1$ by construction. $\square$

As a corollary of Proposition 5, we get:

Corollary 1. *$0 \leqslant S(D,i) \leqslant D$, for any D and any $0 \leqslant i \leqslant d^D-1$,*

$$S(D, d^D-1) = \begin{cases} 1 & \text{if } d \text{ is even}, \\ 0 & \text{if } d \text{ is odd}. \end{cases}$$

Proof. If d is odd, by Proposition 5, $S(D, d^D-1) = S(D-1, d^{D-1}-1) = \cdots = S(1, d-1) = 0$. If d is even, $S(D, d^D-1) = S(D-1, 0) + 1 = 1$. $\square$

We are now able to prove the following result which will be used to verify condition (C) in the next section.

Corollary 2. *Let f be the homomorphism of $B(d,D)$ onto $B(d,D-1)$ associated with a binary operation $\diamond$ which satisfies properties (B) and (B') of Proposition 2. Then, in the recursive construction of the hamiltonian path P_D using the paths P^i, $0 \leqslant i \leqslant d-1$, we can compute the end vertex of P^i if we know the beginning vertex of P^i and the end vertex of P_{D-1}.*

Proof. Let $\mathbf{x} = x_{D-1} \ldots x_1 x_0$, $\mathbf{y} = y_{D-1} \ldots y_1 y_0$, be the first, last vertex, respectively, of P^i, and let $\mathbf{z} = z_{D-2} \ldots z_1 z_0$ be the last vertex of P_{D-1}. If d is odd, by Corollary 1, $S(D-1, d^{D-1}-1) = 0$. Since P^i is a digraph isomorphic to P_{D-1}, with the notation of Lemma 1, $S(D-1, j) = S(\mathbf{z})$ if $\mathbf{z}$ is the vertex of rank j on P^i. By Corollary 1, for any i, $0 \leqslant S(D-1, i) \leqslant D-1$. Thus, using Lemma 1, we get $x_0 = y_0$. Similarly, if d is even, $S(D-1, d^{D-1}-1) = 1$ and $y_1 = x_0$. On the other hand, by the homomorphism

f, the end vertex of P^i satisfies the following equations:

$$y_{i+1} \diamond y_i = z_i, \quad i = 0, 1, \ldots, D-2.$$

Since y_0 or y_1 is known, using property (B) or property (B'), the computation of $\mathbf{y}$ is immediate. $\square$

Thus, if we know P_{D-1} and the homomorphism f, we can compute the first and the last vertex of the paths P^i and verify if the condition (C) is fulfilled.

4.3. Choice of the homomorphism f

Let us now precisely describe the suitable homomorphism which allows the recursive construction of the hamiltonian path in $UB(d,D)$. Following the parity of d, we distinguish two cases. In each case, we only have to prove that the chosen homomorphism satisfies condition (C).

We use the following notation. Let $\ldots\alpha\beta$ denote the string of length D which ends by $\alpha\beta$ and continues by alternating β's and α's to the left. The word of length D which begins with a letter a and is composed alternatively with letters a and b will be denoted $(ab)^*$. Let us recall that we denote by a^D the word of length D containing the letter a D times. For example, if $D=3$, then $\ldots 01 = 101$, $(01)^* = 010$, $0^3 = 000$.

Case 1: *d is odd*. In this case, we consider the homomorphism $f_3 : B(d,D) \to B(d,D-1)$ given in Example 1 and defined by

$$f_3(x_{D-1}x_{D-2}\ldots x_0) = (x_{D-1} \oplus x_{D-2})(x_{D-2} \oplus x_{D-3})\ldots(x_1 \oplus x_0),$$

where $\oplus$ is the addition modulo d.

By using this homomorphism, we will construct by induction on D a hamiltonian path in $UB(d,D)$, which begins at vertex $\mathbf{p}_D = 0^D$ and ends at a vertex $\mathbf{t}_D$ of the form α_D^D where $\alpha_D \in \mathbb{Z}_d$ is defined recursively as the solution in $\mathbb{Z}_d$ of the equations:

$$\alpha_1 = d-1, \quad 2\alpha_D = \alpha_{D-1}.$$

Let us now prove by induction on D that we can construct a hamiltonian path in $UB(d,D)$ which begins at vertex 0^D and terminates at vertex α_D^D. An illustration of this construction is given by Fig. 3.

The construction is true for $D=1$. Assume that we have constructed a hamiltonian path P_{D-1} of $UB(d,D-1)$ which begins at vertex $\mathbf{p}_{D-1} = 0^{D-1}$ and terminates at vertex $\mathbf{t}_{D-1} = \alpha_{D-1}^{D-1}$. The pre-images by f of $\mathbf{t}_{D-1}$ are the vertices $x_{D-1}\ldots x_1x_0$ such that $x_{i+1} \oplus x_i = \alpha_{D-1}$. Thus,

$$f^{-1}(\mathbf{t}_{D-1}) = \{\alpha_D^D\} \cup \{(ab)^*, (ba)^*; a \oplus b = \alpha_{D-1}\}.$$

Furthermore,

$$f^{-1}(\mathbf{p}_{D-1}) = \left\{(i(d-i))^*, ((d-i)i)^*; 1 \leqslant i \leqslant \left\lfloor \frac{d}{2} \right\rfloor \right\} \cup \{0^D\}.$$

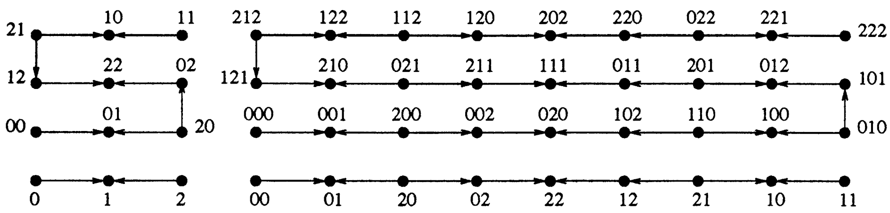

Fig. 3. Recursive construction of the hamiltonian path in $UB(3,3)$.

The path P_{D-1} is the image under the homomorphism f of d node disjoint paths of $B(d,D)$, namely P^0, $P^1,\dots,P^{d-1}$. We denote by P^0 the path which begins at $\mathbf{p}_D = 0^D$. By Corollary 2, we know that P^0 terminates at $\dots\alpha_{D-1}0$. Let P^1 be the path which begins at $\dots(d-\alpha_{D-1})\alpha_{D-1}$, then it terminates at $\dots 0\alpha_{D-1}$. Let P^2 be the path which begins at $\dots\alpha_{D-1}(d-\alpha_{D-1})$, then it terminates at $\dots 2\alpha_{D-1}(d-\alpha_{D-1})$.

More generally, for any $j \geqslant 0$, by Corollary 2,

- the path P^{2j} which begins at vertex $\mathbf{p}^{2j} = \cdots(j\alpha_{D-1})(d-j\alpha_{D-1})$, terminates at vertex $\mathbf{t}^{2j} = \cdots[(j+1)\alpha_{D-1}](d-j\alpha_{D-1})$,
- the path P^{2j+1} which begins at vertex $\mathbf{p}^{2j+1} = \dots[(d-(j+1)\alpha_{D-1})][(j+1)\alpha_{D-1}]$, terminates at vertex $\mathbf{t}^{2j+1} = \cdots[(d-j\alpha_{D-1})][(j+1)\alpha_{D-1}]$.

Notice that the paths P^i can be connected since $(\mathbf{t}^{2j},\mathbf{t}^{2j+1})$ and $(\mathbf{p}^{2j+2},\mathbf{p}^{2j+1})$ are arcs of $B(d,D)$. Hence, condition (C) is fulfilled and the path P_D defined by

$$P_D = P^0 + \overleftarrow{p}^1 + P^2 + \overleftarrow{p}^3 + \cdots + \overleftarrow{p}^{d-2} + P^{d-1}$$

is a hamiltonian path in $UB(d,D)$.

Case 2: *d is even*. In this case the previous homomorphism does not work since it does not enable to construct recursively a hamiltonian path in $UB(2^m,D)$. For instance, the previous homomorphism f_3 fails in the construction of the hamiltonian path in $UB(4,3)$ from the hamiltonian path of $UB(4,2)$ because the 4 paths, P^i, $i=0,\dots,3$ are not pairwise adjacent since $f_3^{-1}(22) = \{111,333,020,202\}$ and there does not exist two arcs induced by these vertices in $B(4,2)$.

In this paragraph, we will define a homomorphism $f_\diamond^D$ of $B(d,D)$ into $B(d,D-1)$ using the binary operation $\diamond$ defined on $\mathbb{Z}_d$ by the table $M_\diamond = [m_{ij}]_{i=0,\dots,d-1}^{j=0,\dots,d-1}$ as follows:

$$m_{00} = m_{d-1d-1} = 0,$$

$$m_{(2i-1)\ (2i)} = m_{(2i)\ (2i-1)} = 0 \quad \text{for any } i = 1,\dots,\tfrac{d-2}{2},$$

$$m_{2i(2i+1)} = m_{(2i+1)2i} = d-1 \quad \text{for any } i = 0,1,\dots,\tfrac{d-2}{2},$$

$$m_{ii} = d-2 \quad \text{for any } i = 1,2,\dots,d-2,$$

$$m_{0j} = d-1-j \quad \text{for } j = 2,\dots,d-2$$

$$m_{0(d-1)} = d-2$$

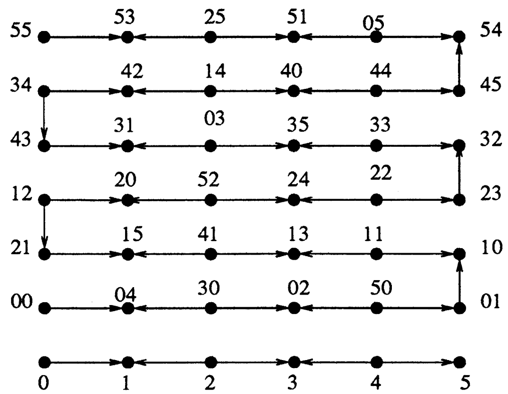

Fig. 4. Recursive construction of the hamiltonian path in $UB(6,2)$.

$$m_{ij} = d + i - j - 1 \quad \text{for } 3 \leqslant 2 + i \leqslant j \leqslant d - 1,$$

$$m_{ij} = i - j - 1 \quad \text{for } 2 \leqslant j + 2 \leqslant i \leqslant d - 1.$$

For example if $d = 6$, we use the homomorphism $f_\diamond^D$, induced by the following table:

$\diamond$	0	1	2	3	4	5
0	0	5	3	2	1	4
1	5	4	0	3	2	1
2	1	0	4	5	3	2
3	2	1	5	4	0	3
4	3	2	1	0	4	5
5	4	3	2	1	5	0

Similarly as for the odd case, by using the homomorphism $f_\diamond$, we will construct inductively a hamiltonian path in $UB(d,D)$ which begins at vertex 0^D and terminates at vertex $(d-1)^D$ (Fig 4).

The construction is true for $D = 1$. Assume that there exists a hamiltonian path of $UB(d,D-1)$, denoted by P_{D-1}, which begins at 0^{D-1} and terminates at $(d-1)^{D-1}$. The path P_{D-1} is the image under the homomorphism $f_\diamond^D$ of d node disjoint paths of $B(d,D)$, namely P^0, $P^1, \dots, P^{d-1}$ such that

- the path P^0 begins at vertex 0^D and terminates at vertex $\dots 01$,
- the path P^{2j+1} begins at vertex $\mathbf{p}^{2j+1} = \cdots (2j+2)(2j+1)$ and terminates at vertex $\mathbf{t}^{2j+1} = \cdots (2j+1)2j$, with $j = 0, \dots, (d-4)/2$,
- the path P^{2j} begins at vertex $\mathbf{p}^{2j} = \cdots (2j-1)2j$ and terminates at vertex $\mathbf{t}^{2j} = \cdots 2j(2j+1)$, with $j = 1, \dots, (d-2)/2$,
- the path P^{d-1} begins at vertex $\mathbf{p}^{d-1} = \cdots (d-1)(d-1)$ and terminates at vertex $\mathbf{t}^{d-1} = \cdots (d-1)(d-2)$.

The reader can verify that condition (C) is fulfilled.

4.4. Algorithms

The discussion and the results given above allow us to introduce the following ranking and unranking algorithms in $UB(d,D)$. Notice that these algorithms are written for both cases of parity of d since this parity is only needed to specify the suitable homomorphism.

Algorithm 1 (*Ranking in* $UB(d,D)$).
Input: $\mathbf{x} = x_{D-1}x_{D-2}\ldots x_0 \in V(UB(d,D))$.
Output: $rank(\mathbf{x}) \in \{0,1,\ldots,d^D-1\}$.
Begin
Initialization: For $i=0,1,\ldots,d-1$ do $rank(i)=i$.
1. Compute $f(\mathbf{x})=\mathbf{y}$.
2. Compute recursively $rank(\mathbf{y})=r$.
3. Compute $S(D-1,r)=k$, by using Proposition 5.
4. Compute i such that $\mathbf{x}\in P^i$, using the fact that the first vertex of P^i, say $z_{D-1}\ldots z_0$, satisfies $x_k=z_0$.
5.

$$rank(\mathbf{x}) = \begin{cases} id^{D-1}+r & \text{if } i \text{ is even,} \\ (i+1)d^{D-1}-1-r & \text{otherwise.} \end{cases}$$

End.

Proof. By construction of the hamiltonian path P_D of $UB(d,D)$, it was shown that if the path P^i begins at vertex $z_{D-1}\ldots z_1z_0$, if $\mathbf{x}\in P^i$ and $S(D-1,r)=k$, then $x_k=z_0$. Thus, we can compute the position of the node $\mathbf{x}$ on the path P_D since it is the rth node on the path P^i. This fact justifies the steps 4 and 5 of Algorithm 1. To get the rank of $\mathbf{x}$ on P_D, it is just sufficient to test the value of x_k.

If d is odd, then we compare the value of x_k to
- 0 if $\mathbf{x}\in P^0$,
- $(d-j\alpha_{D-1})$ if $\mathbf{x}\in P^{2j+1}$, with $j=0,1,\ldots,(d-3)/2$,
- $(j+1)\alpha_{D-1}$ if $\mathbf{x}\in P^{2j}$, with $j=0,1,\ldots,(d-1)/2$.

If d is even, then we compare the value of x_k to
- 0 if $\mathbf{x}\in P^0$,
- $2j$ if $\mathbf{x}\in P^{2j}$ (i.e. $x_k=i$ is even),
- $2j+1$ if $\mathbf{x}\in P^{2j+1}$ (i.e. $x_k=i$ is odd). □

We can invert Algorithm 1 in order to obtain the unranking algorithm as follows.

Algorithm 2 (*Unranking in* $UB(d,D)$).
Input: an integer r such that $0\leqslant r\leqslant d^D-1$.
Output: a word $\mathbf{x}=x_{D-1}x_{D-2}\ldots x_0\in UB(d,D)$.

Begin
1. Compute the quotient and the rest of euclidian division of r to d^{D-1}, i.e. $r = qd^{D-1} + r_0$, with $0 \leqslant r_0 < d^{D-1}$.
2. Compute recursively the label of the node $\mathbf{y} = y_{D-2}y_{D-3}\dots y_0$ of P_{D-1} with rank r' equal to r_0 if q is even and $d^{D-1} - 1 - r_0$ if q is odd.
3. Compute $S(D-1, r') = m$ by using Proposition 5.
4. Compute all the nodes $\mathbf{x} = x_{D-1}\dots x_1x_0$ such that $f(\mathbf{x}) = \mathbf{y}$.
5. Let $\mathbf{z} = z_{D-1}\dots z_1z_0$ be the first node of the path P^q.
The node $\mathbf{x}$ is the word which fulfils $x_m = z_0$.
End.

5. Reconfiguration algorithms for Kautz graphs

Using the preceding study and a vertex- and arc-uniform homomorphism of $K(d,D)$ into $B(d,D-1)$, we will now obtain ranking and unranking algorithms for Kautz graphs.

We consider the binary operation $\diamond$ defined by the table $M_\diamond = [m_{ij}]_{i=0,\dots,d}^{j=0,\dots,d}$ such that

$$m_{ij} = \begin{cases} i - j - 1 & \text{if } j < i \\ d - (j - i) & \text{if } j \geqslant i. \end{cases}$$

Note that each row and each column of the table $M_\diamond$ is a permutation of $\mathbb{Z}_{d+1}$ and that $m_{ii} = d$ for any $i \in \mathbb{Z}_{d+1}$. By Proposition 3, we can define a homomorphism $f_{(\diamond)}^D$ from $K(d,D)$ into $B(d,D-1)$ which satisfies $f_{(\diamond)}^D(x_{D-1}x_{D-2}\dots x_1x_0) = y_{D-2}\dots y_1y_0$, with $y_i = x_i \diamond x_{i+1} = (x_i \ominus x_{i+1}) - 1$, where $\ominus$ is the substraction modulo $d+1$. By Proposition 4, $f_{(\diamond)}^D$ is a surjective, vertex- and arc-uniform homomorphism. Therefore the preimage by $f_{(\diamond)}^D$ of the hamiltonian path P_{D-1} in $B(d,D-1)$ constructed in the preceding section is composed of $d+1$ vertex disjoint paths which are all isomorphic to P_{D-1}. We leave it to the reader to verify that the analogue of condition (C) is fulfilled so that the construction of a hamiltonian path in $UK(d,D)$ is possible in the same way as we have constructed P_D in $UB(d,D)$. Fig. 5 gives an illustration of this construction in $UK(4,2)$.

Ranking and unranking algorithms for Kautz graphs are therefore very similar to Algorithms 1 and 2. So we give them without comments (details can be found in [6]).

Algorithm 3 (*Ranking in $UK(d,D)$*).
Input: $\mathbf{u} = u_{D-1}u_{D-2}\dots u_0 \in V(UK(d,D))$.
Output: $rank(\mathbf{u}) \in \{0, 1, \dots, d^D + d^{D-1} - 1\}$.
Begin
Initialization: For $i = 0$ to d, $rank(i) = i$.
1. Compute $f_{(\diamond)}^D(\mathbf{u}) = v_{D-2}v_{D-3}\dots v_0 = \mathbf{v} \in V(UB(d,D-1))$,

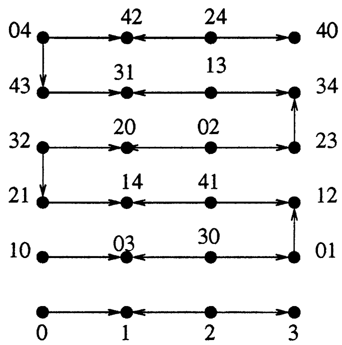

Fig. 5. Recursive construction of the hamiltonian path in $UK(4,2)$.

2. Compute, by using Algorithm 1, $rank(\mathbf{v}) = r$,
3. Compute, by using Proposition 5, $S(D-1, r) = k$,
4. Let $i = u_k$ ($\mathbf{u} \in P^i$),
5.

$$rank(\mathbf{u}) = \begin{cases} id^{D-1} + r & \text{if } i \text{ is even,} \\ (i+1)d^{D-1} - r - 1 & \text{otherwise.} \end{cases}$$

End.

We can invert Algorithm 3 in order to obtain the following unranking algorithm.

Algorithm 4 (*Unranking in $UK(d,D)$*).
Input: an integer r such that $0 \leqslant r \leqslant d^D + d^{D-1} - 1$.
Output: a word $\mathbf{x} = x_{D-1}x_{D-2}\ldots x_0 \in V(UK(d,D))$.
Begin
1. Compute the quotient and the rest of euclidian division of r to d^{D-1}, i.e. $r = qd^{D-1} + r_0$, with $0 \leqslant r_0 < d^{D-1}$ ($\mathbf{x}$ is the label of the vertex of rank r_0 on P^q).
2. Compute recursively the label of the node $\mathbf{y} = y_{D-2}y_{D-3}\ldots y_0$ of P_{D-1} with rank r' equal to

$$\begin{cases} r_0 & \text{if } q \text{ is even,} \\ d^{D-1} - 1 - r_0 & \text{if } q \text{ is odd.} \end{cases}$$

3. Compute $S(D-1, r') = m$, by using Proposition 5.
4. Compute the nodes $\mathbf{x}$ such that $f^D_{(\diamond)}(\mathbf{x}) = \mathbf{y}$.
5. Let $\mathbf{z} = z_{D-1}\ldots z_1z_0$ be the first node of P^q.
Then $\mathbf{x}$ satisfies $x_m = z_0$.
End.

6. Complexity of the algorithms

Let us analyze the serial complexity of Algorithm 1.

Let $T^b(D)$ be the number of elementary operations required for computing the rank of the node $\mathbf{x} = x_{D-1}x_{D-2}\dots x_0$ of $UB(d,D)$. Step 1 of the algorithm 1 needs the computation of $f(\mathbf{x})$, which takes $D-1$ elementary operations. Step 3 needs the computation of $S(D,r)$, which needs at most $2D-1$ operations. Finally, Step 2 is a recursive computation on the rank of $f(\mathbf{x})$. This step requires $T^b(D-1)$ operations. Note that the initialization step needs d operations. The global complexity of Algorithm 1 is bounded by the solution of the following recurrence relation:

$$T^b(1) = d,$$

$$T^b(D) = T^b(D-1) + 3D + \mathrm{O}(1) \quad \text{if } D > 1,$$

$$T^b(D) = d + 3\sum_{i=2}^{D} i + \mathrm{O}(D) = \mathrm{O}(D^2). \tag{1}$$

Similarly, we can prove that the complexity of Algorithms 2–4 is $\mathrm{O}(D^2)$.

7. Conclusion and further work

In this article, we have given ranking and unranking algorithms of efficient parallel embeddings of hamiltonian paths into d-ary de Bruijn and Kautz graphs. Our proofs are based on recursive constructions of hamiltonian paths in de Bruijn and Kautz graphs using suitable uniform homomorphisms of de Bruijn and Kautz graphs of diameter D on de Bruijn graphs of diameter $D-1$.

One can ask whether the method used here can be extended to other embedded graphs into de Bruijn and Kautz graphs, for example to cycles instead of paths. On the other hand, we can also ask for which classes of host graphs can the method give efficient ranking and unranking algorithms for hamiltonian paths.

References

[1] F.S. Annexstein, Parallel implementation of graph embeddings, in: Parallel Architectures and their Efficient Use, Lecture Notes in Computer Science, Vol. 678, Springer, Berlin, 1993, pp. 207–217.

[2] F.S. Annexstein, Ranking algorithms for Hamiltonian paths in hypercubic networks, in: DIMACS Workshop, Interconnection Networks and Mapping and Scheduling Parallel Computations, 1994 D.F. Hsu, A.L. Rosenberg, D. Sotteau (Eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, AMS, Vol. 21, 1995, pp. 1–7.

[3] J.-C. Bermond, C. Peyrat, De Bruijn and Kautz networks: a competitor for the hypercube?, in: F. André, J.P. Verjus (Eds.), Hypercube and Distributed Computers, Elsevier, North-Holland, 1989,, pp. 279–493.

[4] N.G. de Bruijn, A combinatorical problem, Koninklijke Nederlandsche Akademie van Wetenschappen Proc. A 49 (1946) 758–764.

[5] M.A. Fiol, J.L.A. Yebra, I. Alegre, Line digraph iterations and the (d, k) digraph problem, IEEE Trans. Comput. C-33 (5) (1984) 400–403.

[6] R. Harbane, Emulation et tolérance aux pannes dans certains réseaux d'interconnexion, Thèse, Université Paris-Sud, 1996.

[7] W.H. Kautz, Bounds on directed (d,k) graphs, Theory of cellular logic networks and machines, AFCRL-68-0668 Final Report, 1968, pp. 20–28.

[8] P. Tvrdík, R. Harbane, M.-C. Heydemann, Uniform homomorphisms and Divide & Conquer Emulations on de Bruijn and Kautz networks, Discrete Appl. Math. 83 (1998) 279–301.