De Bruijn graphs and their foldings

Peter J. Cameron
University of St Andrews

(Joint work with Collin Bleak and Feyishayo Olukoya)

Shanghai Jiao Tong University
November 2017

Universal circular sequences

De Bruijn graphs were introduced to solve the following problem:

Question

Given n and k, how can we create a cyclic arrangement of length n^k of the letters from an alphabet of size n, with the property that each k-tuple of letters from the alphabet occurs just once in consecutive positions in the cycle?

We will take the alphabet to be $\{0, 1, \ldots, n-1\}$.
For example, for $n = 3$ and $k = 2$, the sequence

$$(0, 0, 1, 1, 2, 0, 2, 2, 1)$$

has the required property.

De Bruijn graphs

The de Bruijn graph $G(n, m)$ is defined as follows:

- the vertices are all m-tuples of elements from the alphabet A of cardinality n;
- there is a directed arc labelled $a_0a_1 \ldots a_{m-1}a_m$ from the vertex $a_0a_1 \ldots a_{m-1}$ to vertex $a_1 \ldots a_{m-1}a_m$.

Each vertex of the graph has n arcs leaving it and n arcs entering it.
Since the graph is connected, it has a closed directed Eulerian trail. Reading around the trail gives the required circular sequence (with $k = m + 1$), since each k-tuple labels a unique edge and occurs once in the cycle.

Digression: a harder problem

This example is an experimental design problem from R. E. L. Aldred, R. A. Bailey, Brendan D. McKay and Ian M. Wanless, Circular designs balanced for neighbours at distances one and two, *Biometrika* **101** (2014), 943–956.

What if we want each ordered pair to occur once at distance 1 and once at distance 2 in the cycle?

It is easily checked that no such cycle exists for $n \leq 4$. The authors conjecture that it is true for all $n \geq 5$ and prove this in many special cases, including $n \leq 1000$.

The authors show that this is equivalent to constructing an Eulerian quasigroup of order n for each $n \geq 5$ (next slide).

Question

Does there exist an Eulerian quasigroup of any order $n \geq 5$?

Eulerian quasigroups

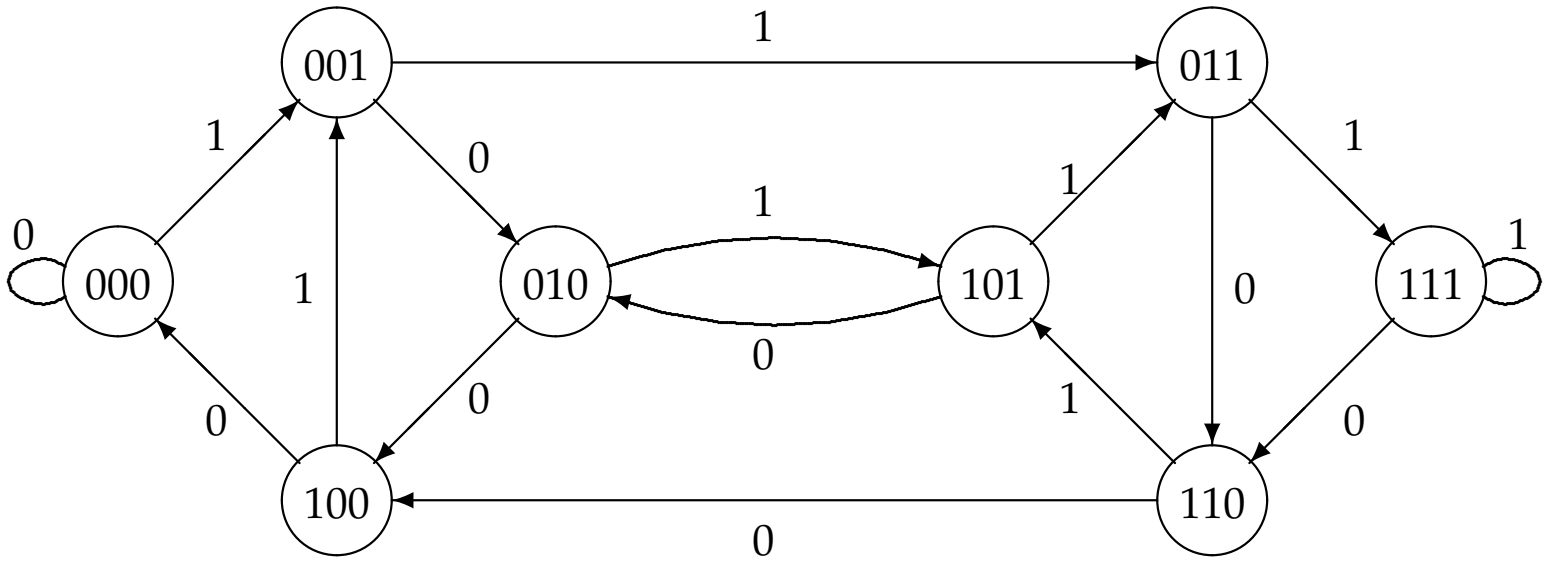
A quasigroup is an algebraic structure with a binary operation so that left division and right division are unique.
Given a quasigroup Q of order n, and two elements $a_0, a_1 \in Q$, form a Fibonacci sequence over Q by the rule that $a_m \circ a_{m+1} = a_{m+2}$ for $m \geq 0$. We say that the quasigroup is Eulerian if this sequence first returns to its starting point after n^2 steps.
Here is an example with $n = 5$.

$\circ$	0	1	2	3	4
0	1	0	2	3	4
1	2	3	1	4	0
2	3	4	0	2	1
3	0	2	4	1	3
4	4	1	3	0	2

$$(1,1,3,4,3,0,0,1,0,2,2,0,3,3,1,2,1,4,0,4,4,2,3,2,4)$$

An example

Back to the de Bruijn graphs. We can save space by labelling the arc from $a_0 \dots a_{m-1}$ to $a_1 \dots a_m$ just by the new symbol a_m added.



The picture shows the de Bruijn graph $G(2,3)$.

De Bruijn graphs as automata

A finite deterministic automaton is a machine M which has a finite set Q of internal states and reads symbols from a finite input alphabet A. It is described by a transition function $\pi : Q \times A \to Q$, so that when the machine is in state q and reads a symbol a, it moves to state $\pi(q, a)$.
An automaton can be represented by a finite edge-labelled directed graph, whose vertex set is Q, with arcs labelled by A; an edge $q \to r$ with label a indicates that $\pi(q, a) = r$. A digraph represents an automaton if and only if each vertex has a unique edge with each possible label leaving it.
Thus, the de Bruijn graph $G(n, m)$ represents an automaton whose state set is A^m and alphabet A, where $|A| = n$.

Synchronization

Am automaton is synchronizing if there is a string w of symbols in the alphabet A such that, after reading the symbols in A, the machine is in a state depending only on w and not on the initial state. Such a sequence is called a reset word.
The de Bruijn graph $G(n, m)$ represents an automaton with a very strong version of the synchronization property: every word of length m is a reset word. After reading the word w of length m, the automaton is in the state labelled w.
We say that an automaton with this property is synchronizing at level m.

Core synchronizing automata

Let M be an automaton which is synchronizing at level m.
There is a map $s : A^m \to Q$ such that, after reading a string w, the machine is in state $s(w)$. Let Q' be the image of s. Then $(Q', A, \pi|_{Q' \times A})$ is an automaton, called the core of M, and written $K(M)$.
We can think of the states in Q' as being recurrent, the others as being transient.
We say that an automaton M which is synchronizing at level m is a core automaton if $M = K(M)$.
A de Bruijn graph $G(n, m)$ represents a core automaton.

Foldings of automata

A folding of an automaton is an equivalence relation $\equiv$ on the set Q of states having the property that, if $q \equiv q'$ and a is any symbol in A, then $\pi(q,a) \equiv \pi(q',a)$.
If $\equiv$ is a folding of M, there is a quotient automaton $M/\equiv$ whose states are the $\equiv$-classes on Q, with an arc $[q] \to [r]$ with label a if $\pi(q',a) \in [r]$ for any $q' \in [q]$, where $[q]$ denotes the $\equiv$-class containing q.

De Bruijn graphs are universal

Theorem

Let $\mathcal{A}$ be an automaton over an alphabet A of length n. Then the following are equivalent:

- *$\mathcal{A}$ is synchronizing at level m, and is core;*
- *$\mathcal{A}$ is a folding of $G(n, m)$.*

The reverse direction is clear. For the forward direction, let $s : A^m \to Q$ be the map defined earlier. Since M is core, s is onto. Define a relation $\equiv$ on the vertex set of $G(n, m)$ by the rule that $w \equiv w'$ if $s(w) = s(w')$. Then verify that $\equiv$ is a folding, and $\mathcal{A}$ is isomorphic to the quotient $G(n, m)/\equiv$.

Universal algebra formulation

An automaton with alphabet A of size n can be regarded as an algebra on the set of states, n unary operations $\nu_0, \ldots, \mu_{n-1}$, where $q\nu_i = \pi(q, i)$ for all q, i.
Automata which are synchronizing at level m form a variety, defined by the laws

$$q\nu_{i_0} \cdots \nu_{i_{m-1}} = r\nu_{i_0} \cdots \nu_{i_{m-1}}$$

for all $q, r \in Q$ and $i_0, \ldots, i_{m-1} \in A$.
The core of the free 1-generator algebra in this variety is the de Bruijn graph $G(n, m)$.

Counting

"I count a lot of things that there's no need to count," Cameron said. "Just because that's the way I am. But I count all the things that need to be counted."

Richard Brautigan, *The Hawkline Monster: A Gothic Western*

Counting foldings

Let $F(n,m)$ be the number of foldings of the de Bruijn graph $G(n,m)$.

Question

Calculate the function $F(n,m)$.

By our previous comments, $F(n,m)$ is the number of n-state automata which are synchronizing at level m and are core.
It is clear that $F(n,1)$ is the Bell number $B(n)$. For in this case the vertices are indexed by symbols from the alphabet A; and, given an arbitrary partition of A, any arc labelled a ends in the part containing a.
We found a formula for $F(n,2)$. Beyond this, only finitely many values are currently known (by brute force computation): for example, $F(2,3) = 30$, $F(2,4) = 1247$.

A formula for $F(n,2)$

Theorem

The number of foldings of the de Bruijn graph with word length 2 *over an alphabet of cardinality n is*

$$\sum_{\pi} \prod_{i=1}^{|\pi|} R(|\pi|, |A_i|),$$

where π runs over partitions of the alphabet, A_i is the ith part, and

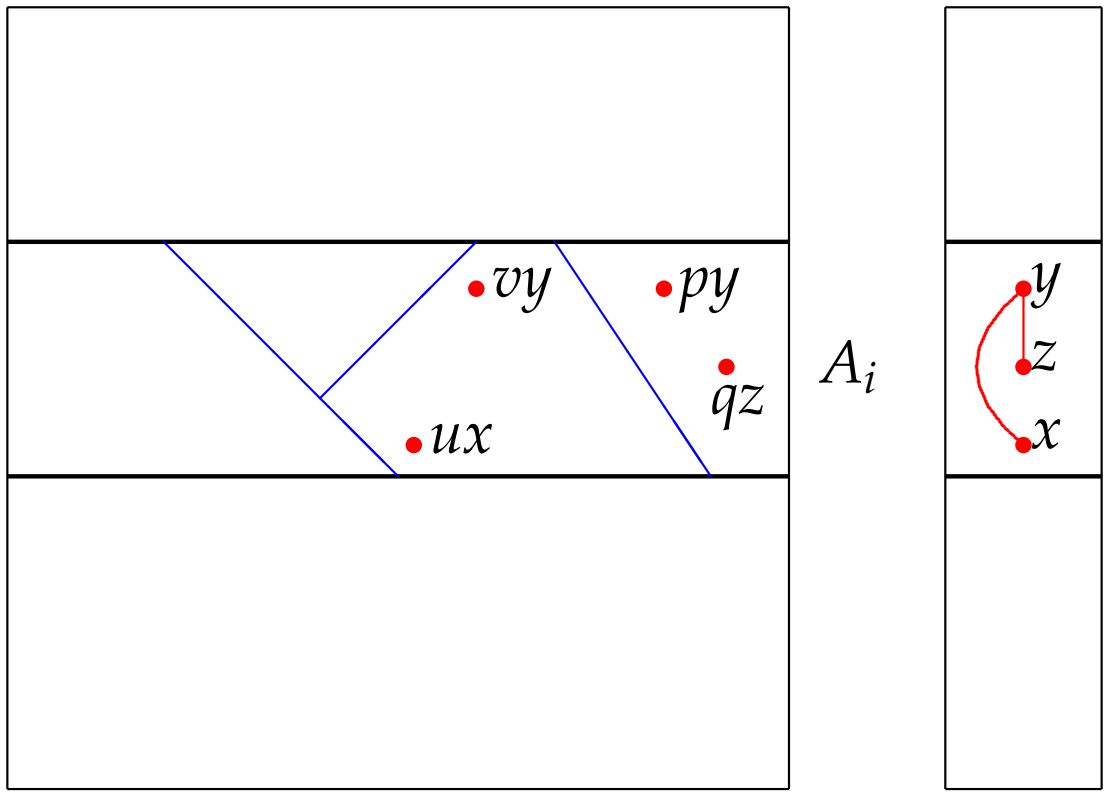
$$R(s,t) = \sum_{\pi} (-1)^{|\pi|-1} (|\pi| - 1)! \prod_{i=1}^{|\pi|} B(|A_i|s),$$

where π runs over all partitions of $\{1,\ldots,t\}$, and A_i is the ith part.

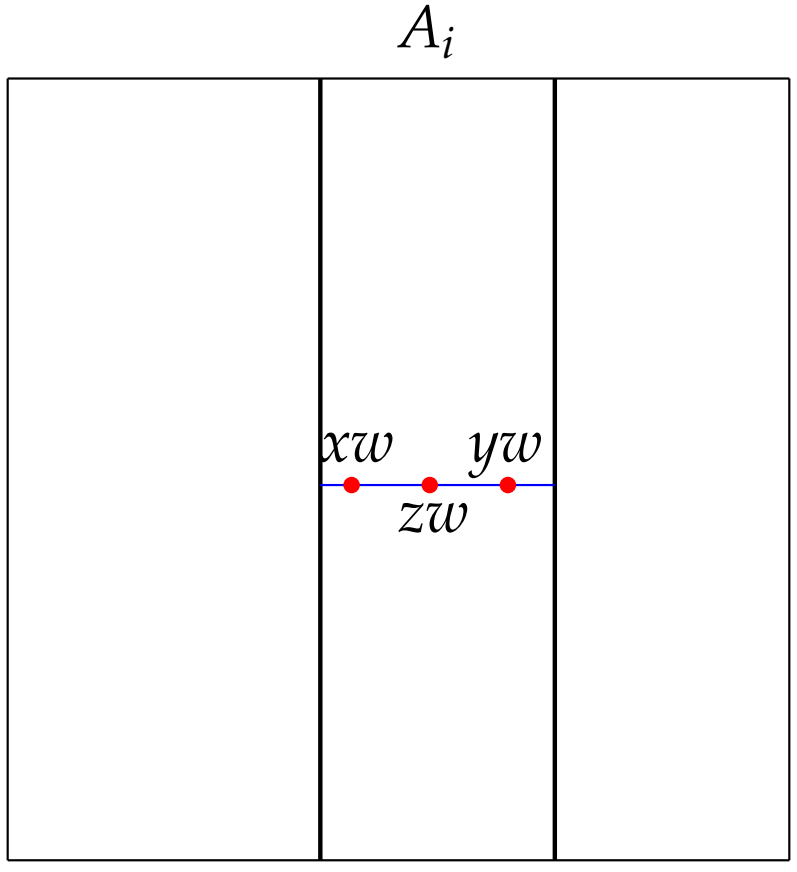
The numbers for $n = 1,\ldots,7$ are 1, 5, 192, 78721, 519338423, 82833228599906, 429768478195109381814.

Sketch proof

We define a graph Γ associated with a folding: the vertex set is the alphabet A, and two vertices x and y are joined if there exist u and v such that $ux \equiv vy$.



Let π be the partition of A into connected components of the graph Γ. If A_i is a part of Γ, then the set $A \times A_i$ (the horizontal stripe in the figure) is a union of parts of the folding: no part can cross into a different horizontal stripe.
Moreover, by the definition of a folding, we see that if $x, y \in A_i$, then xw and yw lie in the same part of the folding.



The sets $A \times A_i$ can be treated independently, so we have to count the number of good partitions of each and multiply them. Moreover, by the last remark, we can shrink each horizontal interval $A_j \times \{v\}$ to a point, so we have to partition $\pi \times A_i$. There are $B(|\pi| \cdot |A_i|)$ partitions of $\pi \times A_i$. We have to filter out the ones which do not induce partitions of $\pi \times B$ for any proper subset B of A_i. By Möbius inversion over the lattice of partitions of A_i, we find that the number of these is $R(|\pi|, |A_i|)$, where R is as defined earlier.
Putting all this together gives the result.

Automorphisms

Any permutation of the alphabet induces an automorphism of the de Bruijn graph (ignoring edge labels). This may induce an automorphism of a quotient of the graph by a folding (if it preserves the folding).

Theorem

The automorphism group of $G(n,m)$ (ignoring labels) is the symmetric group S_n.

Theorem

A folded de Bruijn graph over the 2-letter alphabet $\{0,1\}$ has at most two automorphisms; if there are two, then they are induced by interchanging the alphabet letters.

This depends on a result of interest in its own right:

Lemma

Suppose that a folding of $G(n,2)$ has the property that two vertices whose labels end with different letters are equivalent. Then there is just a single equivalence class.

Sketch proof

Let G be a folding of $G(2,m)$. Assume G has more than one vertex. Then two labels for the same vertex must end in the same letter, by the lemma. Also, using induction, we may assume the result for foldings of $G(2,m-1)$.

Write $v \sim w$ if the two out-neighbours of v and w are the same. By the lemma, two labels for the same vertex end with the same letter; so edges with a given label leaving equivalent vertices arrive at the same vertex. So $\sim$ is a folding. Vertices agreeing except in the first letter are equivalent; so $G/\sim$ is synchronizing at level $m-1$.

A graph automorphism g of G induces an automorphism $\bar{g}$ of $G/\sim$ which (by induction) is induced by a permutation of the alphabet. If $\bar{g}$ is trivial, then g fixes the vertex with label $00\ldots0$; considering a vertex moved by $\bar{g}$ whose distance from $00\ldots0$ is minimal, we reach a contradiction.

The other case is similar.

Transducers

The reason for our interest in foldings of de Bruijn graphs is that they are connected with interesting infinite groups, such as the outer automorphism groups of the finitely-presented Higman–Thompson simple groups, and the automorphism group of the shift dynamical system.
There is only time for a very brief sketch.
A transducer is an automaton with the extra ability that it can write strings from the alphabet A; that is, it has also an output function $\lambda : Q \times A \to A^*$, where A^* is the set of finite strings over A; if the machine is in state q and reads a, it writes $\lambda(q, a)$.
We always assume that *the transducer cannot read infinitely many symbols without writing something*. Equivalently, going round a cycle in the graph of the automaton results in some output being produced.

Maps of Cantor space

Let A^ω be the set of infinite sequences over A. We give A^ω the Tychonov product topology induced from the discrete topology on A.
From our above assumption, an initialised transducer M_q, that is, a transducer M which starts in state q, induces a map from A^ω to itself. It is easy to see that this map is continuous. Since A^ω is compact, if the map is invertible then it is a homeomorphism.
Maps of Cantor space induced by transducers which are synchronizing at some finite level, and whose inverses are also induced by transducers synchronizing at a finite level, are closely connected with automorphisms of the Higman–Thompson groups $G_{n,r}$.

Other groups

Many interesting groups can be defined as groups of maps of Cantor space induced by transducers of various types. The largest such group is the rational group of Grigorchuk, Nekrashevych, and Suschanskiĭ. We can restrict the to be synchronous (that is, they write one output symbol for each input symbol read), or synchronizing at some finite level, or having some "preliminary" states outside the core.
We hope that counting foldings will give group-theoretic information about some of these groups.
I refer the interested reader to our paper, arXiv 1605.09302.