Theory of Computing Systems manuscript No.
(will be inserted by the editor)

Kristoffer Arnsfelt Hansen

Constant Width Planar Computation Characterizes $\mathbf{ACC^0}$

the date of receipt and acceptance should be inserted later

Abstract We obtain a characterization of $\mathbf{ACC^0}$ in terms of a natural class of constant width circuits, namely in terms of constant width polynomial size planar circuits. This is shown via a characterization of the class of acyclic digraphs which can be embedded on a cylinder surface in such a way that all arcs flow along the same direction of the axis of the cylinder.

1 Introduction

This paper deals with the relationship between the computational power of *width* restricted circuits and *depth* restricted circuits. We relate constant width polynomial size *planar* circuits and also nondeterministic branching programs to constant depth polynomial size circuits.

Constant width polynomial size circuits (and branching programs) were shown to have surprising computational power by Barrington [2]. The class of functions computed by constant width polynomial size circuits is exactly $\mathbf{NC^1}$, and is thus considerably more than the functions computed by constant depth polynomial size circuits, being $\mathbf{AC^0}$.

Such connections are very interesting to explore, since they might provide the means for a better understanding of the classes involved, thus approaching lower bounds. Currently however, obtaining lower bounds for $\mathbf{NC^1}$ seems out of reach.

The smallest natural circuit class lacking lower bounds is $\mathbf{ACC^0}$, the subclass of $\mathbf{NC^1}$ computed by constant depth polynomial size circuits allowing MOD gates. By restricting the digraph representation of circuits geometrically, we obtain a characterization of $\mathbf{ACC^0}$ in terms of constant width circuits.

Theorem 1 *Constant width, polynomial size planar circuits compute exactly* $\mathbf{ACC^0}$.

Kristoffer Arnsfelt Hansen
Department of Computer Science. University of Aarhus, Denmark.
E-mail: arnsfelt@daimi.au.dk

Although Barrington and Thérien gave a characterization of $\mathbf{ACC^0}$ in terms of finite monoids [5] and Yao proved a nontrivial upper bound for $\mathbf{ACC^0}$ in terms of a class of threshold circuits [19], our result is the first alternative characterization of $\mathbf{ACC^0}$ by a circuit model. Recently Allender et al [1] have extended our characterization by showing that also constant width, polynomial size circuits of polylogarithmic *genus* only compute functions in $\mathbf{ACC^0}$.

Planarity has been previously employed in circuit lower bounds. While for general circuits, the best lower bounds for explicit functions are linear, superlinear lower bounds are known for general planar circuits. This was first obtained by Lipton and Tarjan based on their planar separator theorem [12], although their lower bound did not allow inputs to appear more than once, as we require in the above characterization. For general planar circuits allowing inputs to occur several times, superlinear lower bounds were proved by Turán, [16] and improved to the current best lower bound of $\Omega(n\log^2 n)$ by Gröger [8].

The circuits of Theorem 1 are restricted in two more aspects than planarity, namely that of constant width and of monotonicity in gate operations. Thus there is hope of obtaining much better lower bounds than just slightly superlinear. Furthermore one could hope that the geometric perspective on computation can lead to new ways of understanding the internal structure of $\mathbf{NC^1}$, thus obtaining progress towards separating $\mathbf{ACC^0}$ and $\mathbf{NC^1}$.

Much of the naturalness of $\mathbf{ACC^0}$ comes from the algebraic characterizations by Barrington and Thérien of the classes $\mathbf{AC^0}$, $\mathbf{ACC^0}$ and $\mathbf{NC^1}$ in terms of restrictions of finite monoids [5]. The class $\mathbf{AC^0}$ has also been characterized in terms of geometric restrictions by Vinay [17] and Barrington et al [3,4]. It was shown that the class of functions computed by constant width polynomial size *upward planar* circuits (and nondeterministic branching programs) is exactly $\mathbf{AC^0}$.

An intermediate geometric restriction between upward planarity and planarity was studied in [10]. There progress towards characterizing $\mathbf{ACC^0}$ was obtained by relaxing the geometrical restriction of upward planarity to that of cylindricality. While the exact relation to constant depth circuits is unknown under this restriction, it was shown that constant width polynomial size cylindrical circuits (and nondeterministic branching programs) can compute a strict superclass of $\mathbf{AC^0}$, while still only computing functions in $\mathbf{ACC^0}$. It is by building upon this result that we obtain Theorem 1.

The restrictions of upward planarity and cylindricality are similar in the sense that each layer of the circuit is drawn "together", in such a way that all arcs flow in a common direction. Under the restriction of planarity, nodes are in contrast allowed to be placed in an arbitrary way in the plane.

The results on the computational power of cylindrical circuits, as well as the characterization of $\mathbf{AC^0}$ in terms of upward planar circuits are actually based upon the algebraic characterizations by Barrington and Thérien[5]. Thus with the present characterization of $\mathbf{ACC^0}$ and the previous characterizations of $\mathbf{AC^0}$ in terms of geometric restrictions, the link between algebra and geometry in a computational setting seems very strong.

The key to applying the results on the computational power of cylindrical circuits for proving Theorem 1, is in identifying exactly which planar digraphs are cylindrical.

Theorem 2 *A layered digraph D is layered cylindrical if and only if it is a subdigraph of an acyclic planar layered digraph with a unique source and sink.*

This theorem (for general digraphs) is implicit in the works of Tamassia and Tollis [14] on *tessellation representations* of graphs in the plane and on a sphere. By "cutting" a digraph along a path from the source to the sink, they effectively reduce the spherical case (or equivalently the cylindrical case) to the planar case.

We give another proof of the theorem, working directly on the given planar embedding, taking advantage of the digraph being layered. This allows us to directly extract the combinatorial characterization of cylindricality used in [10] and also makes later uniformity considerations easier. With appropriate definitions of uniformity we can obtain the following uniform version of Theorem 1.

Theorem 3 *Logspace-uniform constant width, polynomial size planar circuits compute exactly logspace-uniform* $\mathbf{ACC^0}$.

While here, the motivation for exploring cylindrical representations of planar digraphs was the study of constant width computation, it is interesting to compare the above characterization with other previous representations of classes of planar graphs.

Mostly *undirected* graphs and various representations of those have been studied, also on cylinder surfaces [15]. For digraphs, the following characterization of upward planar digraphs was proved independently by Kelly [11] and Di Battista and Tamassia [6].

Theorem 4 (Kelly, Di Battista and Tamassia) *For a digraph D the following statements are equivalent.*

1. *D is a subdigraph of an acyclic planar digraph with a unique source and sink, which can be embedded in the plane with the source and sink both on the outer face.*
2. *D has an upward planar embedding, that is a planar embedding where all arcs are monotonically increasing in the upward direction.*
3. *D has an upward planar straight-line embedding*

There is a rough analogy between Theorem 4 (for upward planar digraphs) and Theorem 2 (for cylindrical digraphs). Let us make this analogy more precise. While Theorem 2 is stated in terms of layered digraphs, it easily generalizes to general digraphs by transforming any given digraph into a layered digraph by adding dummy nodes. Thus the difference between upward planar digraphs and cylindrical digraphs is precisely the condition that an acyclic planar superdigraph with a unique source and sink exists, which can in fact be embedded in the plane with the source and sink both on the outer face. To complete the analogy between Theorem 4 and Theorem 2, we can consider cylindrical embeddings with the arcs being parts of *geodesics*, the natural notion of straight lines on a cylinder surface. For layered digraphs such embeddings are easily obtained by the proof we will present for Theorem 2. For general digraphs the above subdivision argument does not give this kind of embedding. However, this is also obtainable, based on Di Battista and Tamassia's proof of Theorem 4 and by employing the "cutting" operation of [14].

Organisation of Paper

In Sect. 2 we introduce the notions of embeddings and circuits we will consider. We also state some basic properties about planar embeddings of certain digraphs. In Sect. 3 we prove Theorem 2, and in Sect. 4 we prove Theorem 1, as well as characterizing the computational power of planar branching programs. We introduce combinatorial embeddings in Sect. 5 as a means for dealing with uniformity issues and obtaining Theorem 3. We conclude with some discussions and open problems in Sect. 6.

2 Preliminaries

Bounded Depth Circuits

$\mathbf{AC^0}$ is the class of functions computed by polynomial size bounded depth circuits consisting of NOT gates and unbounded fanin AND and OR gates. $\mathbf{ACC^0}$ is the class of functions computed when we also allow unbounded fanin MOD gates computing MOD_m for constants m.

Planar and Cylindrical Embeddings of Digraphs

A digraph $D = (V, A)$ is called *layered* if there is a partition $V = V_0 \cup V_1 \cup \cdots \cup V_h$ such that each arc of A goes from *layer* V_i to the *next layer* V_{i+1} for some i. Given such a partition, we call h the *depth* of D, $|V_i|$ the width of layer i and $k = \max |V_i|$ the width of D.

A digraph is *planar* if it can be embedded in the plane. It is called *upward planar* if it can be embedded in the plane, in a way such that all arcs are monotonically increasing in the vertical direction. It is called *cylindrical* if it can be embedded on a cylinder surface, in a way such that all arcs are monotonically increasing in the direction of the axis of the cylinder.

Clearly every upward planar digraph is also cylindrical, and every cylindrical digraph is also planar.

We call a layered digraph *layered cylindrical* if it can be embedded on a cylinder surface, such that all arcs are monotonically increasing in the direction of the axis of the cylinder and that layers correspond to disjoint circles of the cylinder, which contain all the nodes of the layer.

In [13] the following properties of planar embeddings are proved. They are stated under the assumption of the digraph being 2-connected, but the proof does not use this assumption.

Lemma 5 *Let D be a planar acyclic digraph with a unique source and sink, and let $\widehat{D}$ be any planar embedding of D. Then*

1. *Each face of $\widehat{D}$ consists of two directed paths.*
2. *For any vertex v of $\widehat{D}$ all ingoing (outgoing) arcs of v appear consecutively around v.*

Bounded Width Branching Programs and Circuits

A *nondeterministic branching program* is an acyclic digraph where all arcs are labelled by either a literal, i.e. a variable or a negated variable, or a Boolean constant, and an initial and a terminal node. An input is accepted if and only if there is a path from the initial node to the terminal node in the digraph that results from substituting constants for the literals according to the input and then deleting arcs labelled by 0.

When referring to constant width branching programs we will mean a nondeterministic branching program, which when viewed as a digraph, is layered and of constant bounded width.

By a *constant width circuit* we mean a circuit consisting of fanin 2 AND and OR gates and fanin 1 COPY gates, which when viewed as a digraph (as with branching programs) is layered and of constant bounded width. Input nodes can be literals or Boolean constants, and can occur anywhere in the circuit.

Viewing nondeterministic branching programs and circuits as digraphs, it makes sense to restrict them geometrically. Since all cylindrical embeddings considered are layered cylindrical, we will usually just write cylindrical instead of layered cylindrical.

3 Embedding Digraphs on a Cylinder

In this section we will prove Theorem 2. The "only if" part is easily obtained: Consider a layered cylindrical embedding of a layered digraph D. By adding new arcs one can eliminate sources and sinks appearing in all layers except the first and the last. Now add a new first layer with a single node with arcs to all nodes in the previous first layer. Similarly add a new last layer with a single node with arcs from all nodes in the previous last layer. We have thus obtained an acyclic layered cylindrical superdigraph of D with a unique source and sink. The proof is completed by observing that every cylindrical embedding can be transformed into a planar embedding.

The "if" part is proved in several steps. First we will obtain a suitable partition of the planar embedding, and then later use this to obtain a new embedding on a cylinder surface.

Consider a planar embedding $\widehat{D}$ of a layered digraph D. Let $V(D) = V_0 \cup \cdots \cup V_h$ be the layers of D. Let C be a closed curve in the plane, partitioning the plane into two regions R_1 and R_2. We say that C is a *separating curve* for layer i if the following two properties hold.

1. The intersection of C and $\widehat{D}$ consists exactly of the nodes in V_i.
2. One of the regions R_j contains the embedding of subdigraph induced by $V_0 \cup \cdots \cup V_{i-1}$ as well as the arcs from V_{i-1} to V_i, and the other region contains the embedding of the subdigraph induced by $V_{i+1} \cup \cdots \cup V_h$ as well as the arcs from V_i to V_{i+1}

The following proposition shows that we can find separating curves for all layers in an acyclic planar layered digraph with a unique source and sink. This is illustrated in Fig. 1.

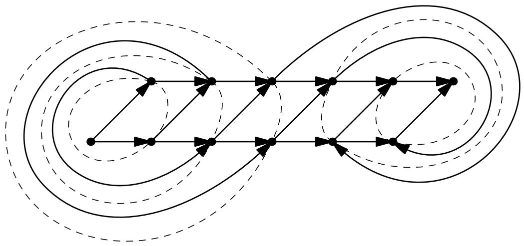

Fig. 1 Curves separating an acyclic planar layered digraph.

Proposition 6 *Let D be an acyclic planar layered digraph with a unique source and sink, with layers $V(D) = V_0 \cup \cdots \cup V_h$, and let $\widehat{D}$ be a planar embedding of D. Then there exist disjoint curves $C_1, \ldots, C_{h-1}$ such that C_i is a separating curve for layer i with respect to $\widehat{D}$ for all i.*

Proof We will construct the separating curves by an iterative process. Assume that we have found separating curves $C_1, \ldots, C_{i-1}$.

Let v be a node in D which is neither the source nor the sink. That is, v has at least one ingoing arc and one outgoing arc, and by Lemma 5 all the ingoing arcs and outgoing arcs appear consecutively around v in $\widehat{D}$. It is then meaningful to talk about the rightmost ingoing (outgoing) arc, and the leftmost ingoing (outgoing) arc.

Since the ingoing arcs to V_{i-1} and outgoing arcs from V_{i-1} are separated by C_{i-1} it follows, that when traversing C_{i-1} from a node between the leftmost ingoing and outgoing arcs the next node is approached between the rightmost ingoing and outgoing arcs.

We now find a separating curve C_i for layer i by the following process: Start in an arbitrary node v in layer i. Follow along the left of the leftmost ingoing arc a to a node w in layer $i-1$. If a is the leftmost outgoing arc of w we follow the curve C_{i-1} to the next node w' and follow the rightmost arc to a node v' in layer i. Otherwise we follow the next outgoing arc (in the counterclockwise order) to a node v' in layer i. Since both v and v' belong to the boundary of the face of $\widehat{D}$ we are within and because they belong to the same layer of D, it follows from Lemma 5 that v' is approached along the rightmost ingoing arc.

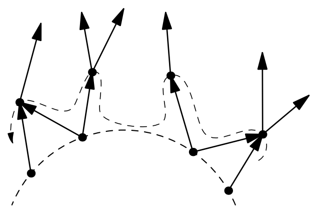

Fig. 2 Finding separating curves.

We now continue the same process, as illustrated in Fig. 2, until a closed curve C_i is found. It is clear that C_i has the following properties:

1. C_i only intersects $\widehat{D}$ in nodes from V_i.
2. C_i is disjoint from $C_1, \ldots, C_{i-1}$.
3. The region R partitioned by C_i containing C_{i-1} does not contain nodes from layers $i, \ldots, h$.

of which the last holds because D has a unique sink.

Now, since all nodes in V_i have an ingoing arc, and since they are not in the region R containing C_{i-1} by (3), it follows from (1) that C_i intersects *exactly* the nodes in V_i. From this and (3) it follows that C_i is a separating curve for layer i. □

The next step is to associate an orientation (clockwise or counterclockwise) to each of the separating curves. These will give the order in which the nodes of each layer are to be drawn around a circle of the cylinder.

We give the first curve C_1 the counterclockwise orientation. Now assume we have assigned orientations to $C_1, \ldots, C_{i-1}$. We assign an orientation to C_i based on whether it contains C_{i-1} or vice versa.

There will be three cases as illustrated in Fig. 3:

(a). C_{i-1} is inside the region bounded by C_i.
(b). The regions bounded by C_{i-1} and C_i are disjoint.
(c). C_i is inside the region bounded by C_{i-1}.

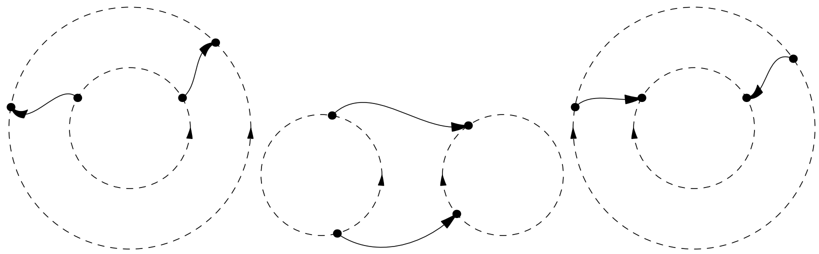

Fig. 3 Orienting the separating curves.

In cases (a) and (c) we assign the same orientation to C_i as C_{i-1}. In case (b) we assign the opposite orientation to C_i as C_{i-1}.

From the properties of separating curves it follows that case (b) can only occur once. In fact there is a k such that if $i < k$, C_i is oriented counterclockwise and if $i > k$, C_i is oriented clockwise.

For nodes a and b on a curve C, define the *segment from a to b*, to be the segment of C traversed by following C from a to b (inclusive) along the orientation associated to C.

The crucial property is now the following: Let a and b be nodes of layer $i-1$ with arcs to nodes c and d of layer i respectively. Then all nodes (except a and b) on the segment from a to b have arcs only into the segment from c to d (see Fig. 3). This is in fact, basically the characterization of cylindricality that was used in [10].

In fact, by this property, it follows that in Proposition 6, the separating curves found are actually traversed according to the orientations assigned above.

Furthermore, assign an ordering of the outgoing (ingoing) arcs from (to) a layer by traversing the separating curve according to the associated orientation, and including the outgoing arcs (ingoing arcs) of each node in counterclockwise (clockwise) order. Then it follows from the above property that the ordering of the outgoing arcs from a layer coincide with the ordering of the next layer, where the arcs are ingoing.

We are now in position to create a layered cylindrical embedding of D. The nodes in layer i are placed in a circle around the cylinder in the order they are met, when traversing the curve C_i according to the associated orientation, and the arcs between layers are simply drawn in the order described above.

This completes the embedding and the proof of Theorem 2.

4 Planar Branching Programs and Circuits

In this section we characterize the computational power of planar circuits (and nondeterministic branching programs). First we show how to compute any $\mathbf{ACC^0}$ function by a constant width polynomial size planar circuit.

The core of the simulation is the following substitution lemma for planar circuits. Using this we only need to show how to compute AND, OR and MOD_m by planar circuits, to establish that planar circuits can compute all of $\mathbf{ACC^0}$.

Lemma 7 *If $f(x_1,\ldots,x_n)$ is computed by a planar circuit of size s_1 and width w_1 and $g_1,\ldots,g_n$ are computed by planar circuits each of size s_2 and width w_2 then $f(g_1,\ldots,g_n)$ is computed by a planar circuit of size $O(s_1 s_2^2)$ and width $O(w_1 w_2)$.*

Proof By exchanging AND with OR gates and vice versa and by negating inputs, $\bar{g}_1,\ldots,\bar{g}_n$ are also computed by planar circuits of size s_2 and width w_2. Consider any planar embedding of a circuit C for f and choose planar embeddings of circuits $C_1,\ldots,C_n$ and $\bar{C}_1,\ldots,\bar{C}_n$ for $g_1,\ldots,g_n$ and $\bar{g}_1,\ldots,\bar{g}_n$ with the output gate appearing on the outer face.

We now stretch each layer of C into s_2 layers, by replacing each arc by a string of $s_2 - 1$ COPY gates, preserving planarity of the embedding. This ensures that input nodes in different layers are at least s_2 layers apart, and only increases the size by a factor $O(s_2)$ and the width by a factor 2.

This now allows us to simply substitute (after appropriate shrinking if necessary) the embedding of C_i for x_i and the embedding of $\bar{C}_i$ for $\bar{x}_i$ in the embedding of C preserving the planarity, without increasing the width by more than a factor $O(w_2)$. □

In [10] the construction for AND, OR and MOD_m is given in terms of cylindrical branching programs, and hence also in terms of planar circuits. For completeness we provide the details of the constructions directly in terms of planar circuits here.

The AND (and OR) of n inputs is easily computed by a planar circuit of width 2 and size $O(n)$ as shown in Fig. 4.

The construction for MOD_m is more complicated, an example of it is shown in Fig. 5.

It can be computed by a planar circuit of width $O(m)$ and size $O(mn)$ constructed as follows: We will have $2n+1$ layers. The first layer consists of a single

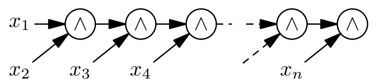

Fig. 4 A planar circuit computing AND.

input gate with the constant 1. In the last $2n$ layers the main part consists of m sufficiently long strings, which we number $0,\ldots,m-1$, of alternating AND and OR gates taking each other as input, the AND gates in odd layers and the OR gates in even layers.

String 0 starts with an AND gate and all other strings start with an OR gate. The constant input 1 in layer 0 will take the place of an OR gate in string 0. The construction will ensure that the OR gate in layer $2i$ of string j evaluates to 1 if and only if $\sum_{k=1}^{i} \equiv j \pmod m$.

The first AND gate in string 0 takes the constant input 1 as input. The first OR gate in the other strings takes a constant input 0 as input. The AND gate in layer $2i+1$ of every string takes $\bar{x}_i$ as the other input. The first $m-1$ OR gates in string 0 take a constant input 0 as the other input. The other OR gates in string 0, say in layer $2i$, take an AND gate as the other input, which is fed by x_i and the OR gate in layer $2(i-1)$ of string $m-1$. In general the OR gate in layer $2i$ of string j takes an AND gate as the other input, which is fed by x_i and the OR gate in layer $2(i-1)$ of string $j-1$. The output gate is the last OR gate of string 0.

Note, that while the above constructions are cylindrical, once we use Lemma 7 to substitute a circuit computing MOD_m on more than m inputs, we leave the class of cylindrical circuits.

To conclude, we can from an $\mathbf{ACC^0}$ circuit construct a constant width polynomial size planar circuit computing the same function as follows: Expand the $\mathbf{ACC^0}$ circuit into an $\mathbf{ACC^0}$ formula, and move the NOT gates to the layer just above the inputs by the usual constructions. Now proceeding layer by layer, we build appropriate components for the AND, OR and MOD_m gates, and compose these using the substitution lemma.

We now prove the last part of Theorem 1, that constant width, polynomial size planar circuits only compute functions in $\mathbf{ACC^0}$.

We need the following result from [10][1].

Theorem 8 *Every Boolean function computed by a constant width, polynomial size cylindrical circuit is in* $\mathbf{ACC^0}$.

This result was proved by showing that the *word problem* for cylindrical circuits, i.e. to compute the output of a given cylindrical circuit on a given input, is in $\mathbf{ACC^0}$. From this we have the following corollary.

Corollary 9 *Let p be a polynomial and w any constant. Then there is a polynomial q and a constant h such that every Boolean function on n inputs computed by a cylindrical circuit of size $3p(n)$ and width $2w$ is computed by an* $\mathbf{ACC^0}$ *circuit of size $q(n)$ and depth h.*

[1] Only an overview of the proof of this result was given. The full proof can be found in the technical report version of the paper [9].

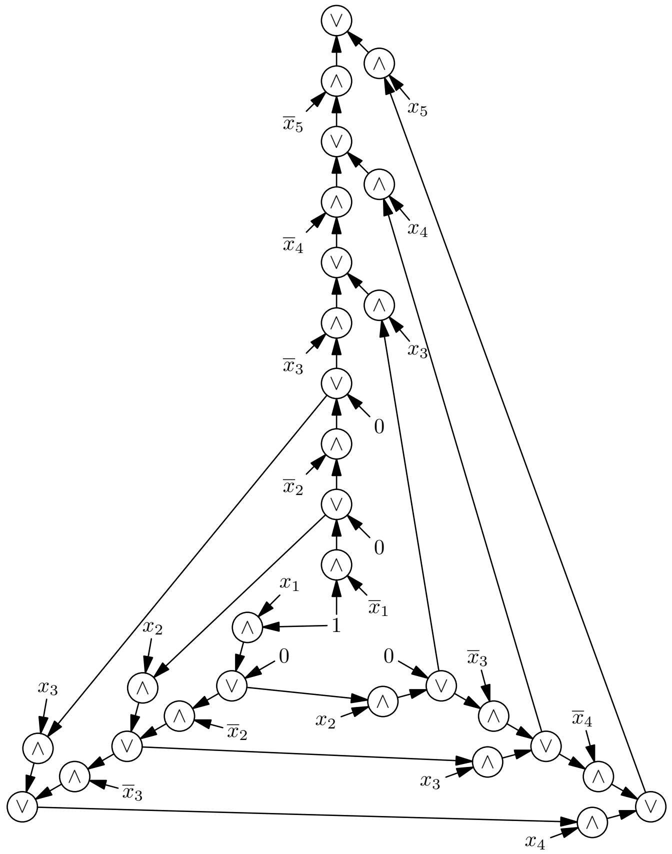

Fig. 5 A planar circuit of width 9 computing MOD_3 on 5 inputs.

Now consider a planar circuit C on n inputs of size at most $p(n)$ and width at most w. Let q and h be the polynomial and the constant from Corollary 9. We show that C is computed by an $\mathbf{ACC^0}$ circuit of size $q(p(n))^w$ and depth wh, assuming without loss of generality that p and q are increasing functions and it holds that $1 + p(n) \leq q(p(n))$.

If the width of C is in fact 1, then the function computed by C is certainly also computed by an $\mathbf{ACC^0}$ circuit of size $q(p(n))$ and depth h.

Otherwise, pick a node v in the first layer of C and let D be the subdigraph of the digraph representation of C, which is induced by the nodes which are on a path from v to the output node of C.

Removing the nodes in D, note that the remaining components are digraph representations of planar circuits. There are at most $p(n)$ of these each of size $p(n)$, but with width $w-1$, since at least one node is removed from each layer. By induction, all these are computed by $\mathbf{ACC^0}$ circuits of size $q(p(n))^{w-1}$ and depth $(w-1)h$.

Now by Theorem 2, D is cylindrical, and we construct a cylindrical embedding of it. For all AND and OR gates that have indegree 1 in D we will add a new input variable in the previous layer. In order to preserve layered cylindricality, we first stretch each layer into two layers using COPY gates. Together this will yield a cylindrical circuit of size at most $3p(n)$ and width at most $2w$ on at most $p(n)$ inputs, which by Corollary 9 is computed by an $\mathbf{ACC^0}$ circuit of size $q(p(n))$ and depth h. Now substitute the functions constructed inductively in this circuit to obtain an $\mathbf{ACC^0}$ circuit of size $q(p(n)) + p(n)q(p(n))^{w-1} \leq q(p(n))^w$ and depth wh computing the function computed by C. With this we have completed the proof of Theorem 1.

Turning to the computational power of planar nondeterministic branching programs, we can easily characterize this by Theorem 2. Indeed we can assume, without loss of generality that every node is on a path from the initial node to the terminal node, and hence Theorem 2 applies to give the following theorem.

Theorem 10 *Constant width, polynomial size planar branching programs compute the same Boolean functions as constant width, polynomial size cylindrical branching programs.*

As noted in [10] this class of functions does not seem to capture all of $\mathbf{ACC^0}$. The compositional approach that worked so well for planar circuits using Lemma 7 fails for planar branching programs: in order to substitute a branching program for an arc, preserving planarity, one would need an embedding of it with both the initial and the terminal node on the external face. By Theorem 4 these would then necessary be upward planar, and thus by the results of Barrington et al [3], substituted branching programs can only compute $\mathbf{AC^0}$ functions.

5 Uniformity Considerations

When considering uniformity, we need to be more precise when we talk about embeddings. We will employ the concept of combinatorial planar embeddings based on Edmonds' permutation technique [7] (see also [18] pages 70–73).

Combinatorial embeddings are most conveniently introduced for undirected graphs. Let $G = (V, E)$ be an undirected graph. For a vertex u define the set of neighbours $N(u) = \{v \mid \{u, v\} \in E\}$ and let $A = \{uv \mid \{u, v\} \in E\}$ be the set of all oriented arcs obtained from E. Now let $p_u : N(u) \to N(u)$ be a cyclic permutation of the neighbours of u. Define $P : A \to A$ by $P(uv) = vp_v(u)$. Observe that P is a permutation of A.

As Edmonds now observed, there is a one-to-one correspondence between choices of $\{p_u\}$ and *2-cell embeddings* of G into closed orientable surfaces, where faces of the embedding correspond to the orbits of P.

We can thus say that $\{p_u\}$ is a combinatorial planar embedding of G if and only if Euler's formula $v - e + f = c + 1$ is satisfied, where $v = |V|$, $e = |E|$, f is the number of orbits in P and c is the number of connected components of G.

A description of a combinatorial planar embedding of a digraph D then simply consists of having for each vertex a list of the neighbours, in clockwise order around the vertex, say, according to an embedding. A description of a layered cylindrical embedding consists of an ordering of the nodes in each layer, corresponding to their order around the cylinder in an embedding.

We thus say that a family of planar circuits or cylindrical circuits are logspace-uniform, if there is a $O(\log n)$ space bounded Turing machine which on input 1^n outputs the description of the circuit on n inputs as well as the above defined description of the embedding.

Below we will sketch how to obtain Theorem 3 using these definitions, by adapting the proof of Theorem 1 appropriately. In our description we will have intermediate results and use recursion to a constant recursion depth. The actual logspace transducers will never store these results but instead recompute them as needed.

First we will show how to convert a logspace-uniform family of $\mathbf{ACC^0}$ circuits into a corresponding logspace-uniform family of planar circuits by a logspace transducer. We will need the fact that this is easily done for the AND, OR and MOD_m gates.

The transducer will be a double recursive algorithm in the given circuit. We perform a depth-first traversal, and when reaching a new gate we will in effect be implementing Lemma 7. First, by recursing on each of the inputs to the gate, we will calculate the maximal size s of any subcircuit connected to an input, when converted to a planar circuit by the algorithm. Next we will generate the appropriate planar construction corresponding to the type of the gate, while stretching each layer into s layers using COPY gates. Whenever we are to generate an input, we instead recurse on the corresponding subcircuit. We can handle negated inputs by passing on to the recursive call whether AND and OR gates need to be exchanged and inputs to be negated.

To convert a logspace-uniform family of planar circuits into a corresponding logspace-uniform family of $\mathbf{ACC^0}$ circuits by a logspace transducer we need a suitable uniform version of Theorem 8.

Theorem 11 *For any fixed constant width w, there exists a logspace transducer converting a logspace-uniform family of width w polynomial size cylindrical circuits into a logspace-uniform family of* $\mathbf{ACC^0}$ *circuits.*

Uniformity was not considered in [10], but the construction there is easily seen to be uniform (even under stronger notions of uniformity).

We will also need to extract cylindrical circuits from parts of planar circuits in logspace.

Proposition 12 *For any fixed constant width w, there exists a logspace transducer which, given a width w planar circuit C along with the description of an embedding and distinguished gates g and g', creates a constant width cylindrical circuit, along with the description of a layered embedding, obtained by considering the part C' of C consisting of gates on a path from g to g', creating a cylindrical embedding of it, stretching each layer into two and adding new input variables to AND and OR gates having indegree 1 (As in the proof of Theorem 1).*

The proposition is basically just a realization of Proposition 6 in logspace. To find the gates in C' we will proceed layer by layer starting from g to find all gates reachable from g. Similarly we proceed backwards layer by layer from g' to find out from which of those gates g' can also be reached. Since the width is constant, we can store and manipulate entire layers of C' at the same time. The ordering of a layer is readily obtained from the ordering of the previous layer, by following the proof of Proposition 6. Stretching and the addition of extra input variables can be done while doing this.

We can now complete the construction be following the last part of the proof of Theorem 1. We identify a gate in the first layer and use Proposition 12 on this and the output gate to obtain a cylindrical circuit. This is then converted into an $\mathbf{ACC^0}$ circuit using Theorem 11, and recursively we can construct $\mathbf{ACC^0}$ circuits for the remaining part of the circuit which are then substituted to give the final $\mathbf{ACC^0}$ circuit.

6 Conclusion

We have obtained a characterization of $\mathbf{ACC^0}$ in terms of a geometrical restriction of the digraph representation of circuits. Together with the previous characterizations of $\mathbf{AC^0}$ this shows a striking similarity to the algebraic characterizations by Barrington and Thérien, as summarized in the following table.

Circuit Class	$\mathbf{AC^0}$	$\mathbf{ACC^0}$	$\mathbf{NC^1}$
Nonuniform Automata on Monoids	Aperiodic	Solvable	Unrestricted
Constant Width Branching Programs	Upward planar	?	Unrestricted
Constant Width Circuits	Upward planar	Planar	Unrestricted

It would be very interesting to further investigate the link between algebra and geometry in this setting. Intuitively, a cylindrical circuit corresponds in some sense to an $\mathbf{ACC^0}$ circuit with just one layer of MOD gates. One could hope to explain this by tightening this link. In [10] a $\mathbf{\Pi}_2 \circ \mathbf{MOD} \circ \mathbf{AC^0}$ lower bound and an $\mathbf{ACC^0}$ upper bound was proved. Perhaps one could give a seemingly better upper bound than $\mathbf{ACC^0}$, for example $\mathbf{AC^0} \circ \mathbf{MOD} \circ \mathbf{AC^0}$.

We don't have a characterization of $\mathbf{ACC^0}$ in terms of geometric restrictions of branching programs, yet they remain attractive for their simplicity. It might very well be within reach to obtain lower bounds for constant width planar branching programs, providing a first step for employing planarity in lower bounds for $\mathbf{ACC^0}$.

Acknowledgements I am grateful to my advisor Peter Bro Miltersen for introducing me to constant width computation and for helpful comments and suggestions.

The author is supported by BRICS, Basic Research in Computer Science, a Centre of the Danish National Research Foundation.

References

1. E. Allender, S. Datta, and S. Roy. Topology inside $\mathbf{NC}^1$. In *20th Annual IEEE Conference on Computational Complexity*. IEEE Computer Society Press, 2005. (to appear).
2. D. A. Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in $\mathbf{NC}^1$. *J. Comput. System Sci.*, 38(1):150–164, 1989.
3. D. A. M. Barrington, C.-J. Lu, P. B. Miltersen, and S. Skyum. Searching constant width mazes captures the $\mathbf{AC}^0$ hierarchy. In *Proceedings of the 15th Annual Symposium on Theoretical Aspects of Computer Science*, pages 73–83, 1998.
4. D. A. M. Barrington, C.-J. Lu, P. B. Miltersen, and S. Skyum. On monotone planar circuits. In *14th Annual IEEE Conference on Computational Complexity*, pages 24–31. IEEE Computer Society Press, 1999.

5. D. A. M. Barrington and D. Thérien. Finite monoids and the fine structure of $\mathbf{NC}^1$. *Journal of the ACM (JACM)*, 35(4):941–952, 1988.
6. G. Di Battista and R. Tamassia. Algorithms for plane representations of acyclic digraphs. *Theoretical Computer Science*, 61(2-3):175–198, 1988.
7. D. Edmonds. A combinatorial representation for polyhedral surfaces. *Notices Amer. Math. Soc.*, 7:646, 1960.
8. H. D. Gröger. A new partition lemma for planar graphs and its application to circuit complexity. In *8th International Symposium on Fundamentals of Computation Theory*, volume 529 of *Lecture Notes in Computer Science*, pages 220–229. Springer, 1991.
9. K. A. Hansen, P. B. Miltersen, and V Vinay. Circuits on cylinders. Technical Report 66, Electronic Colloquium on Computational Complexity, 2002.
10. K. A. Hansen, P. B. Miltersen, and V Vinay. Circuits on cylinders. In *14th International Symposium on Fundamentals of Computation Theory*, volume 2751 of *Lecture Notes in Computer Science*, pages 171–182. Springer, 2003.
11. D. Kelly. Fundamentals of planar ordered sets. *Discrete Mathematics*, 63(2,3):197–216, 1987.
12. R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM Journal on Computing*, 9(3):615–627, August 1980.
13. R. Tamassia and I. G. Tollis. A unified approach to visibility representations of planar graphs. *Discrete & Computational Geometry*, 1(1):312–341, 1986.
14. R. Tamassia and I. G. Tollis. Tessellation representations of planar graphs. In *Proceedings 27th Annual Allerton Conference on Communications, Control and Computing*, pages 48–57. University of Illinois at Urbana-Champaign, September 1989.
15. R. Tamassia and I. G. Tollis. Representations of graphs on a cylinder. *SIAM Journal on Discrete Mathematics*, 4(1):139–149, 1991.
16. G. Turán. On restricted boolean circuits. In *7th International Symposium on Fundamentals of Computation Theory*, volume 380 of *Lecture Notes in Computer Science*, pages 460–469. Springer, 1989.
17. V Vinay. Hierarchies of circuit classes that are closed under complement. In *11th Annual IEEE Conference on Computational Complexity*, pages 108–117. IEEE Computer Society, 1996.
18. A. T. White. *Graphs, Groups and Surfaces*. Elsevier Science Publishers B.V, 1984.
19. A. C.-C. Yao. On $\mathbf{ACC}^0$ and threshold circuits. In *Proceedings 31st Annual Symposium on Foundations of Computer Science*, pages 619–627. IEEE Computer Society Press, 1990.