Characterization of Associative Operations with Prefix Circuits of Constant Depth and Linear Size

Gianfranco Bilardi[1]
Franco P. Preparata[2]

88-911
April 1988

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

[1] Department of Computer Science, Cornell University, Ithaca, NY 14853. Research supported in part by NSF Grant DCI-8602256
[2] Coordinated Science Laboratory, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, IL 61801, USA. Research supported in part by NSF Grant ECS-84-10902.

CHARACTERIZATION OF ASSOCIATIVE OPERATIONS WITH PREFIX CIRCUITS OF CONSTANT DEPTH AND LINEAR SIZE

G. Bilardi[1] and F.P. Preparata[2]

Abstract. The *prefix problem* consists of computing all the products $x_0x_1...x_j$ (j=0,...,$N-1$), given a sequence x = (x_0,x_1,...,x_{N-1}) of elements in a semigroup. It is shown that there are unbounded fan-in and fan-out *boolean circuits* for the prefix problem with *constant depth* and *linear size* if and only if the Cayley graph of the semigroup does not contain a special type of cycle called *monoidal cycle*.

1 Introduction

The *prefix problem* consists of computing all the products $x_0x_1...x_j$ (j=0,...,$N-1$), given a sequence x = (x_0,x_1,...,x_{N-1}) of elements in a semigroup. Prefix computations occur in the solution of several significant problems such as the evolution of finite-state machines [O63, LF80], binary addition [O63, BK82], linear recurrences [K78], digital filtering [BP86], various graph problems [KRS85], sorting in bit-models of computation [CS85, BP85], and scheduling [DS83].

The prefix problem has been extensively investigated (see [P87] for a survey). Various complexity measures such as *size*, *depth*, *width*, and their trade-offs have been studied in [LF80, F83, S86] for *circuits* of *semigroup multipliers*. Algorithms for the EREW-PRAM model have been proposed in [KRS85, R84, CV86]. Implementations on a *tree-connected* network are discussed in [DS83]. *Size-time* trade-offs for *boolean networks*, which are synchronized interconnections of boolean gates and one-bit storage devices, have been characterized in [BP87]. *Size-depth* trade-offs for *boolean circuits* with NOT, OR, and AND gates of unbounded fan-in and fan-out have been studied in [CFL83a, CFL83b], and will also be the subject of this paper.

It is interesting to observe that the prefix problem reveals different facets in different models of computation. The structure and the complexity of prefix circuits are oblivious to the nature of the underlying semigroup if semigroup multipliers are assumed as the primitive modules, whereas they depend on the nature of the semigroup if the primitive modules are boolean gates.

[1] Department of Computer Science, Cornell University, Ithaca, NY 14853. Research supported in part by NSF Grant DCI-8602256.

[2] Coordinated Science Laboratory, Departments of Electrical and Computer Engineering and of Computer Science, University of Illinois at Urbana-Champaign, IL 61801, USA. Research supported in part by NSF Grant ECS-84-10902.

In the present paper we study the size of prefix boolean circuits of constant depth, a subject whose investigation begun in [CFL83a, CFL83b]. The goal is the identification of the semigroup properties that determine the size complexity.

In [CFL83a] it is shown that the size of constant-depth prefix circuits for non group-free semigroups is exponential, as a simple consequence of known bounds on the size of constant-depth circuits for the parity function [FSS81, Y85, H86]. For group-free semigroups, a construction is provided that achieves slightly superlinear size. Indeed, given any primitive recursive function $f(N)$, the construction can be tuned to achieve size $O(N f^{-1}(N))$ and depth constant with N (but depending upon f and upon the underlying semigroup).

In [CFL83b] the above bound is shown to be tight for the *carry semigroup*, whose prefix circuit computes the carries in binary addition. The lower bound is obtained by showing that any prefix circuit for the carry semigroup contains a *weak superconcentrator*, and hence satisfies a lower bound on the size of constant-depth weak superconcentrators due to [DDPW83]. Finally, [CFL83b] show that not all group-free semigroups require superlinear size by constructing a linear-size prefix circuit for the OR semigroup.

Given these results, it is natural to ask which semigroups have linear-size constant-depth prefix circuits. In this work, we introduce the notion of *monoidal cycle*, a special type of cycle in the Cayley graph of a semigroup, and we show that there are constant-depth linear-size prefix circuits for a semigroup if and only if its Cayley graph has no monoidal cycle.

The remainder of this paper is organized as follows. Section 2 contains the necessary technical definitions about semigroups. Section 3 contains a superlinear lower bound for semigroups with monoidal cycles. Section 4 gives a linear upper bound for semigroups without monoidal cycles. The latter is achieved by a non trivial construction and is the main contribution of the paper. Finally, in Section 5, we compare the classification of semigroups related to the size complexity of constant-depth prefix circuits with the classification of semigroups related to the size-time complexity of prefix boolean networks [BP87]. We conclude with some open problems.

2 Preliminaries

A *finite semigroup* is a pair $<A, \cdot>$ where $A = \{a_1, a_2, \ldots, a_s\}$ is a set of *size* s and $\cdot$ is an *associative* binary operation on A, which we call *product*. We denote by xy the product of elements $x, y \in A$. A *finite monoid* is a finite semigroup with a distinguished element e, called the *identity*, such that $xe = ex = x$, for all $x \in A$.

For a sequence $\mathbf{x} = (x_0, x_1, \ldots, x_{N-1}) \in A^N$, the sequence of *prefixes* of $\mathbf{x}$ is defined as $\mathbf{y} = (y_0, y_1, \ldots, y_{N-1})$, with $y_j = x_0 x_1 \ldots x_j$. The *prefix problem* consists in computing $\mathbf{y}$ from $\mathbf{x}$.

In the study of the prefix problem, an important role is played by the *Cayley graph* $G(A) = (A,E)$ of A, containing for each ordered pair (x,y) an arc of the form (x,xy), labelled by y.

An element a of A is said to be *recurrent* if $G(A)$ contains a self-loop at a, that is, if there is an element b of a, not necessarily distinct from a, such that $ab = a$.

Two elements $a,b \in A$ are said to be *equivalent*, denoted $a \equiv b$, if there are $c,d \in A$ such that $ac = b$ and $bd = a$, i.e., a and b are joined in a cycle in $G(A)$. Note that "$\equiv$" is an equivalence relation; its equivalence classes are the strongly connected of components of $G(A)$.

A semigroup $<A,\cdot>$ has a *monoidal cycle* if there are five elements, a,b,c,d, and ω in A (not all necessarily distinct but with $a \neq b$) such that $b = ac$, $a = bd$, $a\omega = a$, and $b\omega = b$. In other words, the Cayley graph $G(A)$ has a two-node cycle with nodes a and b that have equally labelled self-loops. A semigroup is referred to as MC or NMC depending upon whether or not it has monoidal cycles, respectively. We shall see that monoidal cycles are crucial in determining the complexity of the prefix problem.

It is well-known (and also easy to prove) that for each element a in a finite semigroup there are two positive integers k and p such that $a, a^2, \ldots, a^{k+p-1}$ are all distinct, and $a^{k+p} = a^k$. Moreover, if the *period* p of a is larger than 1, then $\{a^k, a^{k+1}, \ldots, a^{k+p-1}\}$ form a group. If $p = 1$ for all the elements, then the semigroup is called *group-free* [MP71].

We now give examples of semigroups that belong to the various classes introduced above. If any element $x \in A$ different from the identity has an inverse x^{-1} such that $xx^{-1} = e$, then (e,x,e) forms a monoidal cycle in $G(A)$ and A is MC. As a corollary, all *groups* are MC.

A special but important case of NMC semigroups are the *cycle-free* (CF) semigroups, defined by the absence of cycles in the Cayley graph. We denote a non-cycle-free semigroup as NCF. An example of CF semigroup is the *left-zero* semigroup $<L_p,\circ>$, where $L_p = \{l_1, l_2, \ldots, l_p\}$ and $l_i \circ l_j = l_i$, for all l_i and l_j. An example of semigroup which is NMC and NCF is the *right-zero* semigroup $<R_q,*>$, where $R_q = \{r_1, r_2, \ldots, r_q\}$ and $q \geq 2$, and $r_i * r_j = r_j$, for all r_i and r_j. This semigroup with $q=2$ and with the adjunction of the identity, becomes the monoid that models the function "carry" in binary addition (the identity representing carry propagation and the two zeros representing carry setting and carry resetting, respectively). With the identity, the semigroup becomes MC, although it remains group-free.

Further examples of CF semigroups are all *semilattices*, where the semigroup operation is commutative and idempotent. Semilattices include the set of the 0-1 vectors of length n with respect to component-wise OR (AND), and the set of the first s nonnegative integers with the MINIMUM (MAXIMUM) operation.

3 Lower Bound

We begin by defining the model of computation. The *boolean circuits* considered in this paper are defined in terms of the underlying graphs. Let G be a directed acyclic graph. Inputs are the vertices of indegree 0, and outputs are the vertices of outdegree 0. Each vertex distinct from an input receives a label from the set $\{AND, OR, NOT\}$, and represents a *gate* with the same name. All gates distinct from the outputs have unbounded outdegree (fan-out); gates AND and OR have unbounded indegree (fan-in). Such vertex-labelled graph is a *boolean circuit*. The *size* of the circuit is the number of arcs of its underlying graph; its *depth* is the number of arcs in the longest path from input to output.

We now recall the notion of weak superconcentrator [DDPW83]. Let $i_h, j_h, h = 1, \ldots, k$, be integers in the range $\{1, 2, \ldots, N\}$. An *interleaved matching* of size k on $\{1, 2, \ldots, N\}$ is a set of pairs $\{(i_1, j_1)(i_2, j_2), \ldots, (i_k, j_k)\}$, so that $i_1 < j_1 < i_2 < j_2 < \cdots < i_k < j_k$. Let $\hat{G}$ be a directed acyclic graph with N vertices $\{u_1, \ldots, u_N\}$ of indegree 0 and N vertices $\{v_1, \ldots, v_N\}$ of outdegree 0. We say that $\hat{G}$ is a *weak N-superconcentrator* if, for arbitrary k, $1 \le k \le N$, and any interleaved matching $\{(i_1, j_1), \ldots, (i_k, j_k)\}$ of size k, $\hat{G}$ contains k vertex-disjoint paths joining $\{u_{i_1}, \ldots, u_{i_k}\}$ and $\{v_{j_1}, \ldots, v_{j_k}\}$.

The size-depth trade-off of weak superconcentrators can be expressed in terms of the functions f_i, $i \ge 1$, defined as $f_1(n) = 2^n$, and $f_{i+1}(n) = f_i^{(n)}(2)$, where $f^{(n)}$ denotes the n-fold iterate of f. Each f_i is primitive recursive, monotone, and grows faster that f_{i-1}. Moreover, each primitive recursive function is asymptotically bounded above by a suitable f_i. If f is monotone increasing, $f^{-1}(n)$ denotes the least k such that $f(k) \ge n$.

In [DDPW83] it is shown that the size of a weak N-superconcentrator of depth exactly $2d$ is $\Omega(N f_d^{-1}(N))$. In [CFL83b] it is shown that any prefix-carry circuit of size N contains a weak N-superconcentrator. Below, we show that a prefix circuit for a semigroup with a monoidal cycle can be used to construct a graph with properties similar, although not identical, to those of a weak N-superconcentrator. In order to become applicable to such a graph, the results of [DDPW83] are first reformulated in more general terms.

We say that X is a p-random subset of Y, with $0 \le p \le 1$, if X is formed by taking each element of Y independently with probability p. Let $\hat{G}$ be directed acyclic graph with N vertices $\{u_1, \ldots, u_N\}$ of indegree 0 and N vertices $\{v_1, \ldots, v_N\}$ of outdegree 0. We say that $\hat{G}$ is a *pseudo N-superconcentrator* if, for $2/N \le p \le 1$, the expected number of vertex-disjoint paths between any p-random subset A of $\{u_1, \ldots, u_N\}$ and any p-random subset B of $\{v_1, \ldots, v_N\}$ is $\Omega(pN)$.

It is simple to show that if $\hat{G}$ is a weak N-superconcentrator, then it is also a pseudo N-superconcentrator. Our interest in pseudo N-superconcentrators is due to the following result, whose proof is a minor adaptation of the proof of Theorem 2 in [DDPW83] and is therefore omitted here.

Lemma 1. The size of a pseudo N-superconcentrator of depth exactly $2d$ is $\Omega(N f_d^{-1}(N))$.

We are now ready to state the main result of this section.

Theorem 1. If $<A, \cdot>$ is an MC semigroup, then any length-N prefix circuit for $<A, \cdot>$ of depth d has size $\Omega(N f_{d+1}^{-1}(N))$.

Proof. We shall construct a pseudo $(N-1)$-superconcentrator C by modifications of the prefix circuit for $<A, \cdot>$, which increase the depth by at most one and the size by $O(N)$. Then, the result follows from Lemma 1.

Throughout this proof, a and b form a monoidal cycle such that $a\omega = a$, $b\omega = b$, $ac = b$, and $bd = a$.

In the prefix circuit constructed with boolean gates, values of input and output semigroup variables are encoded in binary. Specifically, we let x_{ij}, $j = 1,2,\ldots,r(i)$, be the binary variables encoding $x_i \in A$ and y_{ij}, $j = 1,2,\ldots,s(i)$, the binary variables encoding $y_i \in A$. Note that we allow different encodings to be used for different I/O variables. We let $x_{ij}(z)$ be the value of x_{ij} in the encoding of semigroup element z. Likewise we define $y_{ij}(z)$.

We now construct a circuit C by adding to the prefix circuit $(N-1)$ new binary terminals $u_1, \ldots, u_{N-1}$ and by connecting u_i to a subset of the x_{ij}'s according to the following scheme. For odd i, if $x_{ij}(\omega) = 0$ and $x_{ij}(c) = 1$, then connect u_i to x_{ij}; if $x_{ij}(\omega) = 1$ and $x_{ij}(c) = 0$, then connect $\bar{u}_i$ (obtained at the output of an inverter with input u_i) to

x_{ij}; otherwise (i.e., when $x_{ij}(\omega) = x_{ij}(c)$) no connection is created. For even i, a similar scheme is used, with c replaced by d. Observe that, if the x_{ij}'s that are not connected to u_i are kept set to the value $x_{ij}(\omega)$, then $u_i = 0$ implies $x_i = \omega$ for every i, and $u_i = 1$ implies $x_i = c$, for odd i, and $x_i = d$, for even i.

Since $a \neq b$, there exists a smallest j such that $y_{ij}(a) \neq y_{ij}(b)$, and we choose v_i as the terminal that outputs y_{ij} in C, for $i = 1,...,N-1$. We shall prove that C is a pseudo $(N-1)$-superconcentrator with respect to the sets of vertices $U = \{u_1, \ldots, u_{N-1}\}$ and $V = \{v_1, \ldots, v_{N-1}\}$.

Next, we define the convenient notion of witness. Let $I = (i_1, \ldots, i_{2q})$ and $J = (j_1, \ldots, j_q)$ be two increasing sequences of integers between 1 and $N-1$ such that, for $h = 1,...,q$, i_{2h-1} is odd, i_{2h} is even, and $i_{2h-1} \leq j_h < i_{2h}$. Then, the pair (A,B), with $A \triangleq \{u_i: i \in I\}$ and $B \triangleq \{v_j: j \in J\}$, is called a *witness* of size q.

The interesting property of a witness of size q is that there exist q vertex-disjoint paths between vertices of A and vertices of B. To establish this fact, let us consider the set of 2^q input assignments for the u_i's satisfying the conditions $u_{i_{2h-1}} = u_{i_{2h}}$ and $u_i = 0$ for $i \notin I$. It is easy to see that, if x_0 is set to a and the x_{ij}'s not connected to x_i are set to $x_{ij}(\omega)$, then the assignments of the u_i's defined above correspond to instances of the prefix problem such that $x_0 = a$, $x_i = \omega$ for $i \geq 1$ not in I, $x_{i_{2h-1}} = x_{i_{2h}} = \omega$ for $u_i = 0$, and $x_{i_{2h-1}} = c$, $x_{i_{2h}} = d$ for $u_i = 1$. Correspondingly, $y_{j_h} = a$ whenever $x_{i_{2h-1}} = \omega$ and $y_{j_h} = b$ whenever $x_{i_{2h-1}} = c$. Since the binary variable on terminal v_i has different values for $y_i = a$ and $y_i = b$, the variables in B take on 2^q different configurations.

Summarizing, there are 2^q configurations of the variables in A generating 2^q different configurations of the variables in B. Then, each set of vertices of C whose removal disconnects A and B must have cardinality at least q, since the values of these vertices uniquely determine the values of those in B. We conclude from Menger's theorem that there are q vertex-disjoint paths from A to B. (The argument in this paragraph is patterned after a similar one in [CFL83b].)

To conclude the proof of the theorem, we need to show that if U' is a p-random subset of U, and V' is a p-random subset of V, then there is a witness (A,B), with $A \subseteq U'$ and $B \subseteq V'$, of expected size $\Omega(pN)$. Indeed, we shall extract the witness from U' and V' by scanning the indices from 1 to $N-1$ in increasing order and successively choosing triples of the form (i,j,i') with $i \in U'$ odd, $i' \in U'$ even, $j \in V'$, and $i \leq j < i'$. The expected number of indices to be scanned in order to form a triple (i,j,i') is computed as follows. Let j^* denote the smallest odd index not smaller than j. Starting from an odd h, the expected lengths of the intervals $[h,i]$, $[i+1,j^*]$, and $[j^*+1,i']$ are easily found to be $(2/p-1)$, $(1/p-1+(1-p)/(2-p))$, and $(2/p-1)$, respectively, so that the expected value of $(i'-i+1)$ is $5/p-3+(1-p)/(2-p)<5/p$. Hence, in a sequence of $N-1$ indices the expected number of triples is at least $p(N-1)/5=\Omega(pN)$. From these triples, the desired [illegible] is readily constructed. □

Theorem 1 shows that the prefix circuits proposed in [CFL83a] for group-free semigroups, which have size $O(N f_d^{-1}(N))$ and depth $O(d)$, have optimal size to within a constant factor for the depth, when the group-free semigroup is in particular an MC semigroup.

4 Upper Bound

In this section we give a constructive proof that if $<A,\cdot>$ is an NMC-semigroup, then there is a constant-depth prefix circuit for $<A,\cdot>$ of linear size. We begin by deriving a crucial property of $G(A)$ as a direct consequence of the absence of monoidal cycles.

Lemma 2. Let x be a recurrent element of A, and, for some u and v in A, let $ux=a$ and $vx=b$, with $a\equiv b$. If $<A,\cdot>$ is an NMC-semigroup, then $a=b$.

Proof. Assume, for a contradiction, that $a\neq b$. For any $c\in A$, let $SL(c)\triangleq\{\omega|\ c\omega=c\}$, i.e., the set of labels of the self-loops of c in $G(A)$. Since x is recurrent, then $|SL(x)|\geq 1$. We claim that $SL(x)\subseteq SL(a)$. Indeed, for any $\omega\in SL(x)$, $x\omega=x$; using the hypothesis $ux=a$, we have $a\omega=(ux)\omega=u(x\omega)=ux=a$, i.e., $a\omega=a$, as claimed. Analogously, we show $SL(x)\subseteq SL(b)$. But the absence of monoidal cycles means that for two equivalent elements a and b we have $SL(a)\cap SL(b)=\emptyset$. This implies $SL(x)=\emptyset$, contrary to the hypothesis that x is recurrent. □

Let m' be the number of nodes on the longest path of $G(A)$ consisting exclusively of nonrecurrent elements of A. (Note that $m'=0$ if every $a\in A$ is recurrent.) We now establish the following important property of NMC-semigroups:

Theorem 2. Let $<A,\cdot>$ be an NMC-semigroup and a,b,c, and d be equivalent elements of A. Let $av_1v_2\cdots v_{m'+1}=c$ and $bv_1v_2\cdots v_{m'+1}=d$, where $v_i\in A$, $1\leq i\leq m'+1$. Then $c=d$.

Proof. Since $c\equiv a$ by hypothesis, then $av_1v_2\cdots v_j\equiv a$ for all $j\in[1,m'+1]$, by the definition of strongly connected component of $G(A)$; similarly, we establish $bv_1v_2\cdots v_j\equiv b$. Moreover, according to the above definition of m', the sequence of the prefixes of the sequence $(v_1, v_2, \ldots, v_{m'+1})$ contains at least one recurrent element. Let $w_k = v_1 v_2 \cdots v_k$ $(k \le m'+1)$ be one such prefix. Since w_k is recurrent and $aw_k \equiv a \equiv b \equiv bw_k$, by Lemma 2 we have $aw_k = bw_k$. Then it trivially follows that $c = av_1 \cdots v_{m'+1} = aw_k v_{k+1} \cdots v_{m'+1} = bw_k v_{k+1} \cdots v_{m'+1} = bv_1 \cdots v_{m'+1} = d$. □

This theorem shows that the mere knowledge that ax and a are in the same strongly connected component of $G(A)$ for some recurrent x dispenses us from knowing exactly the value of a if we wish to know ax.

Before proceeding to the construction of the prefix circuit, we introduce an important structural parameter of $G(A)$, denoted h_A, and defined as the maximum number of distinct strongly connected components visited by any (directed) path in $G(A)$.

The prefix circuit is best described as an $h_A \times N$ array. The rows of this array are referred to as "stages" numbered from 1 to h_A. Each stage contains N modules, numbered from 0 to $N-1$, responsible for "local" processing, and circuits spanning the entire stage, responsible for global processing. Module $M_j^{(h)}$ appears in the $(j+1)$-st column of the h-th stage.

Any given instance of prefix computation corresponds to the segmentation of the interval $[0, N-1]$ into a collection of contiguous intervals $\{[j_{i-1}, j_i-1] \mid i=1,\ldots,h'\}$, where $0 = j_0 < j_1 < \cdots < j_{h'} = N$, such that $y_{j_{i-1}} \equiv \cdots \equiv y_{j_i-1}$ for $i=1,\ldots,h'$ and $y_{j_i-1} \not\equiv y_{j_i}$, for $i=1,\ldots,h'-1$. In other words, all prefixes with indices in a given interval are in the same strongly connected component and $h' \le h_A$. Referring to Figure 1, the indices of the N modules in the generic stage h are partitioned into three contiguous intervals: the interval $[0, j_{h-1}]$ of the *passive* modules, the interval $[j_{h-1}+1, j_h]$ of the *productive* modules, and the interval $[j_h+1, N-1]$ of the *irrelevant modules*. The reasons for these denotations are: A passive module $M_j^{(h)}$ simply passes to $M_j^{(h+1)}$ the (previously correctly computed) prefix y_j it receives from $M_j^{(h-1)}$; a productive module $M_j^{(h)}$ correctly computes prefix y_j and forwards it to $M_j^{(h+1)}$; an irrelevant module $M_j^{(h)}$ computes a possibly incorrect prefix, not to be forwarded to $M_j^{(h+1)}$.

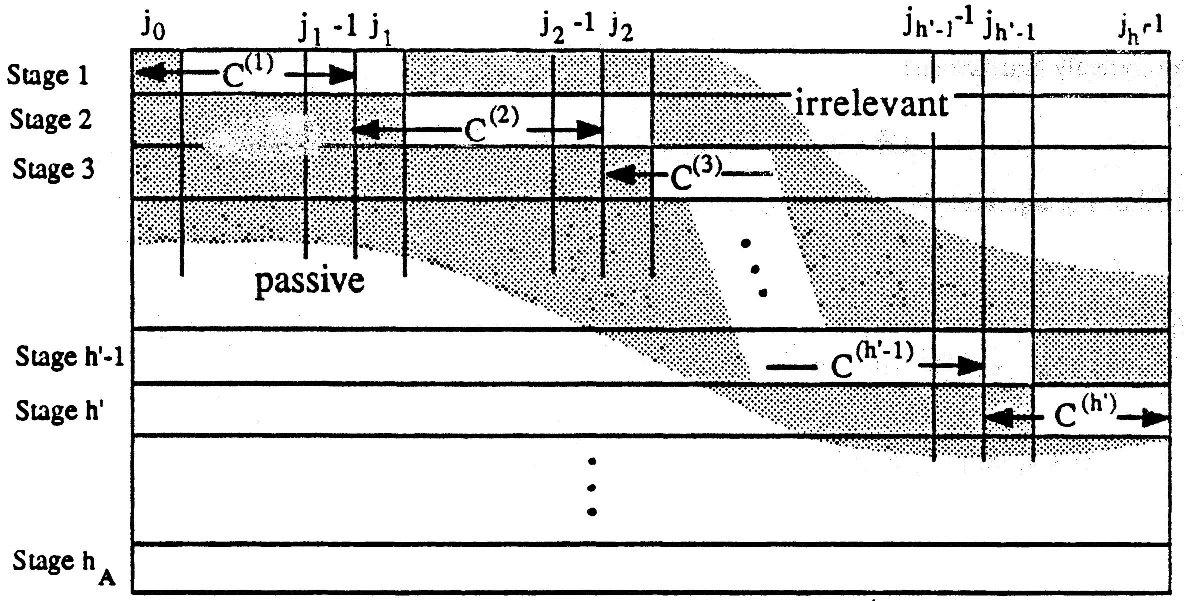

Fig. 1.
Schematic layout of the linear-size prefix circuits for NMC semigroups. Each stage is segmented into three portions: the passive modules (left), the productive modules (center), and the irrelevant modules (right).

We now describe in detail the operation of the generic stage h, where $1 \le h \le h_A$. Notationally, $u = (u_0, u_1, \ldots, u_{N-1})$ is an N-component vector, where u_j, semigroup set, or $B = \{0,1\}$). Stage h receives three vectors from Stage $h-1$:

(i) the *enable* vector $e^{(h-1)} = (e_0^{(h-1)}, \ldots, e_{N-1}^{(h-1)})$, where $e_j^{(h-1)}, 0 \le j \le N-1$, is boolean. This vector is used to separate the passive modules from the productive and the irrelevant ones, i.e., $e_0^{(h-1)} = e_1^{(h-1)} = \cdots = e_{j_{h-1}}^{(h-1)} = 0$ and $e_{j_{h-1}}^{(h-1)}+1 = \cdots = e_{N-1}^{(h-1)} = 1$.

(ii) the *prefix* vector $y^{(h-1)} = (y_0^{(h-1)}, \ldots, y_{N-1}^{(h-1)})$, where $y_j^{(h-1)}$, $0 \le j \le N-1$, is an element of A. Of these components, $(y_0^{(h-1)}, \ldots, y_{j_{h-1}}^{(h-1)})$ represent correct values – i.e., $y_j^{(h-1)} = y_j$ for $j = 0, 1, \ldots, j_{h-1}$ – while $y_j^{(h-1)} = y_{j_{h-1}}$ for $j_{h-1} < j \le N-1$. (Note that $y_{j_{h-1}} \ne y_{j_{h-1}-1}$.)

(iii) the *infix* vector $w = (w_0, \ldots, w_{N-1})$, where $w_j \in A$, $0 \le j \le N-1$. Letting $m \triangleq m'+2$, the components of w are defined as follows

$$w_j = \begin{cases} x_0 x_1 \cdots x_j & \text{for } j < m, \\ x_{j-m+1} \cdots x_j & \text{for } j \ge m. \end{cases}$$

Notice that w does not depend on the index h, and is computed by a linear-size constant-depth network. Vectors $e^{(0)}$ and $y^{(0)}$ are correctly initialized as

$$e^{(0)} = (0,1,1,\ldots,1) \text{ and } y^{(0)} = (x_0,x_0,\ldots,x_0). \tag{1}$$

The following algorithm illustrates the operation of Stage h:

1. $z_j^{(h)} = \begin{cases} w_j & \text{for } j=0,\ 1,\ldots,\ m-1 \\ y_{j-m}^{(h-1)}\ w_j & \text{for } j=m,\ m+1,\ldots,\ N-1 \end{cases}$

2. $u_{j+1}^{(h)} = \begin{cases} 0 \text{ if } z_j^{(h)} = y_j^{(h-1)} \text{ or } e_j^{(h-1)} = 0 \\ 1 \text{ if } z_j^{(h)} \neq y_j^{(h-1)} \text{ and } e_j^{(h-1)} \qquad j=0,1,\ldots,N-2 \end{cases}$

3. $e^{(h)} = PREFIX-OR\ (u^{(h)})$

4. $t_j^{(h)} = \neg\, e_j^{(h)} \wedge e_{j+1}^{(h)}$

5. $Y^{(h)} = z_{j}^{(h)}$ where $t_{j}^{(h)} = 1$

6. $y_j^{(h)} = \begin{cases} y_j^{(h-1)} \text{ if } e_j^{(h)} = 0 \\ Y^{(h)} \text{ if } e_j^{(h)} = 1 \end{cases}$

From the performance viewpoint, we note that each step is executed by a circuit with size $O(N)$ in constant depth. This is trivial for Steps 1,2,4,5, and 6; as for Step 3, this result is achieved by the prefix-OR circuit of [CFL83b]. Thus, the boolean circuit implementing the above computation has linear size and constant depth.

We now turn our attention to the correctness of the above procedure. We assume inductively that $e^{(h-1)}$ and $y^{(h-1)}$ are correct. The basis for the induction is provided by (1). Since, by Step 3, $e^{(h)}$ is the PREFIX-OR of vector $u^{(h)}$, we wish to show that $u_j^{(h)} = 0$ for $j \le j_h$ and $u_{j_h+1}^{(h)} = 1$. Indeed, for $j \le j_{h-1}$ we have $u_j^{(h)}=0$, since $e_j^{(h-1)}=0$ by the inductive hypothesis. For $j=j_{h-1}+1,\ldots,j_h$, we argue as follows. Let $C^{(h-1)}$ and $C^{(h)}$ be the strongly connected components of the correct prefixes computed by stages $(h-1)$ and h respectively. Depending upon whether $j_h \le j_{h-1} + m$ or not, we distinguish two cases:

1. $j \le j_h \le j_{h-1}+m$. In this case, since $y_j^{(h-1)}=y_j$ for $j \le j_{h-1}$ (with $y_{j_{h-1}-1} \in C^{(h-1)}$ and $y_{j_{h-1}} \in C^{(h)}$), by virtue of Step 1 above, $z_j^{(h)} = y_j$ for $j \le j_{h-1}+m$. Thus the smallest value $i \in [j_{h-1}, j_{h-1}+m]$ for which $z_i^{(h)} \notin C^{(h)}$ is detected by

the condition $(e_i^{(h-1)}=1) \wedge (z_i^{(h)} \neq y_i^{(h-1)})$ and $u_{i+1}^{(h)}=1$ is correctly generated. Note that $u_j^{(h)}=0$ for $j \le i$.

2. $j_h \ge j > j_{h-1}+m$. In this case, the precise value of y_{j-m+1} is not known, but we know that $y_{j-m+1} \in C^{(h)}$. Since w_j is the product of m semigroup elements, one of its prefixes is a recurrent element x of A. Then $y_{j-m+1} \cdot x$ is, by Theorem 2, a unique element of $C^{(h)}$, independent of the specific $y_{j-m+1} \in C^{(h)}$. From this point on, we can argue as in Case 1 above.

This analysis shows that Step 2 always generates $u_{j_h+1}^{(h)}=1$ while $u_k^{(h)}=0$ for $k \le j_h$. Therefore Step 3 correctly generates a vector $e^{(h)}$ such that $e_i^{(h)}=0$ for $i \le j_h$ and $e_i^{(h)}=1$ for $i > j_h$. Subsequently, Step 4 uniquely detects position j_h (since $e_{j_h}^{(h)} e_{j_h+1}^{(h)} = 01$), and Steps 5 and 6 construct vector $y^{(h)}$ in a straightforward manner. This completes the correctness proof and we have:

Theorem 3. If $\langle A, \cdot \rangle$ is an NMC-semigroups, then there exists a length-N bounded-depth prefix circuits for $\langle A, \cdot \rangle$ of size $O(N)$.

5 Remarks and Conclusions

The results of [CFL83a, CFL83b] and of this paper show that the optimal size of constant-depth prefix circuits is exponential for non-group-free semigroups, slightly superlinear for group-free MC semigroups, and linear for (group-free) NMC semigroups.

It is interesting to contrast these results with those on the size-time complexity of prefix boolean networks [BP87]. In this context, semigroups can be divided into two classes: *friction* semigroups, for which the size-time product is superlinear ($ST = \Theta(N \log (N/T))$), and *frictionless* semigroups, for which the size-time product is linear ($ST = \Theta(N)$). The class of friction semigroups can be more precisely defined as the intersection of two other classes: The NCF semigroups and the memory-inducive semigroups. The latter are those in which products of arbitrary length are true functions of all their factors, and can be structurally characterized as those semigroups whose recurrent subsemigroup is not isomorphic to a direct product of a left-zero and a right-zero semigroup. The inclusion relationship among the relevant classes of semigroups mentioned above is illustrated by means of a Venn diagram in Figure 2.

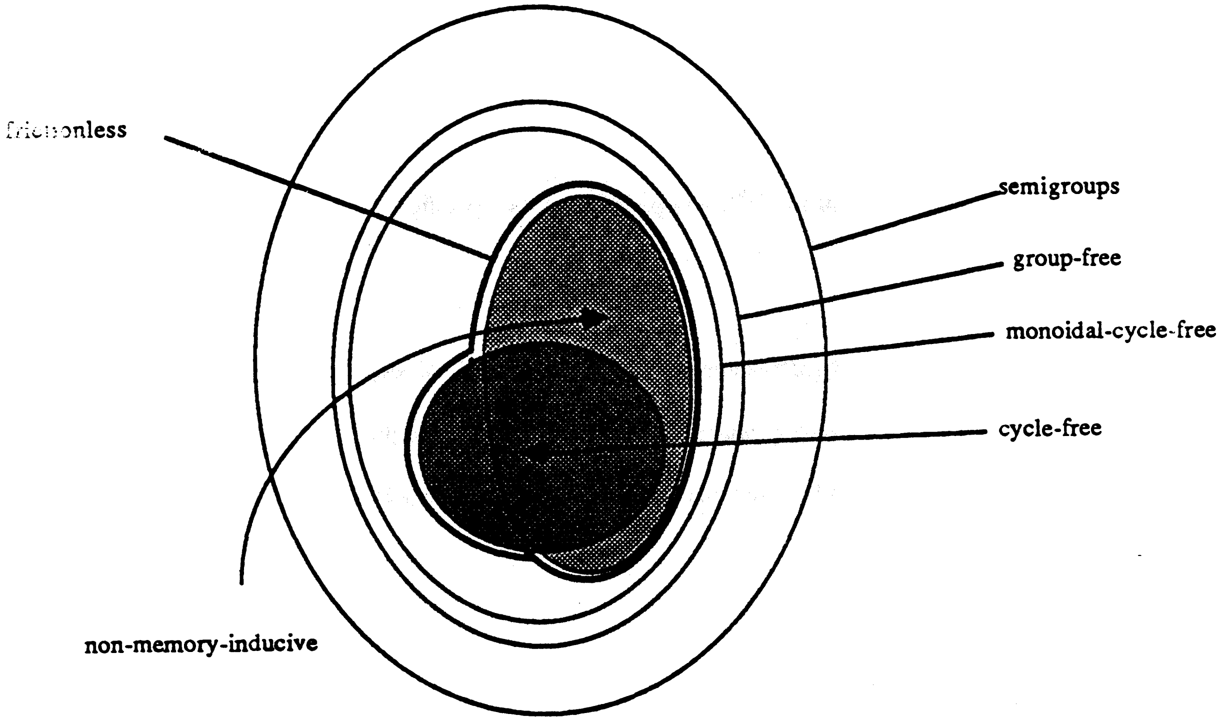

Fig. 2.
Inclusion relationships among classes of semigroups with respect to the complexity of their prefix circuits.

We conclude by mentioning an open problem. We can show that cycle-free and right-zero semigroups are NMC. We can also show that the class of NMC semigroups is closed under direct product and under homomorphism. It would be interesting to ascertain whether NMC semigroups can be characterized as the homomorphic images of direct products of particular types of semigroups.

Acknowledgement

We are indebted to N. Pippenger for bringing to our attention the problem solved in this paper, and for several penetrating suggestions.

References

[BK82] R.P. Brent and H.T. Kung, "A regular layout for parallel adders," *IEEE Transactions on Computers*, vol. C-31, n. 3, pp. 260-264, March 1982.

[BP85] G. Bilardi and F.P. Preparata, "The influence of key length on the area-time complexity of sorting," *Proceedings I.C.A.L.P., (Ed. W. Brauer)* Nafplion, Greece, Springer-Verlag, pp. 53-62, July 1985.

[BP86] G. Bilardi and F.P. Preparata, "Digital filtering in VLSI," *Proceedings of Aegean Workshop on Computing* (Eds.F. Makedon et al.) Loutraki, Greece, Springer-Verlag, pp. 1-11, July 1986.

[BP87] G. Bilardi and F.P. Preparata, "Size-time complexity of boolean networks for prefix computations," *Proceedings of the 19th ACM Annual Symposium on Theory of Computing*, New York, NY, pp. 436-442, May 1987.

[CFL83a] A.K. Chandra, S. Fortune, R. Lipton, "Unbounded fan-in circuits and associative functions," *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, Boston, MA, pp. 52-60, April 1983.

[CFL83b] A.K. Chandra, S. Fortune, R. Lipton, "Lower bounds for constant depth circuits for prefix problems" *Proceedings I.C.A.L.P.*, 1983.

[CP61] A. H. Clifford and G.B. Preston, *The algebraic theory of semigroups.* Vol. 1. American Mathematical Society, Providence RI, 1961.

[CS85] R. Cole and A. Siegel, "On information flow and sorting: New upper and lower bounds for VLSI circuits," *Proceedings 26th Annual Symposium on the Foundations of Computer Science*, Portland, OR, pp. 208-221, October 1985.

[CV86] R.Cole and U. Vishkin, "Approximate and exact parallel scheduling with applications to list, tree and graph problems," *Proceedings 27th Annual Symposium on the Foundations of Computer Science*, Toronto, Canada, pp. 478-491, October 1986.

[DS83] E. Dekel and S. Sahni, "Binary trees and parallel scheduling algorithms," *IEEE Transactions on Computers*, vol. C-32, n. 3, pp. 307-315, March 1982.

[DDPW83] D. Dolev, C. Dwork, N. Pippenger, A. Widgerson, Superconcentrators, generalizers, and generalized connectors with limited depth," *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, Boston, MA, pp. 42-51, April 1983.

[F83] F.E. Fitch, "New bounds for parallel prefix circuits," *Proceedings 15th Annual ACM Symposium on Theory of Computing, Boston*, MA, pp. 100-109, April 1983.

[FSS81] M. Furst, J. Saxe, and M. Sipser, "Parity, circuits, and the polynomial time hierarchy," *Proceedings of 22nd Annual IEEE Symposium on Foundations of Computer Science*, pp.260-270, 1981.

[H86] J. Hastad, "Almost optimal lower bounds for small depth circuits," Proceedings 15th Annual ACM Symposium on Theory of Computing, Berkeley, CA, pp. 6-20, May 1986.

[K78] D.J. Kuck, *The structure of computers and computations*, New York: Wiley, 1978.

[KRS85] C.P. Kruskal, L. Rudolph, M. Snir, "The power of parallel prefix," *IEEE Transactions on Computers*, vol. C-34, n. 10, pp. 965-968, October 1985.

[LF80] R.E. Ladner and M.J. Fischer, "Parallel prefix computation," *Journal of the ACM*, vol. 27, n. 4, pp. 831-838, October 1980.

[MP71] R. McNaughton and S. Papert, *Counter-free automata.* M.I.T. Press, Cambridge, MA, 1971.

[O63] Yu. Ofman, "On the algorithmic complexity of discrete functions," *Soviet Physics-Doklady*, vol. 7, no. 7, pp. 589-591, January 1963.

[P87] N. Pippenger, "The complexity of computations by networks", *IBM Journal of Research and Development*, vol. 23, no. 2, pp. 235-243, March 1987.

[R84] J. Reif, "Probabilistic parallel prefix computation," *Proceedings of 1984 International Conference on Parallel Processing'* pp. 291-298, August 1984.

[S86] M. Snir, "Depth-size trade-offs for parallel prefix computation," *Journal of Algorithms*, vol. 7, no. 2, pp. 185-201, June 1986.

[Y85] A. C. Yao, "Separating the polynomial-time hierarchy by oracles," *Proceedings 26th Annual IEEE Symposium on Foundation of Computer Science*, Portland, Oregon, pp.1-10, May 1985.