Probabilistic Parallel Prefix Computation

J. H. REIF
Department of Computer Science, Duke University
Durham, NC 27706, U.S.A.

Abstract—Given inputs $x_1, \ldots, x_n$, which are independent identically distributed random variables over a domain D, and an associative operation $\circ$, the *probabilistic prefix computation problem* is to compute the product $x_1 \circ x_2 \circ \cdots \circ x_n$ and its $n-1$ prefixes. Instances of this problem are finite state transductions on random inputs, the addition or subtraction of two random n-bit binary numbers, and the multiplication or division of a random n-bit binary number by a constant.

The best known constant fan-in circuits for these arithmetic operations had logarithmic depth, linear size, and produce no errors. Furthermore, matching lower bounds for depth and size (up to constant factors between the upper and lower bounds) had previously been obtained for the case of constant fan-in circuits with no errors.

We give arithmetic circuits for probabilistic prefix computation, which for these random arithmetic operations have constant fan-in, linear size, $O(\log \log n)$ depth, but error probability less than $n^{-\alpha}$ for any given $\alpha > 0$. For any constant fan-in circuits computing these random arithmetic operations with error probability $n^{-\alpha}$, we prove the circuit depth must be bounded from below by $\Omega(\log \log n)$. Hence, we conclude our circuits have asymptotically optimal depth among circuits with error probability $n^{-\alpha}$.

We also give error-free circuits for these random arithmetic operations with constant fan-in at all nodes but one, linear size, and $O(\log \log n)$ expected delay for their parallel evaluation.

1. INTRODUCTION

1.1. Previously-Known Circuits for Prefix Computation and Arithmetic

Given a function which can be computed sequentially with finite memory, we wish to design a circuit for its parallel computation. This practical problem is reduced, in [1], to the prefix computation problem: given n inputs $x_1, \ldots, x_n$ taken from a domain D, compute all the products $x_1 \circ x_2 \circ \cdots \circ x_i$ for $1 \leq i \leq n$, where $\circ$ is an associative operation. Ladner and Fischer [1] also give circuits for prefix computation with linear size and logarithmic depth (where the *depth* is the length of the longest path in the circuit). For certain parameters, their circuits are identical to well-known circuits used for addition. Recently, Fich [2] has improved on the size of these circuits by constant factors. No previous work on prefix computation has considered a random distribution of inputs.

It has long been known [3,4] that the expected length of the longest carry during the addition of uniformly distributed random n-bit binary numbers is $O(\log n)$. Some early addition circuits of Gilchrist, Pomerene and Wong [5] and Hendrickson [6] employed carry-completion testing. However, these circuits use only a sequential method for propagating carries, so their depth is $\Omega(\log n)$. On the other hand, Kilburn, Edwards and Aspinall [7,8] devised a parallel method for propagating carries, commonly known as the carry-look-ahead method (see [9]), but do no

This work has been supported in part by DARPA/ARO Contract DAAL03-88-K-0195, Air Force Contract AFOSR-87-0386, DARPA/ISTO Contract N00014-88-K-0458 and N00014-91-J-1985, NASA Subcontract 550-63 of Prime-contract NAS5-30428. Approved for public release: distribution is unlimited.

Preprint of paper appearing in Computers and Mathematics with Applications, Vol. 26, Number 1, July 1993, pp. 101-110.

carry-completion testing. The best known constant fan-in circuit for addition [10] employs a complicated variant of the carry-look-ahead method, has linear size and $\Omega(\log n)$ depth with no improvement for random inputs. In fact, for any of the above arithmetic operations over random input, the best known constant fan-in circuits have $\Omega(\log n)$ depth. Recently, Chandra, Fortune and Lipton [11] gave an addition circuit with near linear size and constant depth, but with $\Omega(n)$ nodes of unbounded fan-in, so they were not practical for VLSI applications.

1.2. Our Circuits for Probabilistic Prefix Computation and Arithmetic

The goal of this paper is to develop some fundamental techniques for the design of circuits which take random input.

In Section 2 of this paper, we formulate a probabilistic version of the prefix computation problem with random input. This probabilistic prefix computation problem has important practical applications (see Section 3) to arithmetic operations on uniformly distributed random numbers, such as

(i) addition or subtraction of two random n-bit binary numbers;
(ii) multiplication or division of a random n-bit binary number by a constant.

In Section 2, we describe circuits for probabilistic prefix computation. Our circuits have constant fan-in, linear size, and generally far less than logarithmic depth. These probabilistic circuits are defined using a parameter, the dependence length, which is related to the number of compositions of random input symbols required to compute an output within given likelihood. The depth of our circuits are the logarithm of the dependence length. The dependence length is $O(\log n)$ (NOTE: Throughout this paper, logarithms with no base indicated will be taken base 2.) for the above arithmetic operations over random inputs. Hence, our circuits have $O(\log \log n)$ depth and furthermore, have error probability which can be set to $n^{-\alpha}$ for any constant $\alpha > 0$. In applications with truly random inputs, this error probability might be set lower than the circuit component reliability.

In applications where we are not assured of random inputs, these probabilistic circuits may not be appropriate, since they do allow errors on certain fixed inputs. Instead, they can be used as the fundamental building blocks for variable delay circuits which make no errors. We modify our probabilistic circuits to detect all errors by introducing a single node of unbounded fan-in and a secondary error-free prefix circuit of logarithmic depth which is evaluated only on detection of an error in the primary circuits' computation. The *delay* of the resulting error-free circuit is the number of parallel stops required for its evaluation. For the above arithmetic operations (i) and (ii) over random inputs, our error-free circuits have at most $O(\log \log n)$ delay with probability at least $1 - n^{-\alpha}$, although they have $O(\log n)$ delay in the worst case. Our variable delay circuits for arithmetic may be of practical use in VLSI applications, since a single node of unbounded fan-in can easily be implemented in current technology by a single dedicated layer in the VLSI chip.

1.3. Lower Bounds for Arithmetic

Winograd [12] showed that $\Omega(\log_f n)$ is a lower bound on the depth to compute addition of n-bit binary numbers by a circuit of fan-in f. In Section 4, we prove that for the above arithmetic operations on random n-bit binary numbers, any circuit of fan-in f, with error probability at most $n^{-\alpha}$, must have $\Omega(\log_f \log n)$ expected delay. Thus, our constant fan-in circuits for these random arithmetic operations are asymptotically optimal. There were no previously known circuit depth lower bounds for any random-input problems.

2. DEFINITIONS AND CIRCUIT CONSTRUCTIONS

2.1. Definition of Probabilistic Prefix Computations

Fix a semigroup $\langle D, \circ \rangle$ (i.e., the product operation $\circ$ is closed over D and is associatve). Let $d : D \to [0,1]$ be a density funtion over D, i.e., satisfying $\sum_{a \in D} d(a) = 1$. A *probabilistic semigroup* is specified by $\langle D, \circ, d \rangle$. Let $x_1, \ldots, x_n$ be independent random variables, each with distribution function d (so each x_i is randomly chosen to be some domain element $a \in D$ with probability $d(a)$, independently of the choice of any other x_j, $j \neq i$). The *probabilistic prefix computation problem* is to compute the n outputs:

$$x_1, x_1 \circ x_2, \ldots, x_1 \circ x_2 \circ \cdots \circ x_n,$$

given n random inputs $x_1, \ldots, x_n \in D$.

Define $x \in D$ to be a *prefix invariant*, if for all $a \in D$, $a \circ x = x$. Obviously, if x is prefix invariant, then so is $x \circ y$ for any $y \in D$. For any ε, $0 \leq \varepsilon \leq 1$, let the *$\varepsilon$-dependence length* $\ell(\varepsilon)$ be the minimum integer $\ell \geq 1$ such that

$$\text{Prob}\,(x_1 \circ \cdots \circ x_\ell \text{ is prefix invariant}) \geq 1 - \varepsilon,$$

for independent random variables $x_1, \ldots, x_\ell$ with density function d. Intuitively, the ε-dependence length gives the minimal number of variables which must be composed together before the resulting product is prefix invariant with probability at least $1 - \varepsilon$. Thus, if we are attempting to compute $x_1 \circ \cdots \circ x_i$, we need only compute $x_{i-\ell(\varepsilon)+1} \circ \cdots \circ x_i$ and with probability $1 - \varepsilon$, we need not perform the further $i - \ell(\varepsilon)$ compositions. For the arithmetic operations over random input considered in Section 3, we show the ε-dependence length is $\ell(\varepsilon) = O(\log n)$ for $\varepsilon = n^{-\alpha}$.

We will derive an upper bound on the ε-dependence length as a function of

$$p = \sum_{a \in D \text{ is prefix invariant}} d(a).$$

Since the x_i are independent,

$$\begin{aligned}(1-p)^{\ell(\varepsilon)-1} &\geq \text{prob}\,(x_1 \circ \cdots \circ x_{\ell(\varepsilon)-1} \text{ is not prefix invariant}) \\ &> \varepsilon \text{ by definition of minimality of } \ell(\varepsilon).\end{aligned}$$

Hence, we have the following proposition.

PROPOSITION 1. $\ell(\varepsilon) \leq \lceil \log(\varepsilon)/\log(1-p) \rceil$.

2.2. Circuit Definitions

A *circuit* $\mathfrak{C}$ is a labelled acyclic direction-oriented graph. The *in-degree* (*out-degree*) of a node is the number of entering (departing, respectively) edges. The *input* nodes are those of in-degree O. The *output* nodes are a distinguished set of nodes. Let n, n' be the number of input and output nodes, respectively. The *fan-in* of $\mathfrak{C}$ is the maximum in-degree of any node. The *size* of $\mathfrak{C}$ is the number of edges. (NOTE: Ladner and Fischer [1] define the size to be the number of non-input nodes, but this is not appropriate for circuits of unbounded fan-in.) The *depth* of $\mathfrak{C}$ is the length of the longest path. (See Section 2.4 for the definition of delay.)

Associated with the circuit $\mathfrak{C}$ is a set D and a set of mappings B from tuples of D to D. B is called the *basis* of $\mathfrak{C}$. Each of the input nodes is labelled by an element in D. The label of each non-input node v is a mapping $D^i \to D$ in basis B, where i is the in-degree of v. A parallel evaluation of the circuit $\mathfrak{C}$ can be done in at most depth($\mathfrak{C}$) parallel steps. The circuit $\mathfrak{C}$ thus yields a mapping $D^n \to D^{n'}$. For example, a *Boolean* circuit has $D = \{0,1\}$ and basis

$B = \{\vee, \wedge, \neg\}$. A Boolean circuit for addition of two $(n/2)$-bit numbers will have n input nodes, each of which is labelled by a distinct bit of the input numbers, and will have $n' = 1 + n/2$ output nodes, each of which is labelled by a disticnt bit of the output.

A *product* circuit has basis $\{\circ\}$, where $\langle D, \circ\rangle$ is a semigroup. Ladner and Fischer [1] show the following proposition.

PROPOSITION 2. *For $0 \leq k \leq \log n$, there is a product circuit $\mathfrak{P}_k(n)$ for prefix computation with fan-in 2, size $4(1 + 2^{-k})\, n - o(n)$ and depth $k + \lceil \log n \rceil$, which makes no output errors.*

So as to make our paper self-contained, we give the Ladner and Fischer proof in circuit in Figures 1a and 1b. Again, note that we consider the size to be the number of edges. Fich [2] gives a constant factor improvement in the size of such circuits.

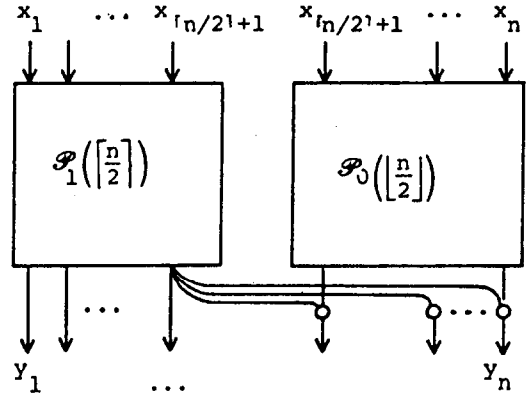

Figure 1a. Ladner and Fischer's construction of $\mathfrak{P}_0(n)$.

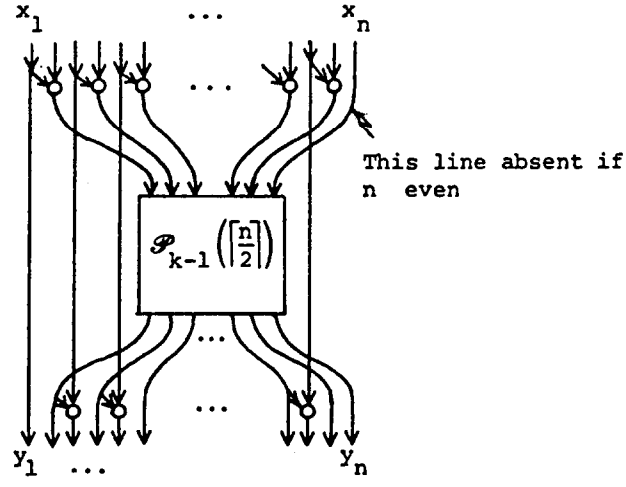

Figure 1b. Ladner and Fischer's construction of $\mathfrak{P}_k(n)$, $k \geq 1$.

2.3. Our Product Circuits for Probabilistic Prefix Computation

Let $\langle D, \circ, d\rangle$ be a probabilistic semigroup as defined in Section 2.1.

Let $\ell(\varepsilon)$ be the ε-dependence length for $\langle D, \circ, d\rangle$. A key lemma of this paper follows.

LEMMA 1. *For any $n \geq 1$, $0 \leq k \leq \log n$ and $0 < \varepsilon < 1/n$, there is a product circuit for probabilistic prefix computation with fan-in 2, size $(6 + 2^{2-k})\, n - 2\ell(\varepsilon) - o(n)$, depth $k + \lceil \log \ell(\varepsilon) \rceil + 1$ and error probability at most $n\varepsilon/\ell(\varepsilon)$.*

PROOF. Fix $\ell = \ell(\varepsilon)$ and let $r = \lfloor (n-1)/\ell \rfloor$. $\mathfrak{P}_{k,\ell}(n)$ is illustrated in Figure 2.

$\mathfrak{P}_{k,\ell}(n)$ has input nodes $x_1, \ldots, x_n$ and output nodes $y_1, \ldots, y_n$. $\mathfrak{P}_{k,\ell}(n)$ has r subcircuits $\mathfrak{C}_o, \ldots, \mathfrak{C}_{r-1}$ each a copy of $\mathfrak{P}_k(\ell)$, and an additional subcircuit $\mathfrak{C}_r$, which is a copy of $\mathfrak{P}_k(n - r\ell)$.

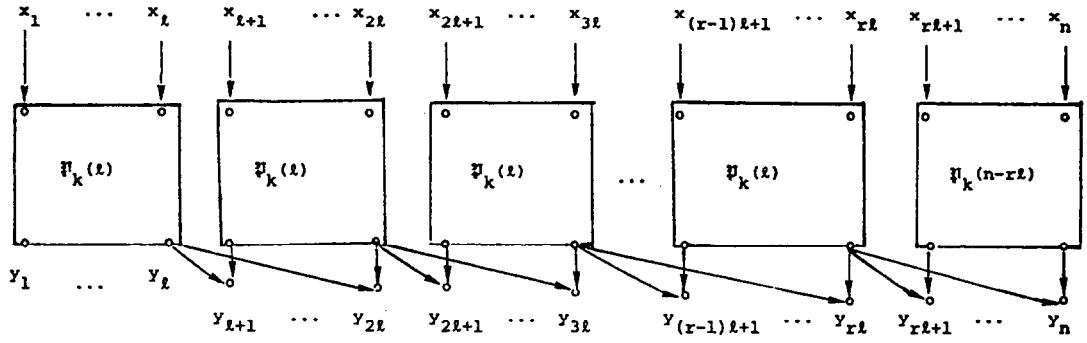

Figure 2. Our product circuit $\mathfrak{P}_{k,\ell}(n)$ for probabilistic prefix computation.

For $j = 0, \ldots, r$ and $i = 1, \ldots, \ell$, where $j\ell + i \leq n$, the i^{th} input to the subcircuit $\mathfrak{C}_j$ is $x_{j\ell+i}$. For $i = 1, \ldots, \ell$, the i^{th} output of $\mathfrak{P}_{k,\ell}(n)$ is the i^{th} output of $\mathfrak{C}_o$, which is always $y_i = x_1 \circ \cdots \circ x_i$, since $\mathfrak{C}_o$ is a copy of the errorless circuit $\mathfrak{P}_k(\ell)$. For $j = 1, \ldots, r$ and $i = 1, \ldots, \ell$ where $j\ell + i \leq n$, the $(j\ell+i)^{\text{th}}$ output of $\mathfrak{P}_{k,\ell}(n)$ is the composition of the ℓ^{th} output node of $\mathfrak{C}_{j-1}$ and the i^{th} output node of $\mathfrak{C}_j$. This gives $y_{j\ell+i} = x_{(j-1)\ell+1} \circ \cdots \circ x_{j\ell} \circ x_{j\ell+1} \circ \cdots \circ x_{j\ell+i}$, so $\text{prob}(Y_{j\ell+i} = x_1 \circ \cdots \circ x_{j\ell+i}$ for all $i = 1, \ldots, \ell) \geq \text{prob}((x_{(j-1)\ell+1} \circ \cdots \circ x_{j\ell}$ is prefix invariant$) \geq 1 - \varepsilon$. Hence, the probability that there is an error is at most $r\varepsilon \leq n\varepsilon/\ell$.

The size and depth bounds can now be computed from known bounds on $\mathfrak{P}_k$. We have $\text{size}(\mathfrak{P}_{k,\ell}(n)) \leq 2(n-\ell) + r \cdot \text{size}(\mathfrak{P}_k(\ell)) + \text{size}(\mathfrak{P}_k(n-r\ell)) \leq 2(n-\ell) + \text{size}(\mathfrak{P}_k(n)) \leq (6+2^{2-k})\,n - 2\ell(\varepsilon) - o(n)$. Furthermore, we have $\text{depth}(\mathfrak{P}_{k,\ell}(n)) \leq 1 + \max(\text{depth}(\mathfrak{P}_k(\ell)), \text{depth}(\mathfrak{P}_k(n-r\ell)) \leq k + \lceil \log \ell(\varepsilon) \rceil + 1$. ∎

2.4. Error-Free Product Circuits for Probabilistic Prefix Computations

Practical applications generally require that circuits make no undetected errors. Given probabilistic semigroup $\langle D, \circ, d \rangle$, we give error-free circuits with extended basis $\{\circ, \wedge, \vee, \neg, =, \tau\} \cup D$ containing the product operation $\circ$, the usual Boolean operations $\wedge, \vee, \neg$ equality $=$, an operation τ such that

$$\forall v \in \{0,1\}, a_1, a_2 \in D, \qquad \tau(v, a_1, a_2) = \begin{cases} a_1 & \text{if } v = 1, \\ a_2 & \text{if } v = 0, \end{cases}$$

and the basis also contains the elements of D construed to be constant functions.

We allow these error-free circuits to indicate that the outputs have been computed by setting a distinguished Boolean termination switch. In the case this switch evaluates to one, then the rest of the circuit need not be evaluated. Otherwise, the rest of the circuit must be evaluated before the outputs can be known. Let the *delay* be the number of parallel stops required until the termination switch is set. Thus, the delay is a random variable depending on the distribution of inputs. (Note that the delay can be considerably less than the depth, which is independent of the inputs.) In any case, the output never gives any errors.

We derive our errorless circuis $\mathfrak{Q}_{k,\ell}(n)$ from our product circuit $\mathfrak{P}_{k,\ell}(n)$ defined above. The outputs of $\mathfrak{P}_{k,\ell}(n)$ are correct if each $x_{(j-1)\ell+1} \circ \cdots \circ x_{j\ell}$ is a prefix invariant for $j = 1, \ldots, r$. For a given j, this can be verified by taking the disjunction of the equality tests $x_{(j-1)\ell+1} \circ \cdots \circ x_{j\ell} = a$ for each $a \in D$ which is a prefix invariant. The *termination switch* for our circuit is a Boolean conjunction of these r disjunctions for all $j = 1, \ldots, r$. In the case the termination switch evaluates to 1, then the outputs to $\mathfrak{Q}_{k,\ell}(n)$ are identical to $\mathfrak{P}_{k,\ell}(n)$, and otherwise we use the outputs computed by the errorless prefix circuit $\mathfrak{P}_k(n)$ and then set the termination switch to true. The correct output for $i = 1, \ldots, 2\ell$ is given by the i^{th} output of $\mathfrak{P}_{k,\ell}(n)$, and for $i = 2\ell + 1, \ldots, n$ is given by using a τ operation as the i^{th} output node of $\mathfrak{Q}_{k,\ell}(n)$. The arguments of this τ operation are first the output switch, then the i^{th} output node of $\mathfrak{P}_{k,\ell}(n)$, and last the i^{th} output node of $\mathfrak{P}_k(n)$. Thus, we have the following result.

THEOREM 1. *For any $n \geq 1$, $0 \leq k \leq \log n$ and $0 < \epsilon < 1/n$, there is an error-free circuit for probabilistic prefix computation which has constant fan-in (except at the termination switch), size $\left[13 + 2^{3-k} + (3|D| + 1)/\ell(\epsilon)\right] n - 8\ell(\epsilon) - o(n)$, depth $k + \lceil \log n \rceil + 2$, but delay less than $k + \lceil \log \ell(\epsilon) \rceil + 5$ with probability at least $1 - n\epsilon/\ell(\epsilon)$.*

PROOF. We use our error-free circuit $\mathfrak{Q}_{k,\ell}(n)$ where $\ell = \ell(\epsilon)$. The depth of the entire circuit $\mathfrak{Q}_{k,\ell}(n)$ is the max depth of $\mathfrak{P}_k(n)$ or of $\mathfrak{P}_{k,\ell}(n)$ plus 1, which is at most $k + \lceil \log n \rceil + 2$. By the proof of Lemma 1, the probability that the entire circuit need be evaluated is at most $n\epsilon/\ell$, and otherwise the evaluation time for $\mathfrak{Q}_{k,\ell}(n)$ is the depth of $\mathfrak{P}_{k,\ell}(n)$ plus 4. The size of $\mathfrak{Q}_{k,\ell}(n)$ is at most $(3|D| + 1)r + 3(n - 2\ell)$ plus the sum of the sizes of $\mathfrak{P}_{k,\ell}(n)$ and $\mathfrak{P}_k(n)$. ∎

This size bound can further be reduced by observing that the $\mathfrak{P}_k(\ell)$ subcircuits required by $\mathfrak{P}_{k,\ell}(n)$ can be found in $\mathfrak{P}_k(n)$.

3. APPLICATIONS OF PROBABILISTIC PREFIX COMPUTATION CIRCUITS

3.1. Finite State Transducers with Random Input

A (Mealy) deterministic *finite state transducer* is a six-tuple $M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$ where Q is a finite set of states, q_0 is the initial state, Σ, Δ are the finite *input* and *output alphabets*, respectively. $\delta : Q \times \Sigma \to Q$ is the *transition function* and $\lambda : Q \times \Sigma \to \Delta$ is the *output function*. Ladner and Fischer [1] show that computing the output of a deterministic finite state transducer can be expressed as a parallel prefix computation. We now generalize this to random inputs. Fix a density function $d : \Sigma \to [0, 1]$. We assume that M takes random input $x_1, \ldots, x_n \in \Sigma$ such that each input x_i is an independent random variable chosen with density function d. (For example, Figures 2.1 and 2.2 give finite state diagrams for some arithmetic operations on uniformly distributed random binary numbers. An arc in the diagrams from state q to state q' is labelled $(a/z) : \rho$ if $\delta(q, a) = q'$ is the new state, $\lambda(q, a) = z$ is the output and $f(a) = \rho$ is the probability of the transition.)

We consider each input symbol $a \in \Sigma$ to be a mapping $a : Q \to Q$ such that $a(q) = \delta(q, a)$ for all $q \in Q$. Thus, the domain $D = \Sigma$ is the set of functions $Q \to Q$. Let $a_1 \circ a_2(q) = a_2(a_1(q))$ for each $a_1, a_2 \in D$ and $q \in D$. Thus, $\langle D, \circ, d \rangle$ is a probabilistic semigroup as defined in Section 2.1. Let $\ell(\epsilon)$ be the system's ϵ-dependence length.

Given input $x_1, \ldots, x_n \in \Sigma$, the prefix computation gives for $i = 1, \ldots, n$ a function $x_1 \circ \cdots \circ x_n$ which maps the initial state q_0 to the state $q_i = x_1 \circ \cdots \circ x_i(q_0)$ reached on the i^{th} step. The *output* string is $\lambda(q_0, x_1), \lambda(q_1, x_2), \ldots, \lambda(q_{n-1}, x_n)$.

We can use the probabilistic prefix circuits $\mathfrak{P}_{0,\ell(\varepsilon)}(n)$ of Lemma 1 to compute the $x_1 \circ \cdots \circ x_i$ functions. Each of the required functional compositions over constant size domains require only constant size of circuitry and constant delay. Hence, we have the following lemma.

LEMMA 2. *The output of M can be computed by a product circuit with constant fan-in, size $O(n)$, depth $O(\log \ell(\varepsilon))$ and error probability at most $n\varepsilon/\ell(\varepsilon)$.*

Applying Theorem 1, we get the following theorem, using the circuit $\mathfrak{Q}_{o,\ell(\varepsilon)}(n)$.

THEOREM 2. *The output of M can be computed by an error-free circuit with constant fan-in except at one node, size $O(n)$, and delay $O(\log \ell(\varepsilon))$ with probability at least $1 - n\varepsilon/\ell(\varepsilon)$.*

3.2. Circuits for Binary Addition and Subtraction of Random Numbers

We design circuits for addition and subtraction of two uniformly distributed random n-bit binary numbers $x^{(1)} = x_n^{(1)} x_{n-1}^{(1)} \ldots x_1^{(1)}$ and $x^{(2)} = x_n^{(2)} x_{n-1}^{(2)} \ldots x_1^{(2)}$. The input is $x_1, \ldots, x_n$, where the i^{th} input x_i is the concatenation of $x_i^{(1)}$ and $x_i^{(2)}$ and thus gives the i^{th} least significant bits of $x^{(1)}$ and $x^{(2)}$. The input alphabet $\Sigma = \{00, 01, 10, 11\}$ then consists of pairs of bits, when

each possible pair has equal probability 1/4. The state set $Q = \{q_o, q_1\}$ consists of an initial state q_0 with carry 0, and a state q_1, with carry 1. The output functions are given explicitly by Figures 3a and 3b. In the case of addition (see Figure 3a), 00 is considered the mapping $q_0 \to q_0$, $q_1 \to q_0$ and 11 is the mapping $q_0 \to q_1$, $q_1 \to q_1$. Note that both 00 and 11 are prefix invariants, but the other transitions 01 and 10 are not prefix invariants. Since 00 and 11 have total probability 1/2, and $\{01, 10\}*$ is exactly the set of words over Σ^*, which are not prefix invariant, we have $\text{prob}(x_1 \circ \cdots \circ x_\ell$ is prefix invariant$) = \text{prob}(x_i = 00$ or 11 for some $1 \le i \le \ell) = 1 - 1/2^\ell$. This implies from the definition of ε-dependence length that the ε-dependence length for addition is $\ell(\varepsilon) \le \lceil -\log(\varepsilon) \rceil$ for $0 < \varepsilon < 1$.

In the case of subtraction (see Figure 3b), both 00 and 11 are the mapping $q_0 \to q_0$, $q_1 \to q_1$, 01 is the prefix invariant mapping $q_0 \to q_1$, $q_1 \to q_1$, and 10 is the prefix invariant mapping $q_0 \to q_0$, $q_1 \to q_0$. Each input x_i has probability 1/2 of being prefix invariant, so

$$\text{prob}\,(x_1 \circ \cdots \circ x_\ell \text{ is prefix invariant}) = \text{prob}\,(x_i = 01 \text{ or } 10 \text{ for some } 1 \le i \le \ell) = 1 - \frac{1}{2^\ell},$$

and thus, the ε-dependence length for subtraction is $\ell(\varepsilon) \le \lceil -\log(\varepsilon) \rceil + 1$ for $0 < \varepsilon < 1$.

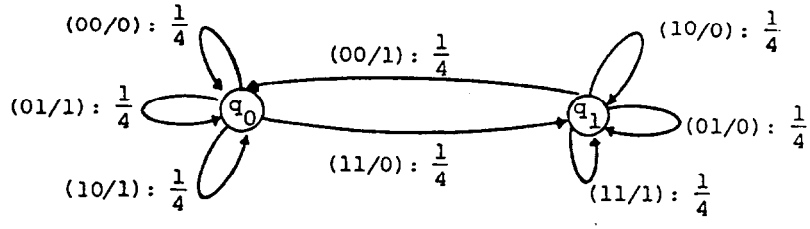

Figure 3a. A finite state transducer for addition of random binary numbers. Each transition has equal probability 1/4 on random input.

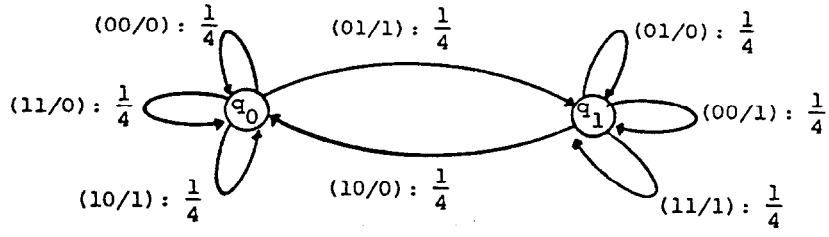

Figure 3b. A finite state transducer for subtraction of random binary numbers. Each transition has equal probability 1/4 on random input.

Applying Lemma 2 and Theorem 2, we get the following corollary for any $\alpha > 1$ and integer $n \ge 0$.

COROLLARY 1. *There are Boolean circuits for addition and subtraction of random n-bit binary numbers with*

(a) *constant fan-in, linear size, depth* $O(\log((\alpha+1)\log n))$ *and error probability at most* $n^{-\alpha}$;
(b) *also, errorless Boolean circuits with constant fan-in except for a single node, linear size, depth* $O(\log n)$, *but delay at most* $O(\log((\alpha+1)\log n)$ *with probability at least* $1 - n^{-\alpha}$.

3.3. Circuits for Constant Multiplication and Division of Random Numbers

We now consider the operation of multiplying a uniformly distributed random binary number x by an integer constant $m \ge 2$. It is convenient to assume x consists of bn random bits, where $b = \lceil \log m \rceil$. We will partition x into n consecutive blocks. So $x = x_n, \ldots, x_1$, where each x_i is assumed to be independently randomly chosen from $\Sigma = \{0, 1, \ldots, 2^b - 1\}$. The state set is $Q = \{q_o, \ldots, q_{m-1}\}$, where q_0 is the initial state. For each $c = 0, \ldots, m-1$, state q_c is associated with a carry of c. Given input symbol $x_i \in \Sigma$ in state q_c, the output h is the residue mod 2^b

of $x_i\, m + c$ and x_i defines a transition to state q_c, with carry $c' = (x_i\, m + c - h)\, 2^{-b}$. Note that any input symbol $x_i = 0$ is prefix invariante since then, the transition from any state is to state q_0. Hence, the probability of a prefix invariant input symbol is at least $1/2^b$. This implies $\text{prob}(x_1 \circ \cdots \circ x_\ell \text{ is prefix invariant}) \geq 1 - (1 - 2^{-b})^\ell$. So for the case of multiplication by m, the ε-dependence length is $\ell(\varepsilon) \leq \lceil \log(\varepsilon)/\log(1 - 2^{-b}) \rceil + 1$ for $0 < \varepsilon < 1$. A similar argument gives the same upper bounds on the ε-dependence length for division of a random number by m. (NOTE: In the special case where m is a power of two, then the ε-dependence length for both these operations is $\ell(\varepsilon) = 1$ for all ε.)

Lemma 2 and Theorem 2 imply for any $\alpha > 0$ and integers $m \geq 2$, $n \geq 0$ and $\beta = -(\alpha + 1)/\log(1 - 2^{-b})$.

COROLLARY 2. *There are Boolean circuits for multiplication and division of a random* $\lceil \log m \rceil$ *n-bit binary number by integer m, with*

(a) *constant fan-in, linear size, depth* $O(\log(\beta \log n))$ *and error probability at most* $n^{-\alpha}$, *and*
(b) *also, errorless circuits with constant fan-in except at a single node, size* $O(n)$, *depth* $O(\log n)$, *but delay at most* $O(\log(\beta \log n))$ *with probability at least* $1 - n^{-\alpha}$.

NOTE: Of course, a circuit for multiplication by a power of two is trivial, since the required bit shifts can be done by simply defining outputs at the appropriate input nodes.

4. LOWER BOUNDS ON CIRCUIT DEPTH FOR COMPUTATIONS WITH RANDOM INPUTS

In this section, we devise a general technique for proving lower bounds on the depth of any circuit on computing a function $F(x_1, \ldots, x_n)$ with given error where inputs $x_1, \ldots, x_n$ are independent identically distributed random variables. Then, we apply this lower bound technique to some arithmetic problems of interest. Let a circuit be *x_i-oblivious* if its output does not depend on input x_i.

LEMMA 3. *Any x_i-oblivious circuit $\mathfrak{C}$ computes $F(x_1, \ldots, x_n)$ with error probability at least $\sigma_i/|D|$, where $\sigma_i = \text{prob}(\exists\ a \in D$ such that $F(x_1, \ldots, x_n) \neq F(x_1, \ldots, x_{i-1}, a, x_{i+1}, \ldots, x_n))$.*

PROOF. Let $S_i = \{(a_1, \ldots, a_n) \in D^n | \exists\ a' \in D$ such that $F(a_1, \ldots, a_n) \neq F(a_1, \ldots, a_{i-1}, a', a_{i+1}, \ldots, a_n)\}$ so $\sigma_i = |S_i|/|D|^n$. Let $E = \{(a_1, \ldots, a_n) \in D^n |$ on input $(a_1, \ldots, a_n)$, $\mathfrak{C}$ does not compute $F(a_1, \ldots, a_n)\}$ and let $H = (D^n - E) \cap S_i$.

Suppose $\mathfrak{C}$ has error probability less than $\sigma_i/|D|$. Then, $|E| < |S_i|/|D|$, so we have $|H| > |S_i|(1 - \frac{1}{|D|}) \geq 0$. Hence, H is not empty, so $\exists \vec{a} = (a_1, \ldots, a_n)$, $\vec{a}' = (a_1, \ldots, a_{i-1}, a', a_{i+1}, \ldots, a_n)$ such that $\vec{a}, \vec{a}' \in H$ and $F(\vec{a}) \neq F(\vec{a}')$.

Since $\vec{a}, \vec{a}' \in D^n - E$, $\mathfrak{C}$ outputs $F(\vec{a})$ and $F(\vec{a}')$ on inputs $\vec{a}, \vec{a}'$, respectively. But $\mathfrak{C}$ is x_i-oblivious, so $\mathfrak{C}$ must compute the same output for both $\vec{a}$ and $\vec{a}'$, a contradiction. ∎

For each ε, $0 < \varepsilon < 1$, we define *input dependence* of the system to be $I(\varepsilon, n) = |\{i | \sigma_i/|D| \geq \varepsilon, 1 \leq i \leq n\}|$.

THEOREM 3. *Any circuit for $F(x_1, \ldots, x_n)$ with n inputs, error less than ε and fan-in f has depth at least $\log_f(I(\varepsilon, n))$.*

PROOF. By contradiction. Suppose there exists a circuit $\mathfrak{C}$ of fan-in f which computes $F(x_1, \ldots, x_n)$ with error less than ε. If the depth of $\mathfrak{C}$ is less than $\log_f(I(\varepsilon, n))$, then $\mathfrak{C}$ must be oblivious to more than $n - I(\varepsilon, n)$ inputs. But then, $\mathfrak{C}$ must be x_i-oblivious for some i where $\sigma_i/|D| \geq \varepsilon$. So by Lemma 3, $\mathfrak{C}$ has error at least ε, a contradiction. ∎

For our applications to arithmetic on uniformly distributed random binary numbers, we show the corresponding σ_i are geometrically decreasing functions of i. Hence, $I(n^{-\alpha}, n) \geq \Omega(\log n)$ giving $\log_f \log n - o(n)$ depth lower bounds with fan-in f.

For example, in the case of addition or subtraction of two random n-bit binary numbers, $\sigma_i \geq 1/2^{n+2-i}$ for $i = 1, \ldots, n$. We prove this for only the case of addition (the arguments for the

case of subtraction are similar). Let the random input be $x_1, \ldots, x_n$, as described in Section 3.2. Let A_i be the predicate holding just when $x_j = 01$ or 10 for each $j = i+1, \ldots, n$. Since $\text{prob}(x_j = 01 \text{ or } 10) = 1/2$, $\text{prob}(A_i) = 1/2^{n-i}$. A_i implies that $x_{i+1} \circ \cdots \circ x_n$ is not prefix invariant. Hence, for each $a \in \{00, 01, 10, 11\}$, $\text{prob}(x_1 \circ \cdots \circ x_{i-1} \circ a \circ x_{i+1} \circ \cdots \circ x_n \neq x_1 \circ \cdots \circ x_n | A_i) \geq 1/4$. Thus, $\sigma_i \geq \text{prob}(A_i) \cdot 1/4 = 1/2^{n+2-i}$.

Since $|D| = 4$, this implies for addition and subtraction $\sigma_i/|D| \geq n^{-\alpha}$ for all the indices i when $n + 4 - \alpha \log n \leq i \leq n$. Hence, $I(n^{-\alpha}, n) \geq (\alpha \log n) - 4$, so we have, as a consequence of Theorem 3, the following corollary.

COROLLARY 3. *Any Boolean circuit of fan-in f and error at most $n^{-\alpha}$ for addition or subtraction of random n-bit binary numbers must have depth at least $\log_f(\alpha \log n - 4)$.*

Thus, we conclude that our random addition and subtraction circuit of fan-in 2 and error $n^{-\alpha}$, given by Corollary 1, have asymptotic optimal depth.

Next, we show for m not divisible by two, $\sigma_i \geq 1/(2m)^{n+1-i}$ for $i = 1, \ldots, n$ for multiplication or division of a random $\lceil \log m \rceil n$-bit binary number by m. Let the input be partitioned into blocks $x_1, \ldots, x_n$ of length $b = \lceil \log m \rceil$ described in Section 3.3, so each $x_i \in \{0, 1, \ldots, 2^b - 1\}$. Let B_i be the predicate holding just when $2^b - x_j = s$ is the multiplicative inverse of m modulo 2^b for each $j = i + 1, \ldots, n$. Since $\text{prob}(x_j = 2^b - s) = 2^{-b}$, $\text{prob}(B_i) = 2^{-b(n-i)}$. B_i implies $x_j m = -1 \bmod 2^b$ for each $j = i + 1, \ldots, n$ so $x_{i+1} \circ \cdots \circ x_n$ is not prefix invariant. Hence, for each $a \in \{0, 1, \ldots, 2^b - 1\}$, $\text{prob}(x_1 \circ \cdots \circ x_{i-1} \circ a \circ x_{i+1} \circ \cdots \circ x_n \neq x_1 \circ \cdots \circ x_n | B_i) \geq 2^{-b}$. Thus, $\sigma_i \geq \text{prob}(B_i) \cdot 2^{-b} = 2^{-b(n-i)}\, 2^{-b} \geq (2m)^{-(n-i+1)}$, since $2^{-b} > 1/(2m)$.

Since in this case $|D| = 2^b$, this implies for multiplication and division by m, $\sigma_i/|D| \geq n^{-\alpha}$ for all indices i where $n + 2 - (\alpha \log n)/\log(2m) \leq i \leq n$. Hence, $I(n^{-\alpha}, n) \geq (\alpha \log n)/\log(2m) - 2$, so by Theorem 3, we have the following corollary.

COROLLARY 4. *Any Boolean circuit of fan-in f and error at most $n^{-\alpha}$ for multiplication or division of random $\lceil \log m \rceil n$-bit binary numbers by m, must have depth at least $\log_f((\alpha \log n)/\log(2m) - 2)$, if m is not divisible by two.*

When m is constant and not divisible by 2, this implies the asymptotic optimality of our constant multiplication and division circuits given by Corollary 2.

5. CONCLUSION

Ladner and Fischer [1] observed that many arithmetic operations of practical interest can be sequentially computed with finite memory, so their prefix circuits can compute these arithmetic operations in parallel. It is our observation that for random inputs, these arithmetic operations have prefix invariant properties which allow us to design errorless circuits which have much less expected delay time.

Similarly, a practical design for an efficient circuit for any other problem can, in principle, take into account the distribution of inputs expected in applications. Generally, empirical experimentation of a circuit is done to derive timing parameters. These parameters may be set so that on the vast majority of inputs, the circuit performs correctly, but in a few cases of low likelihood, there may be an error. Such errors might be detected and further corrective computation done, incurring some additional delay with low probabiltiy.

Perhaps the most interesting aspect of the work in this paper is our analytic derivation of timing parameters such as expected delay time. We hope these probabilistic analysis techniques introduced in our paper may be illustrative of how a theoretical result might contribute to the practical aspects of circuit design and parallel computation. In particular, our analysis avoids the repeated experimental execution of circuits on random inputs which might otherwise be used in practice to determine timing parameters of the circuits.

REFERENCES

1. R.E. Ladner and M.J. Fischer, Parallel prefix computation, *Journal of the Assoc. for Computing Machinery* **27** (4), 831–838 (October 1980).
2. F. Fich, New bounds for parallel prefix circuits, In *Proc. of the* 15th *Annual Symp. on Theory of Computation, Boston, MA*, (May 1983).
3. A.W. Burks, H.H. Goldstine and J. von Neumann, Preliminary discussion of the logical design of an electronic computing instrument, Part 1, Vol. 1, Inst. Advanced Study, Princeton, NJ, (1946).
4. G.W. Reifwiesner, The determination of carry propagation length for binary addition, *IRE Transactions on Electronic Computers*, 35–38 (March 1960).
5. B. Gilchrist, J. Pomerene and S.Y. Wong, Fast carry logic for digital computers, *IRE Trans. Electron. Computers* **4**, 133–136 (1955).
6. H.C. Hendrickson, Fast high-accuracy binary parallel addition, *IRE Transactions on Electronic Computers*, 465–469 (December 1960).
7. T. Kilburn, D.B.G. Edwards and D. Aspinall, Parallel addition in digital computers, a new fast carry circuit, *Proc. IEE, Pt. B* **106**, 464–466 (1959).
8. T. Kilburn, D.B.G. Edwards and D. Aspinall, A parallel arithmetic unit using a saturated-transistor fast-carry circuit, *Proc. IEE, Pt. B* **107**, 573–584 (1960).
9. C. Tung, Arithmetic, In *Computer Science,* (Edited by A.F. Cardenas, L. Presser and M.A. Marin), Wiley-Interscience, New York, (1972).
10. V.M. Krapchenko, Asymptotic estimation of addition time of a parallel adder., *Syst. Theory Res.* (Probl. Kibern. **19**, 107–122 (Russ.)) **19**, 105–122 (1970).
11. A. Chandra, S. Fortune and R. Lipton, Unbounded fan-in circuits and associative functions, In *Proc. of the* 15th *Annual Symp. on Theory of Computation, Boston, MA*, (May 1983).
12. S. Winograd, On the time required to perform addition, *J. of Assoc. Computing Machinery* **12**, 277–285 (1965).
13. T.L. Booth, *Sequential Machines and Automata Theory*, Wiley, New York, (1967).
14. R. Brent, On the addition of binary numbers, *IEEE Trans. Comput.* **C-19** (8), 758–759 (1970).
15. M. Lehman and N. Burla, Skip techniques for high-speed carry-propagation in binary arithmetic units, *IRE Trans. Electron. Computers* **10**, 691–698 (1961).
16. M. Lehman and N. Burla, A note on the simultaneous carry generation system for high-speed adders., *IRE Trans. Electron. Computers* **9**, 510 (1960).
17. G. Metze and J.E. Robertson, Elimination of carry propagation in digital computers, *Proc. Intern. Conf. Inform. Processing,* Paris 1959 UNESCO/NS/ICIP/G.2 **10**, 389–396 (1959).
18. Y. Ofman, On the algorithmic complexity of discrete functions, *Sov. Phys. Dokl.* **7**, 589–591 (1963).
19. M.S. Paterson, An introduction to Boolean function complexity, *Societe Math. de France Asterisque* (38/39), 183–201, (Also Tech. Rep. STAN-CS-76-557, Computer Science Department, Stanford University, Stanford, CA, August 1976) (1976).
20. J.E. Savage, *The Complexity of Computing*, Wiley, New York, (1976).
21. A. Schonhage, A lower bound for the length of addition chains, *Theor. Comput. Sci.* **1**, 1–12 (1975).