ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

J. Parallel Distrib. Comput. ▮▮▮ (▮▮▮▮) ▮▮▮–▮▮▮

Journal of Parallel and Distributed Computing

www.elsevier.com/locate/jpdc

Faster optimal parallel prefix circuits: New algorithmic construction

Yen-Chun Lin[a,*], Chin-Yu Su[b]

[a]*Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, 43 Keelung Road, Sec. 4, Taipei 106, Taiwan*

[b]*Department of Electronic Engineering, National Taiwan University of Science and Technology, 43 Keelung Road, Sec. 4, Taipei 106, Taiwan*

Received 19 August 2004; received in revised form 6 January 2005; accepted 30 May 2005

Abstract

Parallel prefix circuits are parallel prefix algorithms on the combinational circuit model. A prefix circuit with n inputs is depth-size optimal if its depth plus size equals $2n-2$. Smaller depth implies faster computation, while smaller size implies less power consumption, less VLSI area, and less cost. To be of practical use, the depth and fan-out of a depth-size optimal prefix circuit should be small. A circuit with a smaller fan-out is in general faster and occupies less VLSI area. In this paper, we present a new algorithm to design parallel prefix circuits, and construct a class of depth-size optimal parallel prefix circuits, named *SU4*, with fan-out 4. When $n \geqslant 30$, *SU4* has the smallest depth among all known depth-size optimal prefix circuits with fan-out 4.

Keywords: Combinational circuits; Depth; Depth-size optimal; Fan-out; Parallel algorithms; Prefix computation; Size optimal

1. Introduction

Prefix computation is a very useful and basic operation [4]. Many have reported its important role in various applications, such as cryptography, binary addition, biological sequence comparison, design of silicon compilers, image processing, job scheduling, loop parallelization, polynomial evaluation, processor allocation, and sorting [1–3,8,10,11,17–19,21,40,42,44,45]. The prefix operation can be defined as follows: Given n values $x_1, x_2, \ldots, x_n$ and an associative binary operation $\otimes$, compute the n values $x_1 \otimes x_2 \otimes \cdots \otimes x_i$, $1 \leqslant i \leqslant n$. Note that the associative binary operation performed in the prefix sums operation is arithmetic addition [8,22]; that is, the prefix sums operation is a special case of the prefix operation.

To accelerate the prefix operation, many parallel prefix algorithms for various parallel computing models have also been proposed [1,7,9,15,17,19,22,23,27,31,32,34–36]. In addition, the prefix operation is built in for MPI parallel programming [13] and implemented in hardware in the thinking machines CM-5 [39].

Parallel prefix circuits are parallel prefix algorithms for the combinational circuit model of computation [1]. Many parallel prefix circuits have been devised and studied [3,5,6,12,14,16,18–20,23–26,28–30,33,37,43–45]. Combinational circuits may be closely related to other parallel computing models. For example, some combinational circuits are known to have corresponding algorithms for other parallel models [19,23]. On the other hand, prefix circuits can be components of algorithms for other models, and can be building blocks for other models [1].

In this paper, we assume that the number of inputs is n, unless otherwise stated. A prefix circuit is represented as a directed acyclic graph containing n input nodes, n output nodes, at least $n-1$ operation nodes, and at least one duplication node. As shown in Fig. 1, an operation node, represented by a black dot, performs the $\otimes$ operation on its two inputs, having indegree 2 and outdegree 1 or more. A duplication node has indegree 1 and outdegree at least 2, denoted by a small circle also in Fig. 1; it takes an input to produce multiple copies of the input as output. Because only the

* Corresponding author. Fax: +886 2 2730 1081.
E-mail address: yclin@computer.org (Y.-C. Lin).

0743-7315/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jpdc.2005.05.017

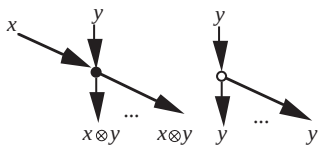

Fig. 1. Operation node and duplication node.

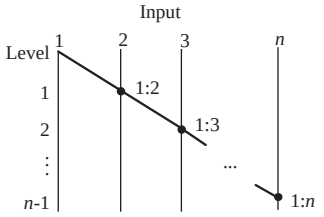

Fig. 2. Sequential prefix circuit $S(n)$.

duplication node has indegree 1 and outdegree at least 2, it need not and will not be explicitly represented by a small circle. Similarly, input and output nodes will not be explicitly represented by any shape, because they can be easily identified. For ease of presentation, in this paper, all the directed edges are assumed to be downward; thus, the arrows need not be shown.

A serial prefix circuit $S(n)$ is shown in Fig. 2, in which vertical edges from left to right are named line 1, line 2, ..., line n, respectively. Input nodes are on the top of a circuit and have indegree 0 and outdegree 1, representing input items. The input node on line i represents input x_i. Output nodes are at the bottom and have indegree 1 and outdegree 0, representing outputs. For any operation node on line i at level j, its left input is from a node at level $j-1$, while its right input is always from line i. In Fig. 2, a duplication node is on line 1 at level 0. The numbers at the left of a prefix circuit denote the depth levels of the nodes to the right. When no confusion is caused, we may use simply D for any prefix circuit $D(p)$, where p may be the number of inputs or a parameter with another meaning.

The size of a prefix circuit D, $s(D)$, is the number of operation nodes in D, and the depth of D, $d(D)$, is the maximum level of operation nodes in D. Smaller depth implies faster computation. Smaller size implies less power consumption and less area in VLSI implementation and thus less cost. Therefore, size and depth are important parameters of prefix circuits. For any prefix circuit D, $d(D)+s(D)\geqslant 2n-2$ [37]. Thus, D is depth-size optimal, or optimal for short, if $d(D)+s(D)=2n-2$. For example, as Fig. 2 has shown, $s(S)=d(S)=n-1$; thus, S is optimal.

Fan-out is the third important parameter. The fan-out of a node is its outdegree. A node having a smaller fan-out is faster and smaller in VLSI implementation [41]. A node has unbounded fan-out if the fan-out is not fixed and is a function of n; otherwise, the fan-out of the node is a constant, or is bounded. The fan-out of prefix circuit D, $fo(D)$, is the maximum fan-out of all nodes in D. For example, $fo(S)=2$. A circuit should have a small bounded fan-out for it to be of practical use.

For ease of presentation, let $i{:}j$ represent the result of computing $x_i\otimes x_{i+1}\otimes\cdots\otimes x_j$, where $i\leqslant j$. We use $ia(D)=a$ to denote that line 1 of prefix circuit D has a duplication node at level a and has no duplication nodes at any level less than a. In addition, we use $l(D)=b$ to denote that line n of D obtains $1{:}n$ at level b.

This paper is on designing new optimal parallel prefix circuits with fan-out 4 and the smallest depth. Such prefix circuits are equivalent to the fastest parallel prefix algorithms on a fully connected multicomputer of n nodes, in which a node can at most receive a message and send out three messages in a communication step and the number of communication steps plus the number of messages equals the lower bound $2n-2$ [23]. Many previous parallel prefix circuits are briefly reviewed in [26]. A recent paper constructs new depth-size optimal prefix circuits with the minimum depth; however, they have unbounded fan-out [43]. So far, *WE4* and *Z4* are optimal prefix circuits with fan-out 4 that have depths less than or equal to those of all the other previously known optimal prefix circuits with the same fan-out [24,25]. It is not easy or impossible, depending on the value of n, to devise optimal prefix circuits with fan-out 4 and a smaller depth, even only decreasing the depth by one. This paper introduces some new prefix circuits, and then systematically constructs a new optimal parallel prefix circuit with fan-out 4, named *SU4*, whose depth is less than or equal to those of *WE4* and *Z4*, for $n\geqslant 30$. A new algorithm is also introduced because of its important role in building *SU4*. The algorithm can construct an infinite number of prefix circuits from a prefix circuit with certain properties.

Section 2 reviews previous results, including the definition of a size optimal prefix circuit and related theorems, that will be used in later sections. Section 3 introduces two new classes of size optimal prefix circuits. These circuits are constructed by using our new algorithm. Section 3 also introduces some other new size optimal prefix circuits. Section 4 defines the optimal prefix circuit *SU4* by composition of prefix circuits given in previous sections. Section 5 compares the depth of *SU4* with those of *WE4* and *Z4*. Section 6 concludes this paper.

2. Review of previous results

2.1. Composition of prefix circuits

Assume that A and B are two prefix circuits with n_1 and n_2 inputs, respectively. By merging the operation node on line n_1 at level $l(A)$ of A with the duplication node on line 1 at level $ia(B)$ of B, A and B are composed into a larger prefix circuit, denoted by $A\bullet B$, with n_1+n_2-1 inputs [37]. Fig. 3 gives an example of composition. Note that the composition operation of prefix circuits is associative.

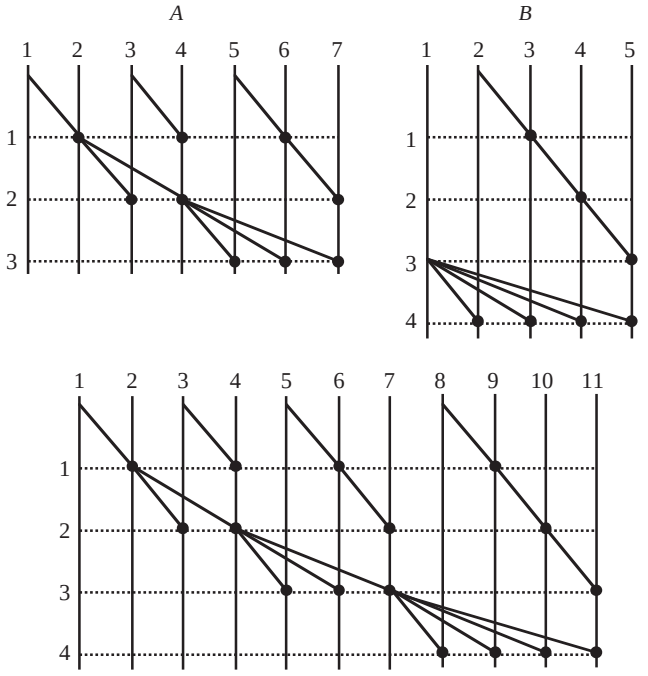

Fig. 3. Two example prefix circuits and their composition.

2.2. Size optimal prefix circuits

It has been shown that for any n-input prefix circuit A, if $ia(A) = a$, and $l(A) = b$, then $s(A) \geqslant 2\pi - 2 + a - b$ [26]. Therefore, we have the following definition:

Definition 1 (*Lin et al. [26]*). For any n-input prefix circuit A, if $ia(A) = a$, $l(A) = b$, and $s(A) = 2n - 2 + a - b$, then A is size optimal; we say that A is $SOPC(n, a, b)$.

Theorem 2 (*Lin et al. [26]*). *If A is $SOPC(n, 0, b)$ and $d(A) = b$, then A is optimal.*

Theorem 3 (*Lin et al. [26]*). *If A and B are $SOPC(n_1, a, b)$ and $SOPC(n_2, c, d)$, respectively, where $b \geqslant c$, then $A(n_1) \bullet B(n_2)$ is $SOPC(n_1 + n_2 - 1, a, b - c + d)$ with depth* $\max\{d(A), d(B) + b - c\}$ *and fan-out* $\max\{fo(A), fo(B)\}$.

Theorem 4 (*Lin et al. [26]*). *The sequential prefix circuit $S(n)$ is $SOPC(n, 0, n - 1)$ with fan-out 2 and depth $n - 1$.*

2.3. Layered prefix circuit P and its compacted version Q

A prefix circuit D can be defined with sets of operation nodes at level i, $i = 1, 2, \ldots, d(D)$ [37]:

$G_i = \{(x, y) \mid$ at level i on line y there is an operation node whose left input is the output of a node on line x at level $i - 1\}$.

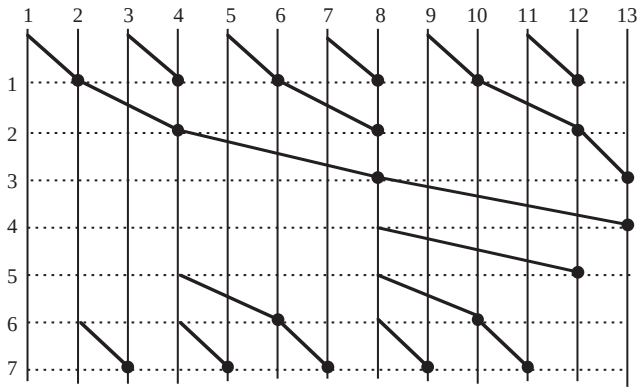

Fig. 4. Layered prefix circuit $P(13)$.

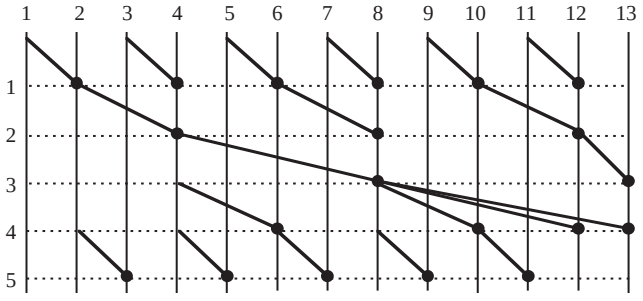

Fig. 5. $Q(13)$.

For example, for the prefix circuit shown in Fig. 4, $G_3 = \{(4, 8), (12, 13)\}$. If $(x, y) \in G_i$, the corresponding operation node can be denoted as $(x, y)_i$.

Let $m = \lceil \lg n \rceil$; the n-input layered prefix circuit $P(n)$ is defined with the following sets of operation nodes [37]:

$$\begin{aligned} G_t = \{&(k2^t - 2^{t-1}, \min(n, k2^t)) \mid k = 1, 2, \ldots, \\ &\lfloor (n-1)/2^t + 1/2 \rfloor\}, \quad t = 1, \ldots, m; \\ G_{m+t} = \{&(k2^{m-t}, k2^{m-t} + 2^{m-t-1}) \mid k = 1, 2, \ldots, \\ &\lfloor (n-1)/2^{m-t} - 1/2 \rfloor\}, \quad t = 1, \ldots, m-1. \end{aligned}$$

An example, $P(13)$, has already been shown in Fig. 4.

$P(n)$ can be compacted by an algorithm to be $Q(n)$, a prefix circuit with fan-out at most 4 and smaller depth [25,26,29]. For example, $P(13)$ can be compacted to be $Q(13)$ with depth 5 shown in Fig. 5.

Theorem 5 (*Lin and Shih [29]*). *The fan-out of $Q(n)$ is at most* 4, *and the depth of $Q(n)$ is*

$$\begin{cases} \lceil \lg n \rceil & \text{when } n \leqslant 7, \\ 2t - 2 & \text{when } t \geqslant 3 \text{ and } 2^t \leqslant n < 3 \times 2^{t-1}, \\ 2t - 1 & \text{when } t \geqslant 3 \text{ and } 3 \times 2^{t-1} \leqslant n < 2^{t+1}. \end{cases}$$

Theorem 6 (*Lin et al. [26]*). *$Q(n)$ is $SOPC(n, 0, \lceil \lg n \rceil)$.*

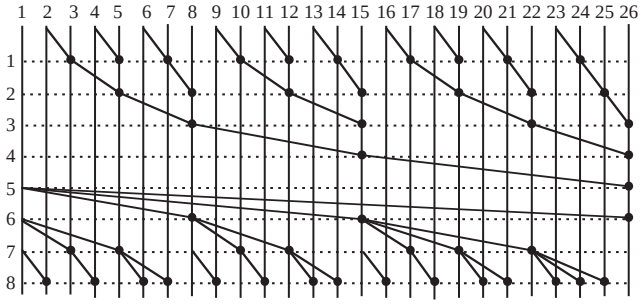

Fig. 6. *SY*(5).

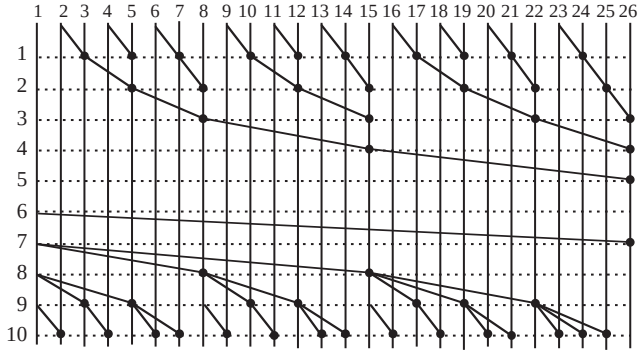

Fig. 7. *YA*(5).

3. Some new size optimal prefix circuits

In Sections 3.1–3.3, we will construct size optimal prefix circuits *SY*, *SZ*, and *Y*, respectively. In Section 3.4, we define two size optimal prefix circuits *GV* and *GW*. These prefix circuits will be used to construct our new optimal prefix circuit *SU4*.

3.1. SY: a size optimal prefix circuit with fan-out 4

Define a size optimal prefix circuit $SY(5)$ as depicted in Fig. 6. It has the following properties: $d(SY(5)) = 8$, $fo(SY(5)) = 4$, $ia(SY(5)) = 5$, $l(SY(5)) = 6$, and $s(SY(5)) = 49 = 2 \times 26 - 2 + 5 - 6$. Therefore, by Definition 1, $SY(5)$ is $SOPC(26, 5, 6)$.

We can move down the operation node $(1, 26)_6$ of $SY(5)$ by one level to become $(1, 26)_7$, and move down all the other operation nodes at levels 6–8 by two levels, to obtain the *YA*(5) shown in Fig. 7. It is $SOPC(26, 6, 7)$ with fan-out 4 and depth 10.

On the other hand, we can move down all the nodes at levels 6–8 of $SY(5)$ by two levels to obtain the *YB*(5) depicted in Fig. 8. It is $SOPC(26, 7, 8)$ with fan-out 4 and depth 10.

Fig. 9 shows $YA(5) \bullet YB(5)$. By Theorem 3, it is $SOPC(51, 6, 8)$ with fan-out 4. If we delete the node $(26, 51)_8$ of $YA(5) \bullet YB(5)$ and add nodes $(26, 51)_6$ and $(1, 51)_7$, resulting in the circuit $SY(6)$ shown in Fig. 10 with $fo(SY(6)) = 4$ and $ia(SY(6)) = 6$. It can be checked that $SY(6)$ is $SOPC(51, 6, 7)$ with $d(SY(6)) = 10$ and $l(SY(6)) = 7$.

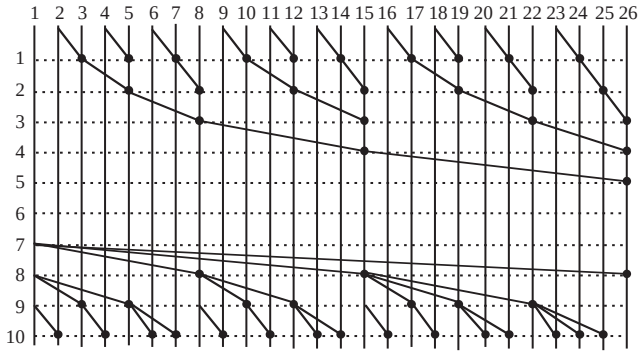

Fig. 8. *YB*(5).

The procedure for deriving $SY(6)$ from $SY(5)$ is generalized to be the following algorithm, which can be used repeatedly to derive an infinite number of prefix circuits from any single prefix circuit with some simple attributes.

Algorithm A(D, n, t, d). Given a prefix circuit $D(t)$ that is $SOPC(n,\ t,\ t+1)$ with depth d, where $t + 1 < d$. Let N be the set of nodes of $D(t)$ at levels $t+1$ through d. We can obtain prefix circuit $D(t+1)$ by the following procedures:

1. Construct $DA(t)$ by moving down the node $(1, n)_{t+1}$ of $D(t)$ by 1 level to become $(1, n)_{t+2}$, and moving down all the other nodes in N by 2 levels.
2. Construct $DB(t)$ by moving down all the nodes in N of $D(t)$ by 2 levels.
3. Delete the node $(n, 2n-1)_{t+3}$ of $DA(t) \bullet DB(t)$, and add nodes $(n, 2n-1)_{t+1}$ and $(1, 2n-1)_{t+2}$, resulting in $D(t+1)$.

The above algorithm is an improvement on, and a generalization of, two previous algorithms [25]. Either of the two previous algorithms derives more prefix circuits from only a particular one. Our new algorithm makes the two previous algorithms obsolete.

Theorem 7. *The prefix circuit* $D(t+1)$ *derived by Algorithm* $A(D, n, t, d)$ is $SOPC(2n-1, t+1,\ t+2)$ *with depth* $d+2$.

Proof. Since $D(t)$ is $SOPC(n, t, t+1)$, $ia(D(t)) = t$ and $l(D(t)) = t+1$. That is, the first duplication node on line 1 of $D(t)$ is at level t, and $1{:}n$ is produced at level $t+1$. This, in turn, implies that line n has value $2{:}n$ at level t, and $D(t)$ has an operation node $(1, n)_{t+1}$ to produce $1{:}n$ at level $t+1$.

It can be checked that $DA(t)$ is $SOPC(n, t+1,\ t+2)$ with depth $d+2$. In addition, $DA(t)$ has no operation nodes at level $t+1$, and has exactly one operation node $(1, n)_{t+2}$ at level $t+2$. On the other hand, $DB(t)$ is $SOPC(n,\ t+2,\ t+3)$

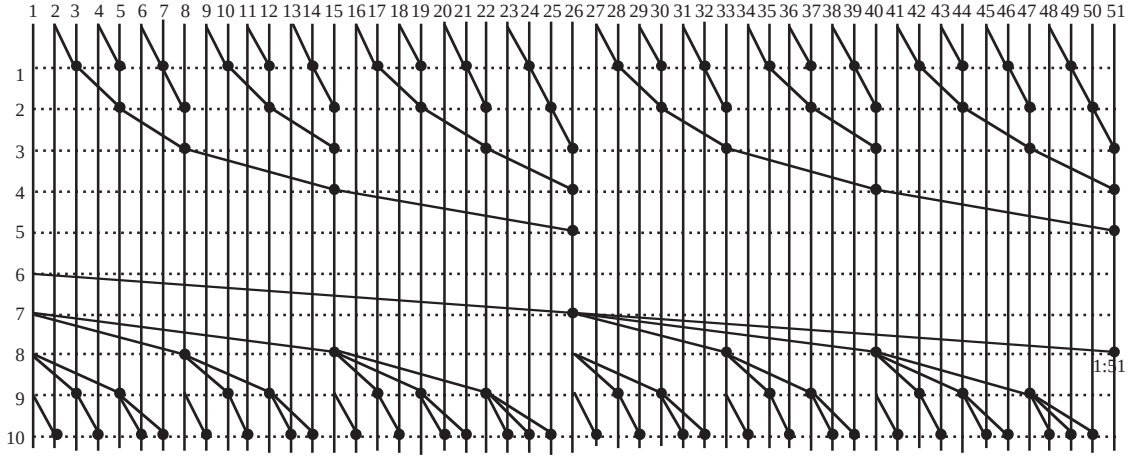

Fig. 9. $YA(5) \bullet YB(5)$.

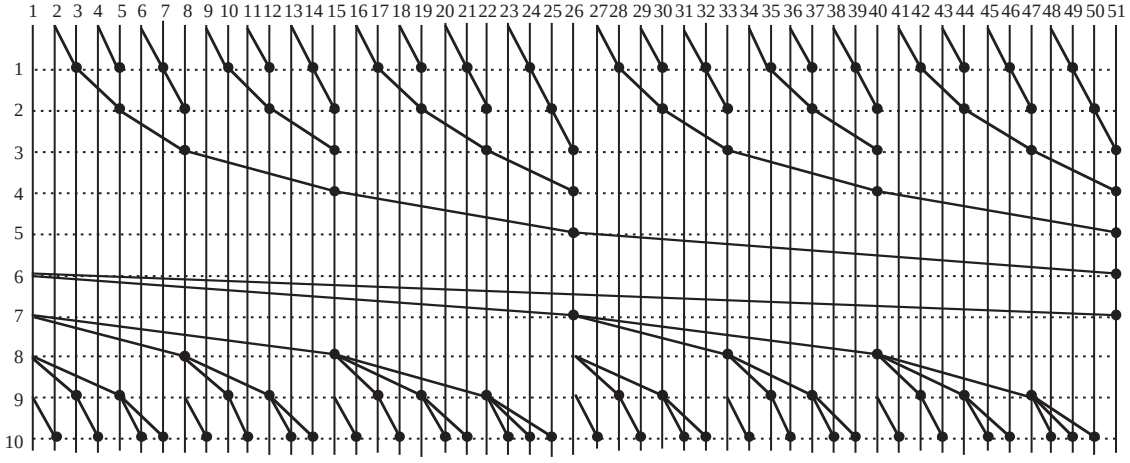

Fig. 10. $SY(6)$.

with depth $d+2$, and $DB(t)$ has no operation nodes at levels $t+1$ and $t+2$.

By Theorem 3, $DA(t) \bullet DB(t)$ is $SOPC(2n-1,\ t+1,\ t+3)$ with depth $d+2$, implying $ia(DA(t) \bullet DB(t)) = t+1$. After deleting node $(n, 2n-1)_{t+3}$ of $DA(t) \bullet DB(t)$ and adding nodes $(n, 2n-1)_{t+1}$ and $(1, 2n-1)_{t+2}$, it is easy to see that $ia(D(t+1)) = t+1$ and $d(D(t+1)) = d+2$. Note that the transformation from $DA(t) \bullet DB(t)$ to $D(t+1)$ does not affect lines i to obtain $1{:}i$, for $1 \leqslant i \leqslant 2n-2$.

In the process of obtaining $DA(t)$ and $DB(t)$, no nodes at level i, $i \leqslant t$, are moved; thus, line n of either $DA(t)$ or $DB(t)$ has value $2{:}n$ at level t. Therefore, $DA(t) \bullet DB(t)$ has $2{:}n$ on line n at level t, and has $n+1{:}2n-1$ on line $2n-1$ at level t. After deleting node $(n, 2n-1)_{t+3}$ and adding node $(n, 2n-1)_{t+1}$, there is $2{:}2n-1$ on line $2n-1$ at level $t+1$. After adding node $(1, 2n-1)_{t+2}$, $D(t+1)$ has $1{:}2n-1$ on line $2n-1$ at level $t+2$. Therefore, $l(D(t+1)) = t+2$ and $D(t+1)$ is a prefix circuit.

Since $DA(t) \bullet DB(t)$ is $SOPC(2n-1,\ t+1,\ t+3)$, we have $s(DA(t) \bullet DB(t)) = 2 \times (2n-1) - 2 + (t+1) - (t+3)$; thus,

$$\begin{aligned} s(D(t+1)) &= s(DA(t) \bullet DB(t)) - 1 + 2 \\ &= 2 \times (2n-1) - 2 + (t+1) - (t+2). \end{aligned}$$

Therefore, by Definition 1, $D(t+1)$ is $SOPC(2n-1,\ t+1,\ t+2)$. □

By Theorem 7, Algorithm A can be used repeatedly to obtain $D(t+c)$, for $c > 1$. For example, we can use Algorithm A(SY, 26, 5, 8) as the first application of Algorithm A to obtain $SY(t)$, for $t \geqslant 6$.

Lemma 8. *Given a prefix circuit $D(t)$ that is $SOPC(n, t, t+1)$ with depth d and fan-out a, where $t+1 < d$ and $a \geqslant 4$. If $D(t)$ has at least two nodes with fan-out a, then the circuit $D(t+1)$ obtained by using Algorithm* A(D, n, t, d) *has a fan-out of a, and $D(t+1)$ has at least two nodes with fan-out a.*

Proof. Since $D(t)$ has at least two nodes with fan-out a, $D(t)$ has a node with fan-out a that is not on line 1 at level t. Thus, the fan-out of both $DA(t)$ and $DB(t)$ is a. $DA(t)$ has at least one node with fan-out a, and $DB(t)$ has at least two nodes with fan-out a. Therefore, $DA(t) \bullet DB(t)$ has at least three nodes with fan-out a. After deleting $(n, 2n-1)_{t+3}$, the fan-out of the circuit is still a, and the circuit now has at least two nodes with fan-out a. Adding two operation nodes does not increase the fan-out of the circuit; therefore, the fan-out of $D(t+1)$ is a, and $D(t+1)$ has at least two nodes with fan-out a. □

Lemma 9. *$SY(t)$ is $SOPC(25 \times 2^{t-5}+1,\ t,\ t+1)$ with fan-out 4 and depth $2t-2$, for $t \geqslant 5$.*

Proof. The proof is by induction on t. We have seen that $SY(5)$ is $SOPC(26, 5, 6)$ with depth 8.

Assume that $SY(k)$ is $SOPC(25 \times 2^{k-5}+1,\ k,\ k+1)$ with fan-out 4 and depth $2k-2$. By Theorem 7, $SY(k+1)$ is $SOPC(25 \times 2^{k-4}+1,\ k+1,\ k+2)$ with depth $2k$. In addition, since $SY(5)$ has three nodes with fan-out 4, by Lemma 8, $SY(t)$ has at least two nodes with fan-out 4 and its fan-out is 4 for $t \geqslant 5$. □

3.2. SZ: a size optimal prefix circuit with fan-out 4

Define a size optimal prefix circuit $SZ(6)$ as shown in Fig. 11. It has the following properties: $d(SZ(6)) = 9$, $fo(SZ(6)) = 4$, $ia(SZ(6)) = 6$, $l(SZ(6)) = 7$, and $s(SZ(6)) = 71 = 2 \times 37 - 2 + 6 - 7$. Therefore, by Definition 1, $SZ(6)$ is $SOPC(37, 6, 7)$. By Theorem 7, we can use Algorithm A to derive $SZ(t)$, for $t \geqslant 7$, from $SZ(6)$.

Lemma 10. *$SZ(t)$ is $SOPC(9 \times 2^{t-4}+1,\ t,\ t+1)$ with fan-out 4 and depth $2t-3$, for $t \geqslant 6$.*

Proof. The proof is by induction on t. We have seen that $SZ(6)$ is $SOPC(37, 6, 7)$ with depth 9.

Assume that $SZ(k)$ is $SOPC(9 \times 2^{k-4}+1,\ k,\ k+1)$ with fan-out 4 and depth $2k-3$. By Theorem 7, $SZ(k+1)$ is $SOPC(9 \times 2^{k-3}+1,\ k+1,\ k+2)$ with depth $2k-1$. In addition, since $SZ(6)$ has seven nodes with fan-out 4, by Lemma 8, $SZ(t)$ has at least two nodes with fan-out 4 and its fan-out is 4 for $t \geqslant 6$. □

3.3. Y: a size optimal prefix circuit with fan-out 4

Define a size optimal prefix circuit Y as shown in Fig. 12. Clearly, $d(Y) = 7$, $fo(Y) = 4$, $ia(Y) = 5$, $l(Y) = 7$, and $s(Y) = 28 = 2 \times 16 - 2 + 5 - 7$. Therefore, by Definition 1, we have the following lemma.

Lemma 11. *Prefix circuit Y is $SOPC(16, 5, 7)$ with fan-out 4 and depth 7.*

3.4. GV and GW: size optimal prefix circuits with fan-out 4

Let $t \geqslant 6$, we define two prefix circuits as follows:

$$GV(t) = SZ(t) \bullet SZ(t-1) \bullet \cdots \bullet SZ(6) \bullet Y,$$
$$GW(t) = SY(t) \bullet GV(t).$$

Lemma 12. *$GV(t)$ is $SOPC(9 \times 2^{t-3}-20,\ t,\ 2t-3)$ with depth $2t-3$ and fan-out 4, for $t \geqslant 6$.*

Proof. By Lemma 10, $SZ(t)$ is $SOPC(9 \times 2^{t-4}+1,\ t,\ t+1)$ with fan-out 4 and depth $2t-3$, for $t \geqslant 6$. By Theorem 3, $SZ(t) \bullet SZ(t-1)$ is $SOPC(9 \times 2^{t-4}+9 \times 2^{t-5}+1,\ t,\ t+2)$ with depth $2t-3$. Using Theorem 3 repeatedly, we can obtain that $SZ(t) \bullet SZ(t-1) \bullet \cdots \bullet SZ(6)$ is $SOPC(9 \times 2^{t-3}-35,\ t,\ 2t-5)$ with depth $2t-3$. Also by Theorem 3, $GV(t) = SZ(t) \bullet SZ(t-1) \bullet \cdots \bullet SZ(6) \bullet Y$ is $SOPC(9 \times 2^{t-3}-20,\ t,\ 2t-3)$ with depth $2t-3$.

Since the fan-out of SZ and Y is 4, by Theorem 3, the fan-out of GV is also 4. □

Lemma 13. *$GW(t)$ is $SOPC(61 \times 2^{t-5}-20,\ t,\ 2t-2)$ with depth $2t-2$ and fan-out 4, for $t \geqslant 6$.*

Proof. By Lemma 9, $SY(t)$ is $SOPC(25 \times 2^{t-5}+1,\ t,\ t+1)$ with fan-out 4 and depth $2t-2$. By Lemma 12, $GV(t)$ is $SOPC(9 \times 2^{t-3}-20,\ t,\ 2t-3)$ with depth $2t-3$ and fan-out 4. Using Theorem 3, we know that $GW(t) = SY(t) \bullet GV(t)$ is $SOPC(61 \times 2^{t-5}-20,\ t,\ 2t-2)$ with depth $2t-2$. Since the fan-out of $SY(t)$ and $GV(t)$ is 4, that of $GW(t)$ is also 4. □

4. Optimal prefix circuit *SU4* with fan-out 4

For $n \geqslant 30$, parallel prefix circuit $SU4(n)$ is defined as follows:

1. When $30 \leqslant n \leqslant 32$ or $n = 47$,
$$SU4(n) = Q(n-16) \bullet S(2) \bullet Y.$$
2. When $33 \leqslant n \leqslant 46$ or $48 \leqslant n \leqslant 62$,
$$SU4(n) = Q(n-15) \bullet Y.$$
3. When $63 \leqslant n \leqslant 72$,
$$SU4(n) = Q(n-40) \bullet SY(5) \bullet Y.$$
4. When $73 \leqslant n \leqslant 84$ or $n = 115$,
$$SU4(n) = Q(n-52) \bullet S(2) \bullet GV(6).$$
5. When $85 \leqslant n \leqslant 114$ or $116 \leqslant n \leqslant 146$,
$$SU4(n) = Q(n-51) \bullet GV(6).$$
6. When $147 \leqslant n \leqslant 165$,
$$SU4(n) = Q(n-101) \bullet GW(6).$$

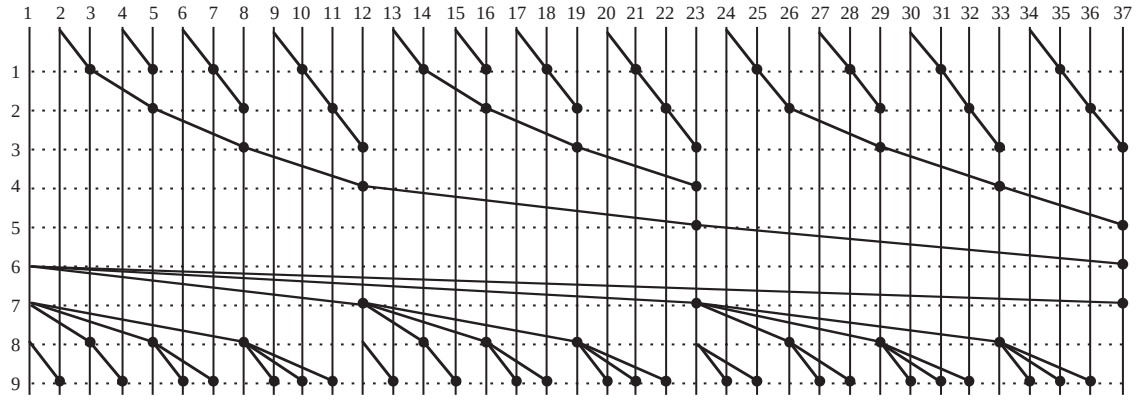

Fig. 11. $SZ(6)$.

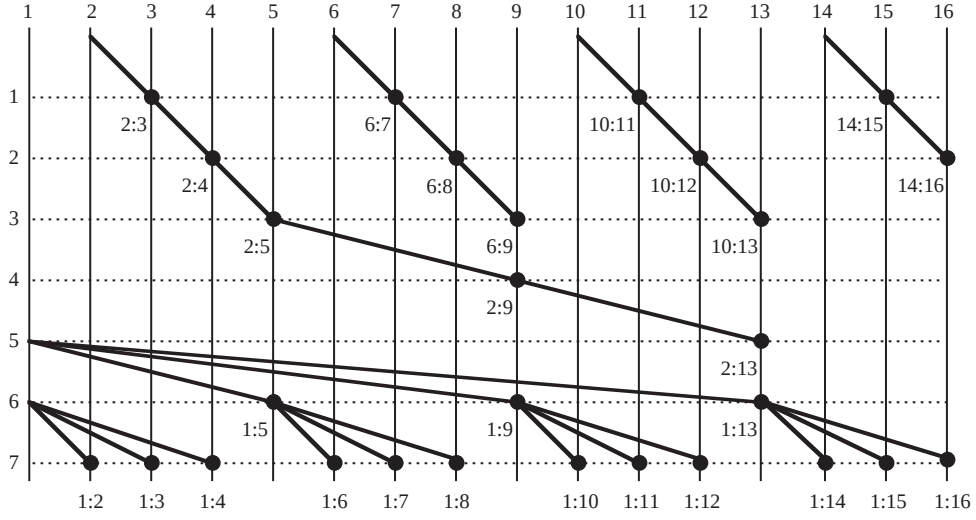

Fig. 12. Prefix circuit Y.

7. When $166 \leqslant n \leqslant 188$ or $n = 251$,

$$SU4(n) = Q(n-124) \bullet S(2) \bullet GV(7).$$

8. When $189 \leqslant n \leqslant 250$ or $252 \leqslant n \leqslant 314$,

$$SU4(n) = Q(n-123) \bullet GV(7).$$

9. When $315 \leqslant n \leqslant 351$,

$$SU4(n) = Q(n-223) \bullet GW(7).$$

10. When $93 \times 2^{t-7} - 20 \leqslant n \leqslant 13 \times 2^{t-4} - 20$ and $t \geqslant 9$,

$$SU4(n) = Q(n - 9 \times 2^{t-4} + 20) \bullet S(2) \bullet GV(t-1).$$

11. When $13 \times 2^{t-4} - 19 \leqslant n \leqslant 2^t$ and $t \geqslant 9$,

$$SU4(n) = Q(n - 9 \times 2^{t-4} + 21) \bullet GV(t-1).$$

12. When $2^{t-1}+1 \leqslant n \leqslant 21 \times 2^{t-5} - 22$, $n \neq 17 \times 2^{t-5} - 21$ and $t \geqslant 10$,

$$SU4(n) = Q(n - 9 \times 2^{t-5} + 21) \bullet GV(t-2).$$

13. When $n = 17 \times 2^{t-5} - 21$ and $t \geqslant 10$,

$$SU4(n) = Q(2^{t-2} - 1) \bullet S(2) \bullet GV(t-2).$$

14. When $21 \times 2^{t-5} - 21 \leqslant n \leqslant 93 \times 2^{t-7} - 21$ and $t \geqslant 10$,

$$SU4(n) = Q(n - 61 \times 2^{t-7} + 21) \bullet GW(t-2).$$

Note that, though not explicitly specified, $t = \lceil \lg n \rceil$. Fig. 13, as an example, shows $SU4(30) = Q(14) \bullet S(2) \bullet Y$.

Theorem 14. *$SU4(n)$ is an optimal prefix circuit with fan-out 4, and its depth is*

$2\lceil \lg n \rceil - 3$ *when* $30 \leqslant n \leqslant 32$;
$2\lceil \lg n \rceil - 4$ *when* $47 \leqslant n \leqslant 64$, $115 \leqslant n \leqslant 128$, *or* $251 \leqslant n \leqslant 256$;
$2\lceil \lg n \rceil - 5$ *when* $33 \leqslant n \leqslant 46$, $73 \leqslant n \leqslant 114$, $166 \leqslant n \leqslant 250$, *or* $93 \times 2^{t-7} - 20 \leqslant n \leqslant 2^t$ *and* $t \geqslant 9$;

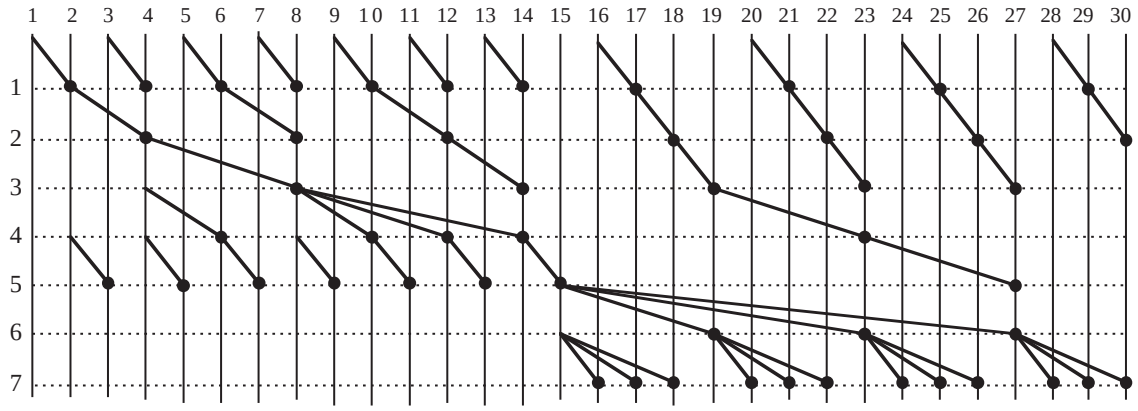

Fig. 13. *SU4*(30).

$2\lceil \lg n \rceil - 6$ *when* $65 \leqslant n \leqslant 72$, $129 \leqslant n \leqslant 165$, $257 \leqslant n \leqslant 351$, *or* $17 \times 2^{t-5} - 21 \leqslant n \leqslant 93 \times 2^{t-7} - 21$ *and* $t \geqslant 10$;

$2\lceil \lg n \rceil - 7$ *when* $2^{t-1} + 1 \leqslant n \leqslant 17 \times 2^{t-5} - 22$ *and* $t \geqslant 10$.

Proof. The fan-out of Q is at most 4, and the fan-out of S is 2. In addition, the fan-out of *SY*, *Y*, *GV*, and *GW* is 4. Since *SU4* is composed of these prefix circuits, by Theorem 3, the fan-out of *SU4* is 4.

We let $t = \lceil \lg n \rceil$ and distinguish the following 24 cases to obtain the depth of *SU4*. Because the proving processes for these cases each are very similar, we will give details for the first three cases only and give tips for the others. More details can be found in [38].

Case 1: When $30 \leqslant n \leqslant 32$, let $m = n - 16$; thus, $14 \leqslant m \leqslant 16$. By Theorem 5, we have $5 \leqslant d(Q(m)) \leqslant 6$. By Theorem 6, $Q(m)$ is $SOPC(m, 0, 4)$, and by Theorem 4, $S(2)$ is $SOPC(2, 0, 1)$ with depth 1. Thus, by Theorem 3, $Q(m) \bullet S(2)$ is $SOPC(m + 1, 0, 5)$ with depth 5 or 6. By Lemma 11, Y is $SOPC(16, 5, 7)$ with depth 7. By Theorem 3, $SU4(n) = Q(m) \bullet S(2) \bullet Y$ is $SOPC(n, 0, 7)$ with depth $7 = 2\lceil \lg n \rceil - 3$. Therefore, by Theorem 2, $SU4(n)$ is optimal.

Case 2: When $33 \leqslant n \leqslant 46$, let $m = n - 15$; thus, $18 \leqslant m \leqslant 31$. By Theorem 5, we have $6 \leqslant d(Q(m)) \leqslant 7$. By Theorem 6, $Q(m)$ is $SOPC(m, 0, 5)$. Since Y is $SOPC(16, 5, 7)$ with depth 7, by Theorem 3, $SU4(n) = Q(m) \bullet Y$ is $SOPC(n, 0, 7)$ with depth $7 = 2\lceil \lg n \rceil - 5$. Therefore, by Theorem 2, $SU4(n)$ is optimal.

Case 3: When $n = 47$, let $m = 31$. By Theorem 5, $d(Q(31)) = 7$. By Theorem 6, $Q(31)$ is $SOPC(31, 0, 5)$. Since $S(2)$ is $SOPC(2, 0, 1)$ with depth 1, by Theorem 3, $Q(31) \bullet S(2)$ is $SOPC(32, 0, 6)$ with depth 7. Since Y is $SOPC(16, 5, 7)$ with depth 7, by Theorem 3, $SU4(n) = Q(31) \bullet S(2) \bullet Y$ is $SOPC(47, 0, 8)$ with depth $8 = 2\lceil \lg n \rceil - 4$. Therefore, by Theorem 2, $SU4(n)$ is optimal.

Case 4: When $48 \leqslant n \leqslant 62$, let $m = n - 15$; thus, $33 \leqslant m \leqslant 47$.

Case 5: When $63 \leqslant n \leqslant 64$, let $m = n - 40$; thus, $23 \leqslant m \leqslant 24$.

Case 6: When $65 \leqslant n \leqslant 72$, let $m = n - 40$; thus, $25 \leqslant m \leqslant 32$.

Case 7: When $73 \leqslant n \leqslant 84$, let $m = n - 52$; thus, $21 \leqslant m \leqslant 32$.

Case 8: When $85 \leqslant n \leqslant 114$, let $m = n - 51$; thus, $34 \leqslant m \leqslant 63$.

Case 9: When $n = 115$, let $m = 63$.

Case 10: When $116 \leqslant n \leqslant 128$, let $m = n - 51$; thus, $65 \leqslant m \leqslant 77$.

Case 11: When $129 \leqslant n \leqslant 146$, let $m = n - 51$; thus, $78 \leqslant m \leqslant 95$.

Case 12: When $147 \leqslant n \leqslant 165$, let $m = n - 101$; thus, $46 \leqslant m \leqslant 64$.

Case 13: When $166 \leqslant n \leqslant 188$, let $m = n - 124$; thus, $42 \leqslant m \leqslant 64$.

Case 14: When $189 \leqslant n \leqslant 250$, let $m = n - 123$; thus, $66 \leqslant m \leqslant 127$.

Case 15: When $n = 251$, let $m = 127$.

Case 16: When $252 \leqslant n \leqslant 256$, let $m = n - 123$; thus, $129 \leqslant m \leqslant 133$.

Case 17: When $257 \leqslant n \leqslant 314$, let $m = n - 123$; thus, $134 \leqslant m \leqslant 191$.

Case 18: When $315 \leqslant n \leqslant 351$, let $m = n - 223$; thus, $92 \leqslant m \leqslant 128$.

Case 19: When $93 \times 2^{t-7} - 20 \leqslant n \leqslant 13 \times 2^{t-4} - 20$ and $t \geqslant 9$, let $m = n - 9 \times 2^{t-4} + 20$; thus, $21 \times 2^{t-7} \leqslant m \leqslant 2^{t-2}$ and $\lceil \lg m \rceil = t - 2$.

Case 20: When $13 \times 2^{t-4} - 19 \leqslant n \leqslant 2^t$ and $t \geqslant 9$, let $m = n - 9 \times 2^{t-4} + 21$; thus, $2^{t-2} + 2 \leqslant m \leqslant 7 \times 2^{t-4} + 21$ and $\lceil \lg m \rceil = t - 1$.

Case 21: When $2^{t-1} + 1 \leqslant n \leqslant 17 \times 2^{t-5} - 22$ and $t \geqslant 10$, let $m = n - 9 \times 2^{t-5} + 21$; thus, $7 \times 2^{t-5} + 22 \leqslant m < 2^{t-2}$ and $\lceil \lg m \rceil = t - 2$.

Case 22: When $n = 17 \times 2^{t-5} - 21$ and $t \geqslant 10$, let $m = 2^{t-2} - 1$.

Case 23: When $17\times 2^{t-5}-20\leqslant n\leqslant 21\times 2^{t-5}-22$ and $t\geqslant 10$, let $m=n-9\times 2^{t-5}+21$; thus, $2^{t-2}<m<3\times 2^{t-3}$ and $\lceil \lg m\rceil = t-1$.

Case 24: When $21\times 2^{t-5}-21\leqslant n\leqslant 93\times 2^{t-7}-21$ and $t\geqslant 10$, let $m=n-61\times 2^{t-7}+21$; thus, $23\times 2^{t-7}\leqslant m\leqslant 2^{t-2}$ and $\lceil \lg m\rceil = t-2$.

By Case 1, when $30\leqslant n\leqslant 32$, $d(SU4)=2\lceil \lg n\rceil -3$. By Cases 3–5, when $47\leqslant n\leqslant 64$, $d(SU4)=2\lceil \lg n\rceil -4$. By Cases 9 and 10, when $115\leqslant n\leqslant 128$, $d(SU4)=2\lceil \lg n\rceil -4$. By Cases 15 and 16, when $251\leqslant n\leqslant 256$, $d(SU4)=2\lceil \lg n\rceil -4$. By Cases 2, 7, and 8, when $33\leqslant n\leqslant 46$ or $73\leqslant n\leqslant 114$, $d(SU4)=2\lceil \lg n\rceil -5$. By Cases 13 and 14, when $166\leqslant n\leqslant 250$, $d(SU4)=2\lceil \lg n\rceil -5$. By Cases 19 and 20, when $93\times 2^{t-7}-20\leqslant n\leqslant 2^t$ and $t\geqslant 9$, $d(SU4)=2\lceil \lg n\rceil -5$. By Cases 6, 11, and 12, when $65\leqslant n\leqslant 72$, $129\leqslant n\leqslant 165$, $d(SU4)=2\lceil \lg n\rceil -6$. By Cases 17 and 18, when $257\leqslant n\leqslant 351$, $d(SU4)=2\lceil \lg n\rceil -6$. By Cases 22–24, when $17\times 2^{t-5}-21\leqslant n\leqslant 93\times 2^{t-7}-21$ and $t\geqslant 10$, $d(SU4)=2\lceil \lg n\rceil -6$. By Case 21, when $2^{t-1}+1\leqslant n\leqslant 17\times 2^{t-5}-22$ and $t\geqslant 10$, $d(SU4)=2\lceil \lg n\rceil -7$. □

Note that the definition of *SU4* can be compacted as follows:

1. When $30\leqslant n\leqslant 32$ or $n=47$,
$$SU4(n)=Q(n-16)\bullet S(2)\bullet Y.$$
2. When $33\leqslant n\leqslant 46$ or $48\leqslant n\leqslant 62$,
$$SU4(n)=Q(n-15)\bullet Y.$$
3. When $63\leqslant n\leqslant 72$,
$$SU4(n)=Q(n-40)\bullet SY(5)\bullet Y.$$
4. When $93\times 2^{z-3}-20\leqslant n\leqslant 13\times 2^z-20$ or $n=17\times 2^z-21$ and $z\geqslant 3$,
$$SU4(n)=Q(n-9\times 2^z+20)\bullet S(2)\bullet GV(z+3).$$
5. When $13\times 2^z-19\leqslant n\leqslant 21\times 2^z-22$, $n\neq 17\times 2^z-21$ and $z\geqslant 3$,
$$SU4(n)=Q(n-9\times 2^z+21)\bullet GV(z+3).$$
6. When $21\times 2^z-21\leqslant n\leqslant 93\times 2^{z-2}-21$ and $z\geqslant 3$,
$$SU4(n)=Q(n-61\times 2^{z-2}+21)\bullet GW(z+3).$$

5. Comparisons of optimal prefix circuits

In this section, *SU4* is compared with two optimal prefix circuits, *WE4* [25] and *Z4* [24]. They all have the same fan-out 4. The depth of either *WE4* or *Z4* is less than or equal to those of all the other previous optimal prefix circuits with the same fan-out. The depths of *WE4* and *Z4* are between $2\lceil \lg n\rceil -6$ and $2\lceil \lg n\rceil -3$. For easy and exact comparisons, Table 1 gives the numbers of inputs that can be processed by the three parallel prefix circuits with specific depths.

Table 1
The numbers of inputs that three representative optimal parallel prefix circuits with specific depths can process

Depth	*SU4*	*WE4*	*Z4*
7	30–46	29–46	30–44
8	47–72	47–64	45–67
9	73–114	65–102	68–98
10	115–165	103–138	99–145
11	166–250	139–214	146–208
12	251–351	215–286	209–303
13	352–522	287–438	304–430
14	523–723	439–582	431–621
15	724–1066	583–886	622–876
16	1067–1467	887–1174	877–1259
17	1468–2154	1175–1782	1260–1770
18	2155–2955	1783–2358	1771–2537
19	2956–4330	2359–3574	2538–3560
20	4331–5931	3575–4726	3561–5095

From Table 1, we see that, when $30\leqslant n\leqslant 5931$, $d(WE4)-1\leqslant d(SU4)\leqslant d(WE4)$ and $d(Z4)-1\leqslant d(SU4)\leqslant d(Z4)$. In fact, we can show analytically that the above relations do hold when $n\geqslant 513$ as follows.

From [25], when $n\geqslant 513$, $d(WE4)$ is

$2\lceil \lg n\rceil -4$ when $7\times 2^{t-3}-9\leqslant n\leqslant 2^t$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -5$ when $37\times 2^{t-6}-9\leqslant n\leqslant 7\times 2^{t-3}-10$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -6$ when $2^{t-1}<n\leqslant 37\times 2^{t-6}-10$ and $t\geqslant 10$.

From Theorem 14, when $n\geqslant 513$, $d(SU4)$ is

$2\lceil \lg n\rceil -5$ when $93\times 2^{t-7}-20\leqslant n\leqslant 2^t$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -6$ when $17\times 2^{t-5}-21\leqslant n\leqslant 93\times 2^{t-7}-21$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -7$ when $2^{t-1}<n\leqslant 17\times 2^{t-5}-22$ and $t\geqslant 10$.

Therefore, it can be checked that when $n\geqslant 513$, $d(WE4)-1\leqslant d(SU4)\leqslant d(WE4)$ [38].

From [24], when $n\geqslant 513$, $d(Z4)$ is

$2\lceil \lg n\rceil -4$ when $7\times 2^{t-3}-2t<n\leqslant 2^t$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -5$ when $5\times 2^{t-3}-2t+2\leqslant n\leqslant 7\times 2^{t-3}-2t$ and $t\geqslant 10$;
$2\lceil \lg n\rceil -6$ when $2^{t-1}<n\leqslant 5\times 2^{t-3}-2t+1$ and $t\geqslant 10$.

It can then be checked that when $n\geqslant 513$, $d(Z4)-1\leqslant d(SU4)\leqslant d(Z4)$ [38]. Together with Table 1, therefore, $d(WE4)-1\leqslant d(SU4)\leqslant d(WE4)$ and $d(Z4)-1\leqslant d(SU4)\leqslant d(Z4)$, for $n\geqslant 30$.

6. Conclusion

In this paper, we have presented Algorithm A, which can construct an infinite number of size optimal prefix circuits from a size optimal prefix circuit with some simple properties. Thereby, *SY* and *SZ* are obtained; together with other prefix circuits, they are used to compose *SU4*, an optimal

parallel prefix circuit with fan-out 4. The algorithm and our approach may contribute to the design of more novel prefix circuits with any fixed fan-out. *SU4* makes obsolete all the other optimal parallel prefix circuits with fan-out 4 for $n \geqslant 30$, because its depth is the smallest of all the depths of known optimal parallel prefix circuits with the same fan-out.

A question arises naturally. Can we systematically construct new optimal parallel prefix circuits with fan-out 4 and smaller depths? From our experience, it is very probable when n is very large, but there is little or no hope for a small n. Our approach serves a basis for achieving this when n is huge. Specifically, we will need to devise new size optimal prefix circuits to replace *SY*, *SZ*, and *Y*; these new size optimal prefix circuits will take many more inputs than their counterparts, respectively. All the other procedures will be similar, including the application of Algorithm A. Without the insight gained from our approach, it would be difficult to have another heuristics that can lead to an optimal prefix circuit with fan-out 4 and a smaller depth for any n that is huge.

A final note: this research is mainly a theoretical study. In spite of the fact that, for standard-cell implementations, the depth and the size of a prefix circuit are in general closely related to the cell delay and cell area, respectively [45], the assumption of relationships between, for example, size and VLSI area as well as depth and computation time, may not be correct for certain implementations.

Acknowledgments

This research was supported in part by the National Science Council of the R.O.C. under contract NSC90-2213-E-011-065.

References

[1] S.G. Akl, Parallel Computation: Models and Methods, Prentice-Hall, Upper Saddle River, NJ, 1997.

[2] S. Aluru, N. Futamura, K. Mehrotra, Parallel biological sequence comparison using prefix computaions, J. Parallel Distrib. Comput. 63 (2003) 264–272.

[3] A. Bilgory, D.D. Gajski, A heuristic for suffix solutions, IEEE Trans. Comput. C-35 (1986) 34–42.

[4] G.E. Blelloch, Scans as primitive operations, IEEE Trans. Comput. 38 (1989) 1526–1538.

[5] R.P. Brent, H.T. Kung, A regular layout for parallel adders, IEEE Trans. Comput. C-31 (1982) 260–264.

[6] D.A. Carlson, B. Sugla, Limited width parallel prefix circuits, J. Supercomput. 4 (1990) 107–129.

[7] L. Cinque, G. Bongiovanni, Parallel prefix computation on a pyramid computer, Pattern Recognition Lett. 16 (1995) 19–22.

[8] R. Cole, U. Vishkin, Faster optimal parallel prefix sums and list ranking, Infom. Control 81 (1989) 334–352.

[9] A. Datta, Multiple addition and prefix sum on a linear array with a reconfigurable pipelined bus system, J. Supercomput. 29 (2004) 303–317.

[10] C. Efstathiou, H.T. Vergos, D. Nikolos, Fast parallel-prefix modulo $2^n + 1$ adders, IEEE Trans. Comput. 53 (2004) 1211–1216.

[11] A. Ferreira, S. Ubeda, Parallel complexity of the medial axis computation, in: Proceedings of International Conference on Image Processing, vol. 2, Washington, DC, 1995, pp. 105–108.

[12] F.E. Fich, New bounds for parallel prefix circuits, in: Proceedings of the 15th Symposium on the Theory of Computing, 1983, pp. 100–109.

[13] W. Gropp, E. Lusk, A. Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, MIT Press, Cambridge, MA, 1994.

[14] T. Han, D.A. Carlson, Fast area-efficient VLSI adders, in: Proceedings of the Eighth Computer Arithmetic Symposium, Como, Italy, 1987, pp. 49–56.

[15] D.R. Helman, J. JaJa, Prefix computations on symmetric multiprocessors, J. Parallel Distrib. Comput. 61 (2001) 265–278.

[16] P.M. Kogge, H.S. Stone, A parallel algorithm for the efficient solution of a general class of recurrence equations, IEEE Trans. Comput. C-22 (1973) 783–791.

[17] C.P. Kruskal, L. Rudolph, M. Snir, The power of parallel prefix, IEEE Trans. Comput. C-34 (1985) 965–968.

[18] R.E. Ladner, M.J. Fischer, Parallel prefix computation, J. ACM 27 (1980) 831–838.

[19] S. Lakshmivarahan, S.K. Dhall, Parallel Computing Using the Prefix Problem, Oxford University Press, Oxford, UK, 1994.

[20] S. Lakshmivarahan, C.M. Yang, S.K. Dhall, On a new class of optimal parallel prefix circuits with $(\text{size} + \text{depth}) = 2n - 2$ and $\lceil \log n \rceil \leqslant \text{depth} \leqslant (2\lceil \log n \rceil - 3)$, in: Proceedings of International Conference on Parallel Processing, St. Charles, IL, 1987, pp. 58–65.

[21] F.T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann, San Mateo, CA, 1992.

[22] R. Lin, K. Nakano, S. Olariu, M.C. Pinotti, J.L. Schwing, A.Y. Zomaya, Scalable hardware-algorithms for binary prefix sums, IEEE Trans. Parallel Distrib. Systems 11 (2000) 838–850.

[23] Y.-C. Lin, Optimal parallel prefix circuits with fan-out 2 and corresponding parallel algorithms, Neural Parallel Scientific Comput. 7 (1999) 33–42.

[24] Y.-C. Lin, J.-N. Chen, Z4: A new depth-size optimal parallel prefix circuit with small depth, Neural Parallel Scientific Comput. 11 (2003) 221–235.

[25] Y.-C. Lin, J.-W. Hsiao, A new approach to constructing optimal parallel prefix circuits with small depth, J. Parallel Distrib. Comput. 64 (2004) 97–107.

[26] Y.-C. Lin, Y.-H. Hsu, C.-K. Liu, Constructing H4, a fast depth-size optimal parallel prefix circuit, J. Supercomput. 24 (2003) 279–304.

[27] Y.-C. Lin, C.M. Lin, Efficient parallel prefix algorithms on multicomputers, J. Inform. Sci. Eng. 16 (2000) 41–64.

[28] Y.-C. Lin, C.-K. Liu, Finding optimal parallel prefix circuits with fan-out 2 in constant time, Inform. Process. Lett. 70 (1999) 191–195.

[29] Y.-C. Lin, C.-C. Shih, Optimal parallel prefix circuits with fan-out at most 4, in: Proceedings of the Second IASTED International Conference on Parallel and Distributed Computing and Networks, Brisbane, Australia, 1998, pp. 312–317.

[30] Y.-C. Lin, C.-C. Shih, A new class of depth-size optimal parallel prefix circuits, J. Supercomput. 14 (1999) 39–52.

[31] Y.-C. Lin, C.-S. Yeh, Efficient parallel prefix algorithms on multiport message-passing systems, Inform. Process. Lett. 71 (1999) 91–95.

[32] Y.-C. Lin, C.-S. Yeh, Optimal parallel prefix on the postal model, J. Inform. Sci. Eng. 19 (2003) 75–83.

[33] J. Liu, S. Zhou, H. Zhu, C.-K. Cheng, An algorithmic approach for generic parallel adders, in: Proceedings of ICCAD, San Jose, CA, 2003, pp. 734–740.

[34] R. Manohar, J.A. Tierno, Asynchronous parallel prefix computation, IEEE Trans. Comput. 47 (1998) 1244–1252.

[35] Y. Pan, S.Q. Zheng, K. Li, H. Shen, An improved generalization of mesh-connected computers with multiple buses, IEEE Trans. Parallel Distrib. Systems 12 (2001) 293–305.

[36] E.E. Santos, Optimal and efficient algorithms for summing and prefix summing on parallel machines, J. Parallel Distrib. Comput. 62 (2002) 517–543.
[37] M. Snir, Depth-size trade-offs for parallel prefix computation, J. Algorithms 7 (1986) 185–201.
[38] C.-Y. Su, Algorithm-based construction of optimal parallel prefix circuits with smaller depth, Master Thesis, Dept. of Electronic Engineering, National Taiwan University of Science and Technology, 2003.
[39] Thinking Machines, Connection Machine Parallel Instruction Set (PARIS), Thinking Machines, Cambridge, MA, 1986.
[40] H. Wang, A. Nicolau, K.S. Siu, The strict time lower bound and optimal schedules for parallel prefix with resource constraints, IEEE Trans. Comput. 45 (1996) 1257–1271.
[41] N.H.E. Weste, K. Eshraghian, Principles of CMOS VLSI Design: A System Perspective, Addison-Wesley, Reading, MA, 1993.
[42] F. Zhou, P. Kornerup, Computing moments by prefix sums, J. VLSI Signal Process. Systems 25 (2000) 5–17.
[43] H. Zhu, C.-K. Cheng, R. Graham, On the construction of zero-deficiency parallel prefix adders, in: Proceedings of the 13th International Workshop on Logic and Synthesis, Temecula, CA, 2004, pp. 280–286.
[44] R. Zimmermann, Non-heuristic optimization and synthesis of parallel-prefix adders, in: Proceedings of International Workshop on Logic and Architecture Synthesis, Grenoble, France, 1996, pp. 123–132.
[45] R. Zimmermann, Binary adder architectures for cell-based VLSI and their synthesis, Ph.D. Thesis, Swiss Federal Institute of Technology (ETH) Zurich, 1997.

Yen-Chun Lin received his B.S. in electrical engineering from National Taiwan University in 1977, M.S. in computer engineering from National Chiao-Tung University in 1983, and Ph.D. in electrical engineering from National Taiwan University in 1988. Since 1988 he has been on the faculty at National Taiwan University of Science and Technology. Dr. Lin has been a full professor since February 1993, in Department of Electronic Engineering up to July 2001 and then in Department of Computer Science and Information Engineering. He served as Program Chair of the 2001 International Conference on Parallel and Distributed Computing, Applications, and Technologies. He was a Visiting Scientist at the IBM Almaden Research Center, San Jose, California, from 1993 to 1994. His research interests include parallel computing and multimedia systems.

Chin-Yu Su received his B.S. in applied mathematics from National Chung-Hsin University, Taichung, Taiwan in 1997. After serving in the military, he received his M.S. in electronic engineering from National Taiwan University of Science and Technology, Taipei, Taiwan in 2003. His research interests include parallel computing.