JOURNAL OF COMPUTER AND SYSTEM SCIENCES 30, 329–342 (1985)

Area–Period Tradeoffs for Multiplication of Rectangular Matrices

FERNG-CHING LIN AND I-CHEN WU

Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, Republic of China

Received February 27, 1984; revised October 9, 1984

A VLSI computation model is presented with a time dimension in which the concept of information transfer is made precise and memory requirements (lower bounds for A) and area–period trade-offs (lower bounds for AP^2) are treated uniformly. By employing the transitivities of cyclic shiftings and binary multiplication it is proved that $AP^{2\alpha} = \Omega((\min(mn, mp, np)\, l)^{1+\alpha})$, $0 \leqslant \alpha \leqslant 1$, for the problem of multiplying $m \times n$ and $n \times p$ matrices of l-bit elements. We also show that $\min(mn, mp, np)\, l$ is the exact bound for chip area.

1. INTRODUCTION

In a VLSI computation, the task is distributed over various processing elements and the interconnecting wires are used for necessary communication. The wires often occupy much space so that area minimization becomes important in chip designs. Recent researches [1, 6, 7, 10, 15–18] indicate that tradeoffs between chip area A and computation time T exist for many problems.

For getting a lower bound for A, one usually considers the memory requirement at a certain stage of the computation. On the other hand, during the computation, some amount of information must be transferred from one part of the chip to the other due to the nature of the problem. If the amount of transferred information, the information content, can be measured from below then a lower bound for AT^2 can be derived. Various techniques for proving lower bounds can be found in the references cited above and also in [3, 19]. Conventional VLSI models for obtaining lower bounds lack handy ways to deal with the transient phenomena that different signals may occupy the same place at different time instances. This complicates the matter by hiding the time dimension in the models.

In this paper, we propose an approach in which the time dimension is brought back into a conventional VLSI model. The concept of information content is made precise and memory requirements and area–time trade-offs are treated uniformly. Furthermore, in pipelined chips where T should be replaced by the computation period P [18], similar results for area–period trade-offs can be derived without any adjustment in our model.

0022-0000/85 $3.00

For problems like cyclic shiftings and binary multiplication, the information contents are easy to measure because of the transitivities they induce. The useful concept of transitivity was raised by Vuillemin [18] and we redefine it in a very straightforward manner. Motivated by the fact that transitivity only occurs within a row or a column in the matrix multiplication problem, we introduce the notion of partitioned transitivity for summing up the information contents.

For the multiplication of $m \times n$ and $n \times p$ Boolean matrices, Savage [16] showed

$$AT^2 = \Omega\left(m^2p^2\left(1 - \frac{(2a-n)(2b-n)}{2ab}\right)^2\right)$$

when $(a-n)(b-n) < n^2/2$, where $a = \max(n, p)$ and $b = \max(n, m)$. Savage set the case when $m, p > n$ and $(m-n)(p-n) \geqslant n^2/2$ as an open question. We answer this question by proving a stronger and more complete result

$$AP^{2\alpha} = \Omega((\min(mn, mp, np)\, l)^{1+\alpha}), \qquad 0 \leqslant \alpha \leqslant 1,$$

where l is the bit length of elements in the input matrices. It puts no constraint on the dimensions of the matrices and goes down to the bit level of the computation. We also show that $\min(mn, mp, np)\, l$ is the exact bound for chip area.

The result we obtain for square matrix multiplication provides indirect proofs of area–period trade-offs for related problems like transitive closure and matrix inversion. We also give a direct lower bound proof for the all-pair shortest-paths problem.

2. The $A \times T$ Grid Model

A variety of VLSI computation models have been proposed [7, 11, 17]; they share many features in common with the grid model which we shall use here. We postulate a rectangular grid in which wires run along the horizontal or vertical grid lines. On any grid line there can be a wire for each layer. Circuit elements, in particular, I/0 pads, contacts, and logic elements reside at the grid points.

Although in different technologies the separation among layers and circuit elements could be different, the spacing between any two parallel grid lines we can view as a constant λ and this does affect the asymptotic complexity analysis. The times and locations at which the input and output bits are available are assumed to be fixed and independent of the input values. In other words, we only consider when and where deterministic chips. We also assume that each input or output bit is available only once, i.e., there is no free memory outside the chip.

Since the time dimension is hiden in such a 2D grid model we like to extend it to a 3D or $A \times T$ grid model. In a 3D coordinate system, a point (x, y, t) stands for the state at the point (x, y) in the 2D grid at time instance t. If we use the clock time τ as the spacing in the time dimension, then $(i\lambda, j\lambda, k\tau)$, where i, j, k are nonnegative integers, represents a grid point in the $A \times T$ grid model. A grid point

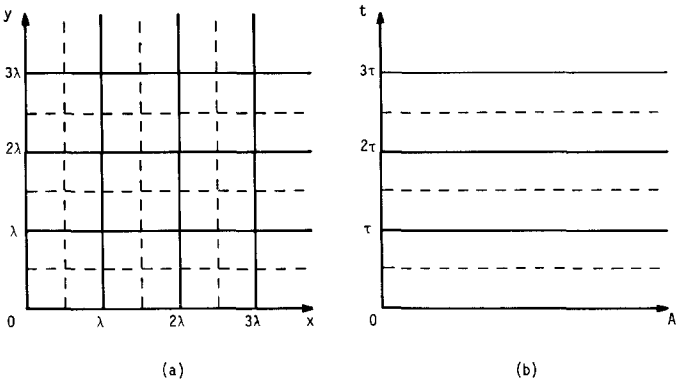

FIG. 1. Subdivision planes.

(x, y, t) will be called an *entrance* if some data bit is input to the point (x, y) at time t. Similarly, a grid point (x, y, t) will be called an *exit* if some data bit is output from the point (x, y) at time t. Note that because any input or output bit is available only once and when and where deterministic, there is an 1–1 correspondence between input (resp. output) bits and entrances (resp. exits).

We devide each $\lambda \times \lambda$ square in A into four $\lambda/2 \times \lambda/2$ subsquares as shown by the dashed lines in Fig. 1a. In the direction of t we also divide each τ interval into two $\tau/2$ subintervals as shown in Fig. 1b. One dashed line stands for a vertical or horizontal plane in the $A \times T$ model which we shall call a *subdivision plane* subsequently.

A *bisection surface* is defined to be a connected surface which partitions the whole grid into two parts and any point of the surface must be on some subdivision plane. We see that no grid point can be on a bisection surface. If certain points, usually part of the exits, are paid close attention then they are called the *observed points.* The main purpose of a bisection surface is to separate the observed points. If a bisection surface is constructed such that it separates the observed points into two equal halves then it is called a *balanced bisection surface.*

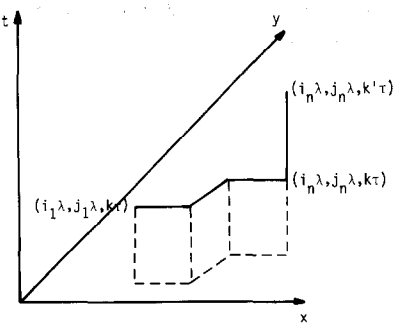

FIG. 2. A typical information transfer.

Suppose, in A, a signal is sent from one point to another by a wire path $(i_1\lambda, j_1\lambda)$,..., $(i_n\lambda, j_n\lambda)$ during the period from $k\tau$ to $k'\tau$, we use, in $A \times T$, the path $(i_1\lambda, j_1\lambda, k\tau)$,..., $(i_n\lambda, j_n\lambda, k\tau)$, $(i_n\lambda, j_n\lambda, k'\tau)$ to represent the *information transfer*. A typical information transfer is shown in Fig. 2. The vertical segment has length $(k'-k)\tau$ which is the propagation delay of the signal. In this way, we can get around the diverse arguments about delay assumption on long wires as discussed in [4, 5, 14, 17].

The number of times information transfers cross a bisection surface is defined to be an *information content* of the computation. It should be apparent that at most a constant number of information transfers can pass through a bisection surface at the same point. Therefore, if the information content is I then we can say that the area of the bisection surface is $\Omega(I)$. Usually we can only bound the information content from below, thereafter we also denote its lower estimation by I.

3. Information Contents and Area–Time Trade-offs

We relate information contents to area lower bounds and area–time trade-offs in the following fundamental theorem.

THEOREM 1. *If the information content of any balanced bisection surface is I then* $AT^{2\alpha} = \Omega(I^{1+\alpha})$ *for* $0 \leqslant \alpha \leqslant 1$.

Proof. When $\alpha = 0, 1$ we have the following two interesting cases:

(a) $A = \Omega(I)$ (lower bound for area)

(b) $AT^2 = \Omega(I^2)$ (area–time trade-off).

It suffices to show (a) and (b) because they readily imply $AT^{2\alpha} = A^{1-\alpha}(AT^2)^{\alpha} = \Omega(I^{1+\alpha})$ for $0 \leqslant \alpha \leqslant 1$. We shall prove (b) and then (a).

Let the width, depth, and height of the grid be w, d, and T, respectively. We first try to construct a balanced bisection surface as perpendicular to x axis as possible and partition the grid points into two parts P_1 and P_2.

(i) Select a subdivision plane perpendicular to x axis by sliding it from left to right. One thing to make sure is that the number of observed points in the left part, i.e., $0 \leqslant x < (i+1/2)\lambda$, does not exceed one half of the number of observed points. Give the left part to P_1.

(ii) If the desired number is not made, go to the plane $x = i+1$ for more observed points. This time we have to move from front to back in the y axis direction and make sure that the number of observed points in the region $[0 \leqslant x < (i+1/2)\lambda]$ v $[0 \leqslant x < (i+3/2)\lambda \wedge 0 \leqslant y < (j+1/2)\lambda]$ is not greater than the desired number. Give the extra region to P_1, the result is shown in Fig. 3a, b.

(iii) If the desired number is still not made then we can cut more observed points to P_1 in the pole $[(i+1/2)\lambda \leqslant x \leqslant (i+3/2)\lambda] \wedge [(j+1/2)\lambda \leqslant y \leqslant (j+3/2)\lambda]$ as shown in Fig. 3c.

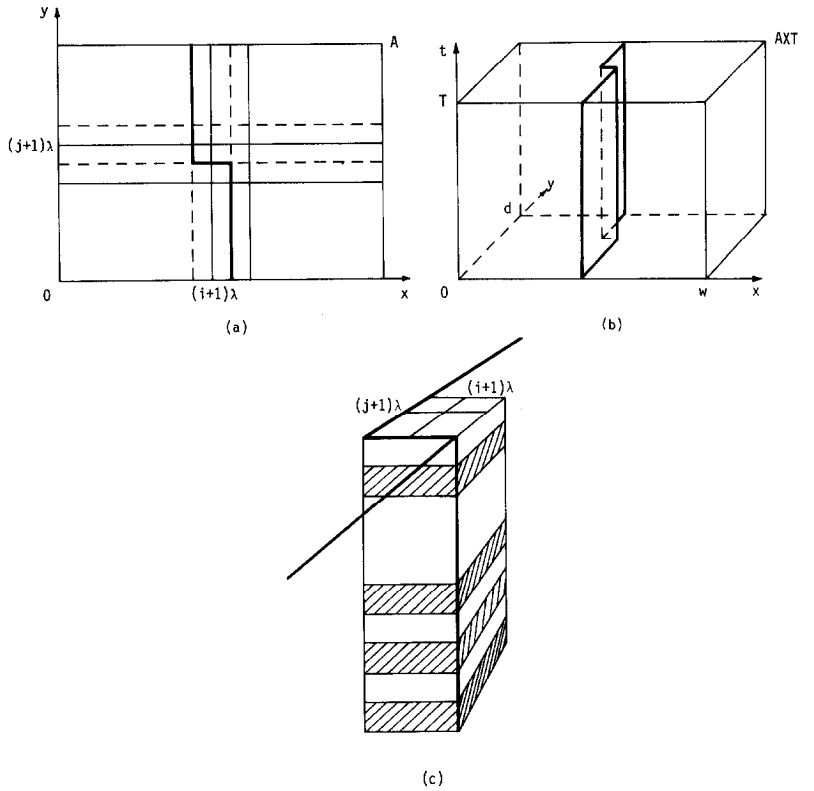

FIG. 3. Constructing a bisection surface.

The area of the bisection surface is clearly no more than $dT + 3T$. This implies $dT = \Omega(I)$. If we construct a balanced bisection surface perpendicular to y axis instead, we have $wT = \Omega(I)$. Put them together, we have $AT^2 = w\,dT^2 = (wT)(dT) = \Omega(I^2)$. This completes the proof of (b). Similar arguments are made for the bisection surface perpendicular to t axis. (See Fig. 4) This time we have $A = wd = \Omega(I)$ or (a). ∎

An important measure for the time complexity of pipelined chips is the computation period P. Vuillemin [18] defined P as the maximal time interval between, two successive data passages at any input or output pad. In this paper we change the definition to the average time interval. Since $P \leqslant T$, any area–period lower bound is automatically an area–time lower bound. The following theorem is a stronger version of Theorem 1.

THEOREM 2. *If the information content of any balanced bisection surface is I then* $AP^{2\alpha} = \Omega(I^{1+\alpha})$ *for* $0 \leqslant \alpha \leqslant 1$.

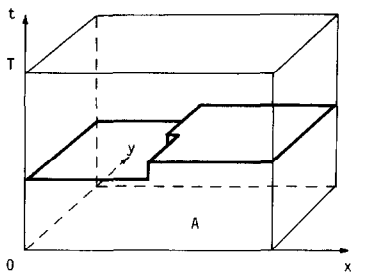

FIG. 4. A bisection surface perpendicular to t-axis.

Proof. For $A = \Omega(I)$ the proof is exactly the same as in Theorem 1, where we only consider one computation. For $dP = \Omega(I)$ and $wP = \Omega(I)$ we have to consider, say m pipelined computations. Let the total computation time be T'. By similar arguements we have $dT' = \Omega(mI)$ and $wT' = \Omega(mI)$. Since $T'/m \to P$ when $m \to \infty$, $dP = \Omega(I)$, and $wP = \Omega(I)$. ∎

4. PARTITIONED TRANSITIVITY

From the previous section we know that information content is an efficacious means to derive area lower bounds and area–period trade-offs. Vuillemin [18] raised the useful concept of transitivity of functions for measuring information contents. Here we want to redefine it in a very natural way. In mathematics, a collection of mappings on a set X is called transitive if, for any pair of elements x_1, x_2 in X, there exists at least one mapping in the collection which maps x_1 into x_2. Now in VLSI computation, for a given problem, if the values of some input variables are properly controlled, one might be able to find out that transitivity does occur between parts of the input and output variables.

DEFINITION 1. Let X be an input variable set and Y be an output variable set. A collection of 1–1 mappings, each of which maps a subset of X into a subset of Y, is said to be *transitive* if, for every x in X and every y in Y, there is at least one mapping in the collection which maps x into y.

EXAMPLE 1 (Cyclic shiftings). We define the collection of cyclic shiftings from $X = \{x_1, x_2, \ldots, x_m\}$ to $Y = \{y_1, y_2, \ldots, y_n\}$ as follows:

(i) If $m \geqslant n$, there are m cyclic left shiftings R_i, $1 \leqslant i \leqslant m$, defined by

$$y_j = R_i(x_{(J+i-1) \bmod m}) \qquad \text{for} \quad 1 \leqslant j \leqslant n.$$

(ii) If $m \leqslant n$, there are n cyclic right shiftings R_i, $1 \leqslant i \leqslant n$, defined by

$$y_{(i+j-1) \bmod n} = R_i(x_j) \qquad \text{for} \quad 1 \leqslant j \leqslant m.$$

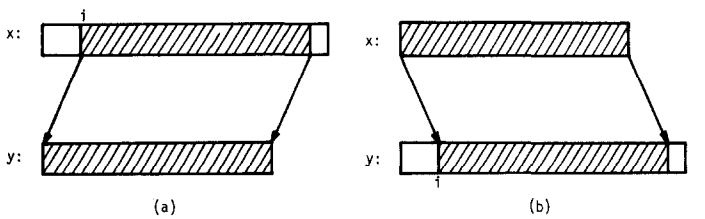

FIG. 5. Cyclic shiftings.

We roughly illustrate the cyclic shifting R_i for cases (i) and (ii) in Figs. 5a and b, respectively. The collection of $\max(m, n)$ cyclic shiftings is certainly transitive.

A transitive collection of mappings can often be induced from a given problem by properly controlling some of its input variables. For example, consider the matrix multiplication problem $C = AB$, where A, B, C are $1 \times m$, $m \times n$,. $1 \times n$ Boolean matrices. We can assign B to be cyclic shift permutation matrices to induce a collection of cyclic shiftings from A to C. Similarly, if A, B, C are $m \times n$, $n \times 1$, $m \times 1$ Boolean matrices, we assign A to be cyclic shift permutation matrices and induce a collection of cyclic shiftings from B to C.

EXAMPLE 2 (Binary multiplication). Write the problem of binary multiplication as $(z_1, z_2, \ldots, z_{2n}) = (x_1, x_2, \ldots, x_n)(y_1, y_2, \ldots, y_n)$. Here we assume n is even. For each i, we can properly fix $y_i = 1$ and $y_j = 0$ for all $j \neq i$. This induces a transitive collection of mappings from $\{x_1, x_2, \ldots, x_{n/2}\}$ to $\{z_{n/2+1}, \ldots, z_n\}$.

LEMMA 1. *For a given problem, suppose there is a set of m input variables and a set of n output variables such that between them a transitive collection of f mappings can be induced. Then the information content of any balanced bisection surface with respect to the n observed exists is at least mn/2f.*

Proof. Consider any balanced bisection surface. Each input variable is mapped into every output variable at least once. This implies that each input variable is mapped into output variables in the other part at least $n/2$ times. Therefore totally these m input variables are mapped into output variables in the other part at least $mn/2$ times. By the pigeon hole principle, there is one mapping which map at least $mn/2f$ input variables into the other part. Hence this particular mapping reveals at least $mn/2f$ information transfers across the bisection surface. ∎

For the special cyclic shiftings problem, $m = n$, Vuillemin [18] has proved the area–period lower bound $AP^2 = \Omega(n^2)$. Brent and Kung [7] proved an area–time lower bound $AT^2 = \Omega(n^2)$ for the binary multiplication problem. Here we apply Lemma 1 and Theorem 2 to obtain stronger results as stated in the following two theorems.

THEOREM 3. *If a problem can induce a collection of cyclic shiftings from m input*

variables to n output variables, then for this problem $AP^{2\alpha} = \Omega((\min(m, n))^{1+\alpha})$ *for* $0 \leqslant \alpha \leqslant 1$.

Proof. As indicated in Example 1, the collection of $\max(m, n)$ cyclic shiftings is transitive. By Lemma 1, the information content of any balanced bisection surface is at least $mn/2 \max(m, n) = \min(m, n)/2$. Applying Theorem 2 we have the result. ∎

THEOREM 4. *For the problem of multiplying two n-bit numbers,* $AP^{2\alpha} = \Omega(n^{1+\alpha})$, *for* $0 \leqslant \alpha \leqslant 1$.

Proof. A transitive collection of n mappings between $n/2$ inputs and $n/2$ outputs can be induced by the problem as indicated in Example 2. By Lemma 1 we know that the information content is at least $(n/2 \cdot n/4)/n = n/8$. Theorem 2 gives the area–period lower bound. ∎

Because the matrix multiplication problem can only induce transitivity inside each row or column but not the whole matrix, we have to generalize the concept of transitivity to partitioned transitivity. We also can extend Lemma 1 naturally.

DEFINITION 2. Let X be an input variable set and Y be an output variable set. X is partitioned into disjoint subsets X_i, $1 \leqslant i \leqslant p$, and Y is partitioned into disjoint subsets Y_i, $1 \leqslant i \leqslant p$. If a collection of 1–1 mappings is induced such that when restricted to each X_i, it forms a transitive collection of mappings from X_i to Y_i then we say that the problem induces *partitioned transitivity.*

LEMMA 2. *Suppose a partitioned transitive collection of f mappings is induced by a problem. If* $|X_i| = m_i$ *and* $|Y_i| = n_i$ *for* $1 \leqslant i \leqslant p$, *and a balanced bisection surface separates each* Y_i *into two parts with* k_i $(\leqslant n_i/2)$ *exists in one part, then the information content is at least* $\sum_{i=1}^{p} m_i k_i / f$.

Proof. Every input variable in X_i is mapped into output variables (in Y_i) belonging to the other side of the bisection surface at least k_i times. Similar arguments as in the proof of Lemma 1 can be applied to finish the proof. ∎

5. MATRIX MULTIPLICATION

Before we can prove an area–period lower bound for the problem of matrix multiplication we need to show an interesting combinatorial lemma.

LEMMA 3. *Suppose we arbitrarily partition an* $m \times n$ *matrix of points into two parts* P_1 *and* P_2 *with* P_1 *containing k points,* $k \leqslant mn/2$. *If, for row i, there are* r_i^1 *(resp.* r_i^2*) points belonging to* P_1 *(resp.* P_2*) and, for column j, there are* c_j^1 *(resp.* c_j^2*) points belonging to* P_1 *(resp.* P_2*), then* $\sum_{i=1}^{m} r_i + \sum_{j=1}^{n} c_j \geqslant k/2$, *where* $r_i = \min(r_i^1, r_i^2)$ *and* $c_j = \min(c_j^1, c_j^2)$.

Proof. In each row or column we mark the points which belong to the lesser number part. (If the numbers are equal we freely choose one part.) We then have $\sum_{r=1}^{m} r_i + \sum_{j=1}^{n} c_j \geqslant$ the number of marked points.

If every column contains at least $\lceil k/2n \rceil$ marked points, then the number of marked points $\geqslant \lceil k/2n \rceil \cdot n \geqslant k/2$ as we desire. So let us assume that there is a column in which at least $m - \lfloor k/2n \rfloor$ points are not marked. Let these points be in row $b_1, \ldots, b_u$, where $u \geqslant m - \lfloor k/2n \rfloor$. They must belong to the same part, say P_i. Points in these rows but not in P_i must be marked. There are two cases, $i = 1$ or 2, to be discussed:

(i) If $i = 1$ then

$$\begin{aligned}\text{number of marked points} &\geqslant \sum_{j=1}^{u} r_{b_j}^2 \geqslant \sum_{j=1}^{u} (n - r_{b_j}^1) \\ &\geqslant (m - \lfloor k/2n \rfloor)\, n - \sum_{j=1}^{u} r_{b_j}^1 \\ &\geqslant ((mn - k) + k/2) - k \\ &\geqslant k/2.\end{aligned}$$

(ii) If $i = 2$ then

$$\begin{aligned}\text{number of marked points} &\geqslant \sum_{j=1}^{u} r_{b_j}^1 \geqslant \sum_{j=1}^{u} (n - r_{b_j}^2) \\ &\geqslant (m - \lfloor k/2n \rfloor)\, n - \sum_{j=1}^{u} r_{b_j}^2 \\ &\geqslant (mn - k/2) - (mn - k) \\ &\geqslant k/2. \quad \blacksquare\end{aligned}$$

We are now ready to prove the main theorem for the problem of matrix multiplication. We define the problem as $C = AB$, where $A = (a_{ij})$, $B = (b_{ij})$, $C = (c_{ij})$ are $m \times n$, $n \times p$, $m \times p$ matrices, respectively. Assume elements in A and B have bit length l.

THEOREM 5. *For the matrix multiplication problem,* $AT^{2\alpha} = \Omega((\min(mn, mp, np)\, l)^{1+\alpha})$ *for* $0 \leqslant \alpha \leqslant 1$.

Proof. For simplicity we assume l is even. If go down to the bit level, A, B, C can be thought as $m \times nl$, $n \times pl$, $m \times 2pl'$ matrices, where $l' \geqslant l$. The bit s of a_{ij} is denoted by $a_{ij}^{(s)}$.

By properly controlling the values of B we can induce a transitive collection of mappings between row i of A and row i of C, $1 \leqslant i \leqslant m$. What we can do is to make each $a_{ij}^{(s)}$, $1 \leqslant j \leqslant n$ and $1 \leqslant s \leqslant l/2$, mapped into any $c_{ij'}^{(s')}$, $1 \leqslant j' \leqslant p$ and $l/2 + 1 \leqslant s' \leqslant l$. This can be accomplished by combining the transitivities of cyclic shiftings

and binary multiplication as mentioned in the previous section. There are totally $\max(n, p)\, l$ mappings. By Lemma 2, any balanced bisection surface with respect to the $mpl/2$ observed point in C has information content bounded below by $\sum_{i=1}^{m} nl/2 \cdot r_i/\max(n, p)_l = (\min(n, p)/2p) \sum_{i=1}^{m} r_i$, where r_i are defined as in Lemma 2.

Let us denote bit slice s of column j of B and C by $B_j^{(s)}$ and $C_j^{(s)}$, respectively. $B_j^{(s)}$ can be cyclically shifted to $C_j^{(s)}$ if the values of A are properly arranged. Here we only consider $l/2+1 \leqslant s \leqslant l$. By Lemma 2 again, the information content of any balanced bisection surface is bounded below by $\sum_{j=1}^{pl/2} nc_j/\max(m, n) = \min(m, n)/m \sum_{j=1}^{pl/2} c_j$, where c_j are defined as in Lemma 2.

If we consider the matrix of points formed by $C_j^{(s)}$, $1 \leqslant j \leqslant p$, and $l/2+1 \leqslant s \leqslant l$, and take $k = mpl/4$ in Lemma 3, then $\max(\sum_{i=1}^{m} r_i, \sum_{j=1}^{pl/2} c_j) \geqslant mpl/16$. Therefore the information content is at least $\min(\min(n, p)/2p,\ \min(m, n)/m)\, mpl/16$. With this and Theorem 2 we finish the proof. ∎

Savage [16] has proved an area–time lower bounds for the matrix multiplication problem at the matrix element lever ($l = O(1)$). Even at the matrix element level our result is more complete because it puts no contraint on the matrix dimensions. In our notation, Savage's result can be written as

$$I \geqslant C\left(\frac{mp}{4}\left(1 - \frac{(2\max(m, n) - n)(2\max(p, n) - n)}{2\max(m, n)\max(p, n)}\right)\right)$$

when $(\max(m, n) - n)(\max(p, n) - n) \leqslant n/2$. Let us compare our result with Savage's in more detail:

(i) When $m, p \leqslant n$, both results imply $I = \Omega(mp)$.

(ii) When $m \leqslant n \leqslant p$ (similarly for $p \leqslant n \leqslant m$), both results imply $I = \Omega(mn)$.

(iii) When $n \leqslant m \leqslant p$ (similarly for $n \leqslant p \leqslant m$), our result gives $I = \Omega(mn)$. Under the condition $(m-n)(p-n) \leqslant n^2/2$, Savage's result becomes $I \geqslant C(n^2 - 2(m-n)(p-n))/8$. So in this case we get a better result.

When restricted to the square matrix case, $m = n = p = N$, Theorem 5 implies $AP^{2\alpha} = \Omega((N^2 l)^{1+\alpha})$. Kung and Leiserson [9] have designed a renowned hexagonal network for computing square matrix multiplication. In this network, if we use the multiplier of Preparata [12] ($A = O(l)$ and $T = O(l^{1/2})$) then for the whole network $A = O(N^2 l)$ and $T = O(Nl^{1/2})$, or $AT^{2\alpha} = O((N^2 l)^{1+\alpha})$ which matches the area–time lower bound. Preparata and Vuillemin [13] have designed a family of pipelined chips for square matrix multiplication which achieve the area–time lower bound with $O(\log(N \cdot l^{1/2}) \leqslant T \leqslant O(Nl^{1/2})$ if the Preparata's multiplier is used.

6. Exact Area Bound

In Theorem 1 or 2, if we set $\alpha = 0$ then $A = \Omega(I)$. From the proof of Theorem 5 we know that for the matrix multiplication problem $I = \Omega(\min(mn, mp, np)\, l)$. In the following we will show that $\min(mn, mp, np)\, l$ is the exact bound for chip area.

For the case $n \geqslant m, p$, we design a mesh-connected network with mp cells as illustrated in Fig. 6.

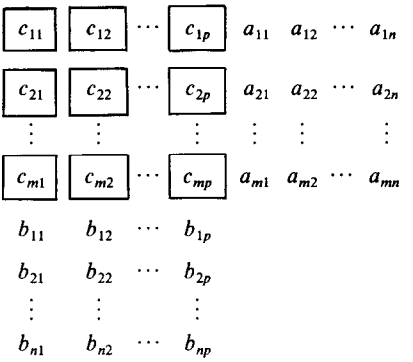

FIG. 6. Network with $O(mpl)$ area.

Each cell contains a register c_{ij} initialized as 0. At stage k, $1 \leqslant k \leqslant n$, the column of input values $a_{1k},\ldots, a_{mk}$ march through the network leftward and stay in each column of cells. Also the row of input values $b_{k1},\ldots, b_{kp}$ march through the network upward and stay in each row of cells. Then $a_{ik} \times b_{kj}$ is added to c_{ij}. After n stages c_{ij} has the final result.

There are one multiplication and one addition that take place in each cell at each stage. In order to keep the area of a cell as small as possible, we can serially input and output the data bits and use the *add-and-shift* method for multiplication and the *ripple-carry* method for addition. Every cell occupies $O(l)$ area and the whole network occupies $O(mpl)$ area.

For the case $m \geqslant n, p$ we also design a mesh-connected network with np cells as illustrated in Fig. 7.

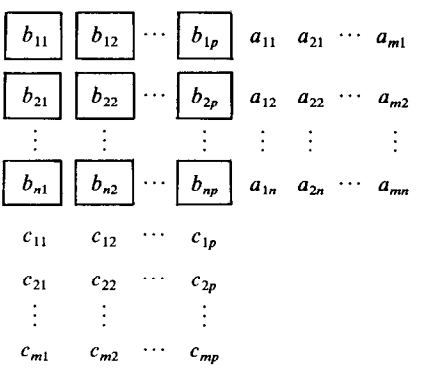

FIG. 7. Network with $O(npl)$ area.

This time the cells hold the values of B. At each stage one column of input values of A march in leftward and stay in each column of cells first. Then one row of C

(initialized as 0s) march in upward and accumulate their results. The area of the network is $O(npl)$.

For the case $p \geqslant m, n$, we can similarly design a circuit with area $O(mpl)$. Since for all cases the area lower bound $\min(mn, mp, np)\, l$ can actually be achieved, we conclude that this bound is exact.

THEOREM 6. *For the matrix multiplication problem,* $A = \Theta(\min(mn, mp, np)\, l)$.

7. RELATED PROBLEMS

As Savage pointed out in [16], both transitive closure and matrix inversion problems can be reduced to the square matrix multiplication problem by the following identities [2, pp. 203, 242]:

$$\begin{pmatrix} I & A & 0 \\ 0 & I & B \\ 0 & 0 & I \end{pmatrix}^{*} = \begin{pmatrix} I & A & AB \\ 0 & I & B \\ 0 & 0 & I \end{pmatrix},$$

$$\begin{pmatrix} I & A & 0 \\ 0 & I & B \\ 0 & 0 & I \end{pmatrix}^{-1} = \begin{pmatrix} I & -A & AB \\ 0 & I & -B \\ 0 & 0 & I \end{pmatrix}.$$

Theorem 5 therefore provides indirect proofs for the area-period lower bounds for these two problems.

THEOREM 7. *For the transitive closure problem,* $AP^{2\alpha} = \Omega(N^{2(1+\alpha)})$ *for* $0 \leqslant \alpha \leqslant 1$. *For the matrix inversion problem,* $AP^{2\alpha} = \Omega((N^2 l)^{1+\alpha})$ *for* $0 \leqslant \alpha \leqslant 1$.

It is also easy to give direct proofs for these two problems by a method analogous to the one for matrix multiplication. Guibas *et al.* [8] have designed a VLSI network for transitive closure problem which achieves the area–time lower bound with $A = O(N^2)$ and $T = O(N)$.

Another related problem is the all-pair shortest-paths problem. The problem is to compute for each pair of vertices in a graph, the weight of the least-weight path between them. Here we only look at the special case when every pair of vertices in the graph of $N = 3n$ vertices is connected by an edge of weight 0 or 1. Let the cost (Boolean) matrix be M and denote the resulting matrix as $S(M)$. Then $S(M) = (\bar{M})^{*}$, where $\bar{M}$ represents the complementary matrix of M. By substituting special matrices for M, we have the following identity:

$$S\begin{pmatrix} \bar{I} & \bar{A} & \bar{0} \\ \bar{0} & \bar{I} & \bar{B} \\ \bar{0} & \bar{0} & \bar{I} \end{pmatrix} = \begin{pmatrix} \bar{I} & \bar{A} & C \\ \bar{0} & \bar{I} & \bar{B} \\ \bar{0} & \bar{0} & \bar{I} \end{pmatrix},$$

where $C = \overline{AB}$.

Now, if A is fixed to be cyclic permutation matrices, there induces a transitive collection of cyclic shiftings from each column in $\bar{B}$ to its corresponding column in C. Same property holds for every row of C when B is fixed to be cyclic permutation matrices. By applying Lemma 2 and Theorem 2, we obtain the area–period lower bound for the all-pair shortest-paths problem.

THEOREM 8. *For the all-pair shortest-paths problem,* $AP^{2\alpha} = \Omega(N^{2(1+\alpha)})$ *for* $0 \leqslant \alpha \leqslant 1$.

The network for transitive closure just mentioned above can be modified by using adders described in [12] ($A = O(l)$ and $T = Ol^{1/2}$)) and comparators by the same design method to solve the all-pair shortest-paths problem with $AT^{2\alpha} = O((N^2 l)^{1+\alpha})$. We conjecture that this is also the lower bound.

ACKNOWLEDGMENT

The authors gratefully acknowledge the insightful comments and valuable suggestions of referee B.

REFERENCES

1. H. ABOLESON AND P. ANDREAE, Information transfer and area–time trade-offs for VLSI multiplication, *Comm. ACM* **23**, No. 1 (1980), 20–23.
2. A. V. AHO, J. E. HOPCROFT, AND J. D. ULLMAN, "The Design and Analysis of Computer Algorithms," Addison–Wesley, Menlo Park, Calif., 1974.
3. A. V. AHO, J. D. ULLMAN, AND M. YANNAKAKIS, On notions of information transfer in VLSI circuits, *in* "Proc. 15th Annual ACM Sympos. on Theory of Computing," Boston, Mass., April 1983, pp. 133–139.
4. G. BILARDI, M. PRACCHI, AND F. P. PREPARATA, A critique of network speed in VLSI models of computation, *IEEE J. Solid-State Circuits*, **SC-17**, No. 4 (1982), 696–702.
5. B. CHAZELLE AND L. MONIER, A model of computation for VLSI with related complexity results, *in* "Proc. 13th Annual ACM Sympos. on Theory of Computing," Milwaukee, WI, May 1981, pp. 318–325.
6. R. P. BRENT AND L. M. GOLDSCHLAGER, Some area-time tradeoffs for VLSI, *SIAM J. Comput.* **11**, No. 4 (1982), 737–747.
7. R. P. BRENT AND H. T. KUNG, The area–time complexity of binary multiplication, *J. Assoc. Comput. Mach.* **28**, No. 3 (1981), 521–534.
8. L. J. GUIBAS, H. T. KUNG, AND C. D. THOMPSON, Direct VLSI implementation of combinatorial algorithms, *in* "Conf. on VLSI Technical Design and Fabrication," California Institute of Technology, 1979, pp. 509–525.
9. H. T. KUNG AND C. E. LEISERSON, Systolic arrays for VLSI, *in* "Introduction to VLSI Systems," Section 8.3, Addison–Wesley, Menlo Park, Calif., 1980.
10. R. J. LIPTON AND R. SEDGEWICK, Lower Bounds for VLSI, *in* "Proc. 13th Annual ACM Sympos. on Theory of Computing," Milwaukee, Wisc., May 1981, pp. 300–307.
11. C. A. MEAD AND L. A. CONWAY, "Introduction to VLSI Systems," Addison–Wesley, Menlo Park, Calif., 1980.
12. F. P. PREPARATA, A mesh-connected area–time optimal VLSI multiplier of large integers, *IEEE Trans. Comput.* **C-32**, No. 2 (1983), 194–198.

13. F. P. PREPARATA AND J. E. VUILLEMIN, Area–time optimal VLSI network for parallel matrix multiplication, *Inform. Process. Lett.*, **11**(1980), 77–80.
14. V. RAMACHANDRAN, On driving many long lines in a VLSI layout, *in* "*Proc.* 23rd Annual Sympos. on Foundations of Computer Science," Chicago, Il., November 1982, pp. 369–372.
15. J. E. SAVAGE, "Planar Circuit Complexity and the Performance of VLSI Algorithms," Technical Report No. CS-69, INRIA, Rocquencourt, 1981.
16. J. E. SAVAGE, Area–time tradeoffs for matrix multiplication and related problems in VLSI models, *J. Comput. System Sci.* **22**, No. 2 (1981), 230–242.
17. C. D. THOMPSON, "A Complexity Theory for VLSI," Ph.D. thesis, Carnegie–Mellon University, 1980.
18. J. E. VUILLEMIN, A combinatorial limit to the computing power of VLSI circuits, *IEEE Trans. Comput.* **C-32**, No. 3 (1983), 294–300.
19. A. C. YAO, The entropic limitations on VLSI computations, *in* "Proc. 13th Annual ACM Sympos. on Theory of Computing," Milwaukee, Wisc., May 1981, pp. 308–311.