

## A PARALLEL ALGORITHM TO SOLVE THE STABLE MARRIAGE PROBLEM

S. S. TSENG and R. C. T. LEE

*Institute of Computer Engineering,  
College of Engineering,  
National Chiao Tung University,  
Hsinchu, Taiwan, Republic of China*

*Institute of Computer and Decision Sciences,  
National Tsing Hua University,  
Hsinchu, Taiwan, Republic of China*

### **Abstract.**

In this paper a parallel algorithm to solve the stable marriage problem is given. The worst case performance of this algorithm is stated. A theoretical analysis shows that the probability of the occurrence of this worst case is extremely small. For instance, if there are sixteen men and sixteen women involved, then the probability that the worst case occurs is only  $10^{-45}$ . Possible future research is also discussed in this paper.

### **1. Introduction.**

In this paper, we are concerned with the stable marriage problem [Gale and Shapley 1962, McVitie and Wilson 1970, McVitie and Wilson 1971, Wilson 1972, Allison 1983]. We shall show that this problem can be solved by a divide-and-conquer approach [Bentley 1980]. Using this approach, a parallel algorithm is proposed in this paper.

It is pointed out in this paper that in the worst case, our parallel algorithm does not help too much. However, a theoretical analysis also shows that the probability that this worst case occurs is extremely small. For instance, if there are eight men and eight women, then the probability of the occurrence of the worst case is only  $5 \cdot 10^{-16}$ . If the number of men is 16, then that probability is reduced to  $10^{-45}$ .

Possible future research is discussed in Section 6.

### **2. The stable marriage problem.**

The stable marriage problem is defined as follows: We are given  $N$  men and  $N$  women and each man ranks each woman from 1 to  $N$  and vice versa. A set of marriages is a one-to-one correspondence of men to women. If there does

not exist an unmarried pair both preferring each other to their current partners, then we say that this set of marriages is stable.

Before giving an example, we shall denote  $M_i(W_i)$  to be Man  $i$  (Woman  $i$ ). This notation will be used for the rest of this paper.

EXAMPLE 2.1. Consider the following case:

$M_1$	1	2		$W_1$	1	2
$M_2$	2	1		$W_2$	2	1
Man's ranking				Woman's ranking		

The set of marriages  $((M_1, W_1), (M_2, W_2))$  is stable because each man has got his first choice, and each woman her first choice as well.

The set of marriages  $((M_1, W_2), (M_2, W_1))$  is not a stable one because there exists a man, namely  $M_1$ , and a woman, namely  $W_1$ , who prefer each other to their present partners;  $M_1$  prefers  $W_1$  to  $W_2$  and  $W_1$  prefers  $M_1$  to  $M_2$ .

The stable marriage problem was first introduced by Gale and Shapley [3] and later discussed in [6]. In this paper, we shall show that the stable marriage problem can be solved by a parallel algorithm. Before giving the algorithm formally, let us first show that we can attack the problem by the divide-and-conquer strategy [2].

### 3. The divide-and-conquer approach to solve the stable marriage problem.

The divide-and-conquer approach was discussed in [2]. If a problem can be solved by a divide-and-conquer approach, a solution can usually be obtained rather easily.

Let us consider the following case:

$M_1$	3	2	1	4
$M_2$	3	1	2	4
$M_3$	4	3	1	2
$M_4$	2	4	3	1
$W_1$	1	3	2	4
$W_2$	4	1	3	2
$W_3$	4	3	1	2
$W_4$	2	4	3	1

In the matrices above we see that  $M_1$  prefers  $W_3$  to  $W_2$ ,  $W_2$  to  $W_1$  and  $W_1$  to  $W_4$ .

We first divide the problem into four subproblems. Each sub-problem contains only one man. Thus each man can propose his choice as follows:

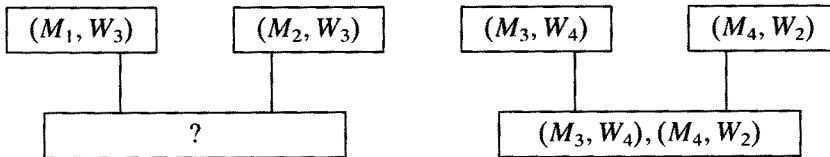
Pair 1  $(M_1, W_3)$

Pair 2  $(M_2, W_3)$

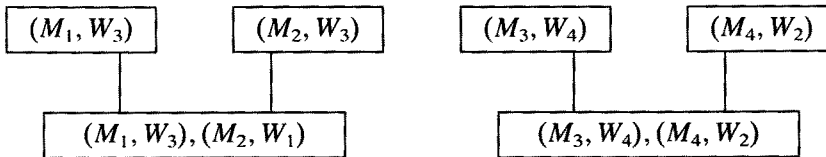
Pair 3  $(M_3, W_4)$

Pair 4  $(M_4, W_2)$

Having obtained four sub-solutions, we shall now try to merge them into one solution. The first stage is to merge Pair 1 with Pair 2 and Pair 3 with Pair 4 as illustrated below:



For the pairs  $(M_3, W_4)$  and  $(M_4, W_2)$ , since different men have chosen different women, we may simply combine these two pairs. For  $(M_1, W_3)$  and  $(M_2, W_3)$ , note that both men have chosen the same woman. We therefore have to do something. So far as  $W_3$  is concerned, she prefers  $M_1$  to  $M_2$ . Therefore,  $W_3$  accepts  $M_1$  and rejects  $M_2$ .  $M_2$ , the rejected man, proposes his next best choice, namely  $W_1$ . Thus our solution is now as follows:



The next merging step produces the following solution:

$$(M_1, W_3) (M_2, W_1) (M_3, W_4) (M_4, W_2)$$

Since no two men propose to the same woman, we have got a stable marriage solution.

The generation of initial sub-solutions can be done in parallel. All of the subsequent merging steps can also be executed in parallel. Thus our algorithm is a parallel one.

If sufficient processors are used, within each merging step, many operations can be executed in parallel. Let us now illustrate this point by an example.

$M_1$	1	4	5	6	2	3	7	8	$W_1$	8	7	6	4	3	5	1	2
$M_2$	1	2	4	8	7	6	3	5	$W_2$	1	5	4	3	6	2	7	8
$M_3$	3	2	1	4	8	7	5	6	$W_3$	1	3	2	4	7	8	6	5
$M_4$	4	6	7	8	1	3	2	5	$W_4$	4	7	8	2	6	1	3	5
$M_5$	2	3	4	6	7	1	8	5	$W_5$	3	1	2	5	6	8	7	4
$M_6$	2	1	8	6	5	7	3	4	$W_6$	1	2	3	4	5	6	7	8
$M_7$	5	3	1	4	8	6	2	7	$W_7$	2	1	3	5	4	6	7	8
$M_8$	5	7	6	3	1	2	4	8	$W_8$	7	4	2	1	5	6	8	3

For the data above, we shall finally produce the following two subsolutions:

Sub-solution:  $(M_1, W_1) (M_2, W_2) (M_3, W_3) (M_4, W_4)$

Sub-solution:  $(M_5, W_2) (M_6, W_1) (M_7, W_3) (M_8, W_5)$

When we merge these two sub-solutions, we obtain

$(M_1, W_1) (M_2, W_2) (M_3, W_3) (M_4, W_4) (M_5, W_2) (M_6, W_1) (M_7, W_3) (M_8, W_5)$ .

There are three conflicting pairs:  $M_1, M_6$ ;  $M_2, M_5$ ;  $M_3, M_7$ .

Each pair of men proposes to the same woman.

$M_1$  and  $M_6$  both choose  $W_1$ . Since  $W_1$  prefers  $M_6$  to  $M_1$ ,  $M_1$  is rejected. Similarly,  $M_2$  and  $M_7$  will also be rejected. These three men,  $M_1$ ,  $M_2$  and  $M_7$ , will all propose to their next best choices. Again, these proposals can be made in parallel.

In the following section, we shall present our algorithm formally.

#### 4. A parallel algorithm to find a male (female) optimal stable solution.

To simplify our discussion we shall assume that there are  $N = 2^n$  men and  $N$  women. A male optimal stable solution is a stable solution where every man is at least as well off under it as under any other stable solutions. A female optimal stable solution is similar except that the women get their best possible choices. In the following we shall give an algorithm which will always produce a male optimal stable solution.

##### ALGORITHM A.

An algorithm which produces a male optimal stable solution.

*Input:* A male ranking matrix and a female ranking matrix.

*Output:* A male optimal stable solution.

*Step 1:* Divide the problem into two sub-problems, by halving the male ranking matrix. Call these two sub-problems  $P_1$  and  $P_2$ .

*Step 2:* Recursively apply this algorithm to find male optimal stable solutions for  $P_1$  and  $P_2$ . Call these two solutions  $S_1$  and  $S_2$ .

*Step 3:* Apply Algorithm B which is a merging algorithm to combine  $S_1$  and  $S_2$  into  $S$ .

In this algorithm, a merging procedure is used. Let us first introduce some definitions.

In a solution, suppose  $M_{i_1}, \dots, M_{i_k}$ ,  $k \geq 2$ , propose to the same woman  $W_i$ . Without loss of generality, we shall assume that so far as  $W_i$  is concerned, the ranking of  $M_{i_k}$  is the highest. That is,  $W_i$  prefers  $M_{i_k}$  to  $M_{i_1}, \dots, M_{i_k}$  to  $M_{i_{k-1}}$ . Then  $W_i$  will accept  $M_{i_k}$  and reject  $M_{i_1}, \dots, M_{i_{k-1}}$ . We shall say that  $\{M_{i_1}, \dots, M_{i_{k-1}}\}$  is the set of rejected men of  $W_i$ . In a solution  $S$ , let  $W_s$  denote the set of women who are proposed to by more than one man. Let  $R_s$  be the union of the sets of rejected men of members in  $W_s$ . Then  $R_s$  is called the set of rejected men associated with  $S$ .

**ALGORITHM B.**

A merging algorithm which produces a male optimal stable solution out of two male optimal stable sub-solutions.

*Input:* Two male optimal stable solutions  $S_1$  and  $S_2$  and their associated ranking matrices.

*Output:* A male optimal stable solution which combines  $S_1$  and  $S_2$ .

*Step 1:* Let  $S$  be the union of  $S_1$  and  $S_2$ .

*Step 2:* If no two men propose to the same woman in  $S$ , then accept  $S$  as the solution and return. Otherwise, go to Step 3.

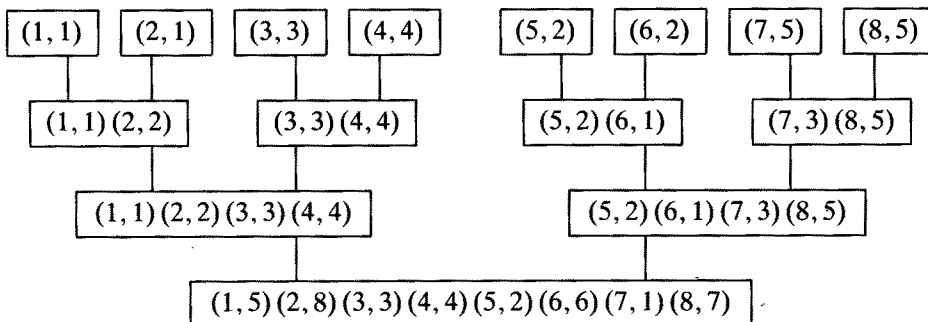
*Step 3:* For each man  $M_i$  in  $R_s$ , and for the set of rejected men associated with  $S$ , replace  $(M_i, W_j)$  in  $S$  by  $(M_i, W_k)$  where  $W_k$  is the next best choice of  $M_i$ .

*Step 4:* Go to step 2.

**EXAMPLE 4.1.** Consider the following case:

$M_1$	1	4	5	6	2	3	7	8	$W_1$	8	7	6	4	3	5	1	2
$M_2$	1	2	4	8	7	6	3	5	$W_2$	1	5	4	3	6	2	7	8
$M_3$	3	2	1	4	8	7	5	6	$W_3$	1	3	2	4	7	8	6	5
$M_4$	4	6	7	8	1	3	2	5	$W_4$	4	7	8	2	6	1	3	5
$M_5$	2	3	4	6	7	1	8	5	$W_5$	3	1	2	5	6	8	7	4
$M_6$	2	1	8	6	5	7	3	4	$W_6$	1	2	3	4	5	6	7	8
$M_7$	5	3	1	4	8	6	2	7	$W_7$	2	1	3	5	4	6	7	8
$M_8$	5	7	6	3	1	2	4	8	$W_8$	7	4	2	1	5	6	8	3
	Men's ranking									Women's ranking							

The merging steps are illustrated as follows:



It can be easily seen that the solution obtained by using Algorithm A is the male optimal stable solution.

### 5. The performance of Algorithm A.

In this section, we shall present the probability that the worst case occurs when our parallel algorithm is employed. Our result is based upon a worst case analysis of the McVitie-Wilson algorithm [6, 7].

Wilson [8] showed that the maximal number of proposals to obtain the male optimal stable solution is  $N^2 - N + 1$  when the McVitie-Wilson algorithm is used. Besides, he also showed that for the problem sizes ( $N$ ) 5, 10, 15, 20, 25, 30, 40 and 50, the theoretical maximum ( $N^2 - N + 1$ ) is much larger than the actual maxima which were obtained by running 1000 random sets of data for each  $N$ . We shall show that the probability that the worst case occurs is of the order  $CN^{-2N+\frac{3}{2}}e^{-N}$  when the McVitie-Wilson algorithm is used.

EXAMPLE 5.1. Consider the following case:

$M_1$ 1 7 6 5 4 3 2 8 $M_2$ 2 1 7 6 5 4 3 8 $M_3$ 3 2 1 7 6 5 4 8 $M_4$ 4 3 2 1 7 6 5 8 $M_5$ 5 4 3 2 1 7 6 8 $M_6$ 6 5 4 3 2 1 7 8 $M_7$ 7 6 5 4 3 2 1 8 $M_8$ 1 7 6 5 4 3 2 8	$W_1$ 7 6 5 4 3 2 1 8 $W_2$ 1 8 7 6 5 4 3 2 $W_3$ 2 1 8 7 6 5 4 3 $W_4$ 3 2 1 8 7 6 5 4 $W_5$ 4 3 2 1 8 7 6 5 $W_6$ 5 4 3 2 1 8 7 6 $W_7$ 6 5 4 3 2 1 8 7 $W_8$ 8 7 6 5 4 3 2 1
Men's ranking	Women's ranking

The male optimal stable solution is (1, 2) (2, 3) (3, 4) (4, 5) (5, 6) (6, 7) (7, 1) (8, 8), and the number of men's proposals in this solution is 57. This is a worst case for  $N$  equal to 8.

By using the sequential algorithm to find the male optimal stable solution, the number of proposals required is at most  $N^2 - N + 1$ . Now, we are concerned with the number of steps required to obtain the male optimal stable solution by using our parallel algorithm.

LEMMA 5.1. *For a worst case execution of McVitie-Wilson's algorithm the following two statements are true:*

1. *Let  $M_k$  finally propose to this  $N$ th choice  $W_i$ . Then the  $N$ th column of the male ranking matrix consists only of one woman  $W_i$  and the  $(N-1)$ th column consists of all the other  $N-1$  women.*
2. *Let  $M_j$  propose to his  $(N-1)$ th choice  $W_a$ . Then the first choice of  $W_a$  must be  $M_j$ .*

PROOF. It is well known that there is only one man  $M_k$  who will propose to his  $N$ th choice  $W_i$  and all of the other men propose to their  $(N-1)$ th choice women if the worst case occurs by using the McVitie-Wilson algorithm. This means that all of the  $(N-1)$ th choice women of men, except the  $(N-1)$ th choice woman of  $M_k$ , are all distinct and all different from  $W_i$ . Furthermore, it can be easily seen that  $W_i$  has not been proposed to by any man before  $M_k$  proposes to her and this proposal must be the last proposal of men. This means that  $W_i$  must be the  $N$ th choice woman of each man. Thus, we can conclude that Statement 1 is true.

For Statement 2, note that  $M_k$  proposes to his  $N$ th choice  $W_i$ . Any other man  $M_j$  proposes to his  $(N-1)$ th choice  $W_a$ . This means that  $W_a$  prefers  $M_j$  to all other men. Equivalently, we may say that the first choice of  $W_a$  is  $M_j$ . Thus, Statement 2 is true. ■

THEOREM 5.1. *If  $N^2 - N + 1$  proposals are needed in McVitie-Wilson's algorithm to obtain the male optimal stable solution, then the probability that this worst case occurs is of the order  $CN^{-2N+\frac{7}{2}}e^{-N}$  with  $2.5 < C < 3.5$ .*

PROOF. If we are given  $N$  men and  $N$  women, then each male ranking matrix has probability  $1/(N!)^N$  of occurring and so does each female ranking matrix. If the worst case occurs, according to Statement 1 of Lemma 5.1, there are  $N$  possible choices of  $W_i$  and  $N(N-1)/2(N-1)!$  possible choices of column  $N-1$ . Therefore, there are totally  $N^2(N-1)/2(N-1)!((N-2)!)^N$  desirable male ranking matrices. Moreover, according to Statement 2 of Lemma 5.1, the total number of desirable female ranking matrices is  $N((N-1)!)^N$ . Let the probability that the worst case occurs for the McVitie-Wilson algorithm be denoted as  $P_s$ .

$$P_s = \frac{N^2(N-1)(N-1)!((N-2)!)^N \cdot N((N-1)!)^N}{2(N!)^N(N!)^N} = \frac{N^3(N-1)(N-1)!}{2(N-1)^N \cdot N^{2N}}$$

$$\simeq CN^{-2N+\frac{7}{2}}e^{-N} \text{ with } 2.5 < C < 3.5 \text{ when } N \geq 2. \quad \blacksquare$$

It is clear that the worst case may occur by using our parallel algorithm only if  $N^2 - N + 1$  proposals are needed when the McVitie-Wilson algorithm is used.

LEMMA 5.2. *The number of steps needed to obtain the male optimal stable solution by using our parallel algorithm is at most  $N^2 - 2N + \lceil \log_2 N \rceil$ .*

PROOF. In our parallel algorithm, for the worst case, if there is always only one man in  $R_s$ , then the parallel algorithm reduces to the sequential algorithm except for the first merging step. It takes  $\lceil \log_2 N \rceil$  to complete the first merging step. It takes  $(N - 2) + (N - 1)(N - 2) = N^2 - 2N$  steps to complete the rest of the execution of our parallel algorithm. Thus the complexity of our algorithm is  $N^2 - 2N + \lceil \log_2 N \rceil$ . ■

Example 5.1 is also an example in which the number of steps needed to obtain the male optimal stable solution is  $N^2 - 2N + \lceil \log_2 N \rceil = 51$  by using our parallel algorithm.

LEMMA 5.3. *In the worst case of our parallel algorithm the first column of the male ranking matrix consists of exactly  $N - 1$  women.*

PROOF. If the first column of the male ranking matrix consists of  $N - 2$  or less women, then more than one conflict are resolved when the first merging is completed. Therefore,  $N^2 - 2N - 1$  or less steps are needed to complete the rest of the execution of our parallel algorithm. This result contradicts the assumption that  $N^2 - 2N + \lceil \log_2 N \rceil$  steps are needed by our parallel algorithm. Thus, this lemma is proved. ■

THEOREM 5.2. *The probability that the worst case occurs in the parallel algorithm is less than  $CN^{-2N+5}e^{-2N}$  with  $30 < C < 70$ , for  $N > 2$ .*

PROOF. Every worst case which occurs when our parallel algorithm is used corresponds to a worst case of the McVitie-Wilson algorithm, but not vice versa. So the probability that a worst case occurs for our algorithm is not greater than the worst case probability of the McVitie-Wilson algorithm.

According to Lemma 5.3, the number of possible choices of the elements of the male ranking matrix from column 1 to column  $N - 2$  is reduced from  $((N - 2)!)^N$  to  $N(N - 1)/2(N - 1)!((N - 3)!)^N$  or less. Let the probability that a worst case occurs for our parallel algorithm be denoted by  $P_p$ . Then by using Theorem 5.1.,

$$P_p = P_s \cdot \frac{N(N - 1)(N - 1)!((N - 3)!)^N}{2((N - 2)!)^N} \simeq CN^{-2N+5}e^{-2N} \quad \blacksquare$$



Theorem 5.2 indicates that the probability that the worst case occurs is extremely small. For example, let  $N$  be equal to 8. The probability that the worst case occurs is less than  $4.7 \cdot 10^{-16}$ . If  $N$  is equal to 16, then that probability is less than  $1.3 \cdot 10^{-45}$ .

## 6. Conclusions and future research.

In this paper we have shown that the stable marriage problem can be solved by a divide-and-conquer approach. Because of this, the problem can be solved in a parallel fashion.

Much research can be done in the future.

(1) The stable marriage problem is a special case of the general assignment problem [4] in operations research. We believe that it will be appropriate to attack the general assignment problem directly to see if that class of problems can be solved in parallel.

(2) We conjecture that for the worst case, the column vectors, except the last one, share the same property, as stated in Lemma 5.3. We hope that we can prove this property in the future.

(3) The average performance of our parallel algorithm is quite important. It appears to be very difficult to obtain the average performance, and we are presently looking into this problem.

## REFERENCES

1. L. Allison, *Stable marriages by coroutines*, Information Processing Letters, Vol. 16, No. 2 (Feb. 1983), 61–65.
2. J. L. Bentley, *Multidimensional Divide-and-conquer*, Communications of the ACM, Vol. 23, No. 4, (Apr. 1980), 214–239.
3. D. Gale and L. S. Shapley, *College admissions and the stability of marriage*, Am. Math. Monthly 69 (1962), 9–15.
4. F. S. Hiller and G. J. Lieberman, *Introduction to Operations Research*, 3rd ed., Holden-Day, San Francisco (1980).
5. D. G. McVitie and L. B. Wilson, *Stable marriage assignment for unequal sets*, BIT 10 (1970), 295–309.
6. D. G. McVitie and L. B. Wilson, *The stable marriage problem*, Communications of the ACM, Vol. 14, No. 7, (July 1971), 486–490.
7. D. G. McVitie and L. B. Wilson, Algorithm 411, Pts. 1–3, *Three procedures for the stable marriage problem*, Communications of the ACM, Vol. 14, No. 7, (July 1971), 491–492.
8. L. B. Wilson, *An analysis of the stable marriage assignment algorithm*, BIT 12 (1972), 569–575.