COT 5405: Advanced Algorithms
Fall 2011

Assignment 2 Solution

1. (20 points) Given eight processors with data 1, 3, 2, 4, 5, 3, 1, 8 respectively, show the messages sent at each step of the all-gather operation, and the data present in each processor at the end of each step.

Answer:

Step 1

Processor	1	2	3	4	5	6	7	8
Initial data	1	3	2	4	5	3	1	8
Message sent to	2	1	4	3	6	5	8	7

Step 2

Processor	1	2	3	4	5	6	7	8
Initial data	13	13	24	24	53	53	18	18
Message sent to	3	4	1	2	7	8	5	6

Step 3

Processor	1	2	3	4	5	6	7	8
Initial data	1324	1324	1324	1324	5318	5318	5318	5318
Message sent to	5	6	7	8	1	2	3	4

Final data: Each processors has: 13245318

2. (20 points) Assume that array *A[n][n][n]* is stored in column major order. Show an ordering of loops which will print all elements of *A* with $O(n^3/L)$ cache complexity, in the ideal cache model. (Hint: If you are unsure about column major ordering in three dimensions, first think about the order in which indices change in two dimensions, with column major ordering. Then, extend this to three dimensions.)

Answer:

```
for k = 1 to n
  for j = 1 to n
    for i = 1 to n
       print A[i][j][k]
```

3. (20 points) Consider the following linear recurrence: $x_{i+3} = a_{i+3}\, x_i + b_{i+3}\, x_{i+1} + c_{i+3}\, x_{i+2}$, where x_0, x_1, and x_2 are given. Formulate the solution of this recurrence as a prefix computation and give the parallel time complexity, speedup, and efficiency if parallel prefix is used to solve it.

Answer:

$x_{3i+3} = a_{3i+3}\, x_{3i} + b_{3i+3}\, x_{3i+1} + c_{3i+3}\, x_{3i+2}$
$x_{3i+4} = a_{3i+4}\, x_{3i+1} + b_{3i+4}\, x_{3i+2} + c_{3i+4}\, x_{3i+3}$
$x_{3i+5} = a_{3i+5}\, x_{3i+2} + b_{3i+5}\, x_{3i+3} + c_{3i+5}\, x_{3i+4}$

$x_{3i+4} = a_{3i+4}\, x_{3i+1} + b_{3i+4}\, x_{3i+2} + c_{3i+4}\, (a_{3i+3}\, x_{3i} + b_{3i+3}\, x_{3i+1} + c_{3i+3}\, x_{3i+2})$
$\quad = c_{3i+4}\, a_{3i+3}\, x_{3i} + (a_{3i+4} + c_{3i+4}\, b_{3i+3})\, x_{3i+1} + (b_{3i+4} + c_{3i+4}\, c_{3i+3})\, x_{3i+2}$

$x_{3i+5} = a_{3i+5}\, x_{3i+2} + b_{3i+5}\, (a_{3i+3}\, x_{3i} + b_{3i+3}\, x_{3i+1} + c_{3i+3}\, x_{3i+2}) + c_{3i+5}\, [c_{3i+4}\, a_{3i+3}\, x_{3i} + (a_{3i+4} + c_{3i+4}\, b_{3i+3})\, x_{3i+1} + (b_{3i+4} + c_{3i+4}\, c_{3i+3})\, x_{3i+2}]$
$\quad = (b_{3i+5}\, a_{3i+3} + c_{3i+5}\, c_{3i+4}\, a_{3i+3})\, x_{3i} + (b_{3i+5}\, b_{3i+3} + c_{3i+5}[a_{3i+4} + c_{3i+4}\, b_{3i+3}])\, x_{3i+1} + (a_{3i+5} + b_{3i+5}\, c_{3i+3} + c_{3i+5}[b_{3i+4} + c_{3i+4}\, c_{3i+3}])\, x_{3i+2}$

From the above, we can write the recurrence as
$X_0 = [x_0, x_1, x_2]^T$, $X_{i+1} = A_i\, X_i$, where X_i is defined as $[x_{3i}, x_{3i+1}, x_{3i+2}]^T$ and

$$A_i = \begin{bmatrix} a_{3i+3} & b_{3i+3} & c_{3i+3} \\ c_{3i+4}\, a_{3i+3} & a_{3i+4} + c_{3i+4}\, b_{3i+3} & b_{3i+4} + c_{3i+4}\, c_{3i+3} \\ b_{3i+5}a_{3i+3}+c_{3i+5}c_{3i+4}\, a_{3i+3} & b_{3i+5}b_{3i+3}+c_{3i+5}[a_{3i+4}+c_{3i+4}b_{3i+3}] & a_{3i+5}+b_{3i+5}c_{3i+3}+c_{3i+5}[b_{3i+4}+c_{3i+4}c_{3i+3}] \end{bmatrix}$$

The recurrence above is the desired prefix computation. Note that X_i needs to be computed only up to $i = n/3 - 1$.

The sequential computation takes around 2n arithmetic operations. The parallel computation requires log n steps with each step involving around 45 arithmetic operations (multiplication of 3x3 matrices) plus log n communications, for a time complexity of $O([1+t_s+t_b] \log n)$.

Speedup = $2n/[(45+t_s+t_b) \log n]$

Efficiency = $2n/([(45+t_s+t_b) \log n]\, n/3) = 6/[(45+t_s+t_b) \log n]$

4. (20 points) Consider a strict quadratic program with 0-1 constraints on the variables. Convert it to an equivalent strict quadratic program without any 0-1 constraints, and prove that the two are equivalent.

Answer:

0-1 strict quadratic program: min $\Sigma_{i,j}\, a_{ij}\, x_i\, x_j$
Subject to:
$\Sigma_{i,j}\, c^k_{ij}\, x_i\, x_j \le b^k$, for $1 \le k \le m$
$x_i \in \{0, 1\}$

Strict quadratic program: min $\Sigma_{i,j}\, a_{ij}\, (v_i\text{-}w_i)\, (v_j\text{-}w_j)/4$
Subject to:
$\Sigma_{i,j}\, c^k_{ij}\, (v_i\text{-}w_i)(v_j\text{-}w_j)/4 \le b^k$, for $1 \le k \le m$
$v_i^2 = w_i^2 = 1$
$(v_i\text{-}w_i)(v_j\text{-}w_j) \ge 0$ for all i,j, if $a_{ij} \neq 0$ or $c^k_{ij} \neq 0$ for any k

They are equivalent for the following reason.

Consider a feasible solution $\mathbf{x}_i$ for the first problem. If $\mathbf{x}_i = 0$ then choose $\mathbf{v}_i = \mathbf{w}_i = 1$ otherwise choose $\mathbf{v}_i = 1$ and $\mathbf{w}_i = -1$, yielding a feasible solution for the second problem with the same objective function value.

Now, consider a feasible solution $\mathbf{v}_i = \mathbf{w}_i$ for the second problem. If $\mathbf{v}_i - \mathbf{w}_i = 0$ then choose $\mathbf{x}_i = 0$ otherwise choose $\mathbf{x}_i = 1$, yielding a feasible solution for the first problem with the same objective function value.

Consequently, the two problems are equivalent.

5. (20 points) Formulate the vertex cover problem with weights on the vertices as a strict quadratic program (without any integer constraints) and show its relaxation to a vector program. Write the vector program as an equivalent semi-definite program.

Answer:

Vertex cover: min $\Sigma_{i,}\, x_i$
Subject to:
$x_i+x_j \ge 1$, for each edge {i,j}
$x_i \in \{0, 1\}$

This is equivalent to: min $\Sigma_{i,}\, x_i^2$
Subject to:
$x_i^2+x_j^2 \ge 1$, for each edge {i,j}
$x_i \in \{0, 1\}$

The above problem can be reformulated as a **strict quadratic program** using the solution of problem 4 as follows: min $\Sigma_{i,}\, (v_i\text{-}w_i)^2$
Subject to:
$(v_i\text{-}w_i)^2+(v_j\text{-}w_j)^2 \ge 1$, for each edge {i,j}
$v_i^2 = w_i^2 = 1$
$(v_i\text{-}w_i)^2 \ge 0$

This is transformed to the following **vector program**, where $\underline{v}_i$, $\underline{w}_i$ are vectors:
min $\Sigma_{i,}\, (\underline{v}_i\text{-}\underline{w}_i)\bullet(\underline{v}_i\text{-}\underline{w}_i)$
Subject to:
$(\underline{v}_i\text{-}\underline{w}_i)\bullet(\underline{v}_i\text{-}\underline{w}_i) + (\underline{v}_j\text{-}\underline{w}_j)\bullet(\underline{v}_j\text{-}\underline{w}_j) \ge 1$, for each edge {i,j}

$\underline{v}_i \bullet \underline{v}_i = \underline{w}_i \bullet \underline{w}_i = 1$
$(\underline{v}_i - \underline{w}_i) \bullet (\underline{v}_i - \underline{w}_i) \geq 0$

An equivalent **semi-definite program** is given below using the standard transformation. Note that v and w will become part of the same matrix. We will number the components of w after those of v. So, w_1 will become variable n+1, w_2 will become variable n+2, etc.
min $\Sigma_{i,}\ y_{ii} + y_{n+i,n+i} - 2\ y_{i,n+i}$
Subject to:
$y_{ii} + y_{n+i,n+i} - 2\ y_{i,n+i} + y_{jj} + y_{n+j,n+j} - 2\ y_{j,n+j} \geq 1$, for each edge {i,j}
$y_{ii} = 1$
$y_{ii} + y_{n+i,n+i} - 2\ y_{i,n+i} \geq 0$

Y is symmetric positive semi-definite