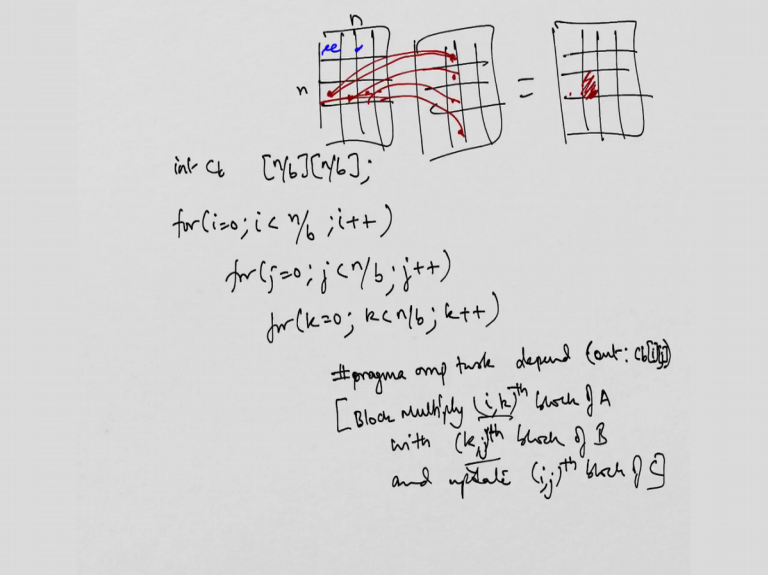
Introduction to Parallel Programming in OpenMP
Dr. Yogish Sabharwal
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi

Lecture – 36
Matrix Multiplication using tasks

(Refer Slide Time: 00:04)



So, matrix multiply right we have already seen how to do matrix multiply this is my matrix a, this is my matrix b, I multiply, then I get c, I know how to do my blocked algorithm, right. So, by now you are all familiar with this right how do I write the code for this. So, I say for i is equal to 0, i is less than number of blocks; what is the number of blocks for simplicity; let me just assume everything is of size n, right n cross n all matrices are n cross n.

So, what would be the number of blocks n by the block size, right and let me just assume that n is a multiple of b, i plus plus that is the number of blocks for j is equal to 0, j is less than n by b, j plus plus for k equal to 0, k is less than n by b k plus plus hash pragma omp task; what am I going to do in this task?

So, I am not writing; I am just telling you; what I will do over here, right, I am going to block multiply i kth block of a with k jth block of b and update i jth block of c, right, I can put this in brackets and write a structured block or I can call a function whatever, right, but that is what is important.

So, I am creating tasks out of every pair of blocks in A and B, right not for every C block. So, the same C block will get updated by multiple tasks is that point clear because I am creating a task for each i kkj.

So, this is going to update i j. So, if k is 1, this i 1 and 1 j are going to update i j, but for k equal to 2, there is another task i 2 2 j which is also going to update i j is this point, clear, there are different tasks which are updating the same block in C.

Now, C is getting updated, but there is going to be race conditions because you are updating the same variables. So, what can I do? So, you say depend out; I just need to refer to the right variable i j. So, how do I refer to i j? So, let me just define C blocks which are of size n by b times n by b. So, it is just the number of blocks. It is just a 2 dimensional area, right and now I say out let me make this names smaller, I am running out of space; let me call it C b. So, I will say out C b i j; what will be the effect of this?

Student: I thought we could only use of variable name.

You can use array elements also over here not an issue.

So, we are just using an array and I can dynamically you know refer to some element of there in that is the advantage here.

Student: (Refer Time: 03:41)

So, what does this ensure? So, I am launching this in this order, right when these tasks are launched, I am launching this first, right, then these 2, then these 2, then these 2. So, it will ensure that this multiplication happens first and which element gets updated; 3 comma 2; it will depend on the starting index, yeah, this one, right, this is what is getting updated.

So, it first; this task will get executed which multiplies these 2 then this task will get executed then this task and then this task. So, there is a dependency in these 4 tasks, but I have launched all the tasks in 1 shot; I go through this loop and I will launch all the tasks. So, all the tasks have been launched which means that these tasks may be happening in parallel to these tasks, right. So, all these tasks which have no dependencies which are not updating the same block of C can execute in parallel all together, there is no dependency between them.