processor : 31
vendor_id : GenuineIntel
cpu family : 6
model : 45
model name: Intel(R) Xeon(R) CPU E5-2670 0 @ 2.60GHz
stepping : 7
microcode : 0x710
cpu MHz : 1519.273
cache size : 20480 KB
physical id : 1
siblings : 16
core id : 7
cpu cores : 8
apicid : 47
initial apicid: 47
fpu : yes
fpu_exception: yes
cpuid level : 13
wp : yes

Seq	Original Sequential: input 1	3.355632s	1.0x
Seq	Original Sequential: input 2	0.995879s	1.0x
Seq1: Tiling/blocking n & k by 2 and 2	Seq1:input 1	5.587805s	1.011635x
Seq1: Tiling/blocking n & k by 2 and 2	Seq1:input 2	7.481525s	1.115092x

Par1: the parallel Code of Seq1	Par1: input 1	0.062205s	12.366694x
Par1: the parallel Code of Seq1	Pae1:input 2	0.186687s	12.037598x

The above is from the sbatch. The sbatch took forever and while I due think the code got an average of about 12, the few times I was able to run it, the sbatch gave me value as low as about 6 and as high as 20. The sbatch was not constant when I run it so I am also giving you the code and performance that I got on my computer, and not the sbatch. The computer information is listed above.

Seq	Original Sequential: input 1	3.355632s	1.0x
Seq	Original Sequential: input 2	0.995879s	1.0x
Seq2: tiling n&k By 2 and 4	Input 1	7.079703s	0.942374x
Seq2: tiling n&k By 2 and 4	Input 2	8.382587s	1.249185x
Par2 = Seq 2	Input 1	0.037328s	25.677987x
Par2 = Seq 2	Input 2	0.232748s	14.609980x
Seq3:tiling n&k By 2 and 2	Input 1	7.622210s	0.840400x
Seq3:tiling n&k By 2 and 2	Input 2	9.275412s	1.103454x
Par3 = Seq 3	Input 1	0.040026s	24.720659x
Par3 = Seq 3	Input 2	0.298427s	11.172139x

The performance increase due to locality is determined by the machine you run it in. The size of the cache. The dimensions by which you tile/block the loops, and the dimensions of the of the given input. Tiling/Blocking the s and r loops is pointless because the dimensions of these two are too small. The same goes for the p and q loops. Tiling/Blocking loop n and k works because there dimensions are always big. The place where you place the tiling also affects the performance of the program. The goal is to put as much into the cache.