







Sources of Overhead in Parallel Programs

- If I use two processors, shouldn't my program run 2 times faster? I used sweeping this classroom as an example and gave 4 answers: (i) Yes, (ii) Faster but less than 2 times. (iii) Faster than 2 times, (iv) Slower than the time needed by 1 sweeper.
- Overheads in parallel execution include wasted computation, communication, idling, and contention which cause degradation in performance.













In Summation Form

Computing the global sum of 16 partial sums using 16 processing elements. Σ^{j}_{i} denotes the sum of numbers with consecutive labels from *i* to *j*.













Parallel Time, Speedup, and Efficiency Example

Consider the problem of edge-detection in images. The problem requires us to apply a 3×3 template to each pixel. If each multiply-add operation takes time t_c , the serial time for an $n \times n$ image is given by $T_s = t_c n^2$.



Example of edge detection: (a) an **8** x **8** image; (b) typical templates for detecting edges; and (c) partitioning of the image across four processors with shaded regions indicating image data that must be communicated from neighboring processors to processor 1.













Effect of Granularity on Performance
 Often, using fewer processors improves performance of parallel systems. Using fewer than the maximum possible number of processing elements to execute a parallel algorithm is called <u>scaling</u> <u>down</u> a parallel system.
 A naive way of scaling down is to think of each processor in the original case as a virtual processor and to assign (or to cluster) these virtual processors equally to scale down the available processors.
 Since the number of processing elements decreases by a factor of <i>n</i> / <i>p</i>, the computation at each processing element increases by a factor of <i>n</i> / <i>p</i>.
 The communication cost should not increase by this factor since some of the virtual processors assigned to a physical processors might communicate to each other (<u>intra-processor communication</u>). This is the basic reason for the improvement from building granularity.
24







