CSE 512/CS 554 Homework Assignment 1 — Solutions

1. (a) If one worker can dig a post hole in one hour, can sixty workers dig the post hole in one minute? Why?

   Answer: No, because so many workers cannot effectively share such a confined space. In effect, there is insufficient "bandwidth" to support this level of concurrency, resulting in excessive contention for a shared resource, namely access to the hole. Moreover, the hole must be dug sequentially, in that upper portions of the hole must be dug before lower portions.

   (b) If one worker can drive a railroad spike with a sledgehammer in two minutes, can two workers with sledgehammers drive the spike in one minute? Why?

   Answer: Yes, by alternating blows, with one worker striking the spike while the other is on his backswing. In effect, each swing is a two-stage pipeline (backswing and forward swing), so that both workers can perform at full speed without interfering with each other, and hence the job is done in half the time.

   (c) What accounts for the difference between the answers to parts (a) and (b)?

   Answer: The degree of concurrency required. In part (a), two workers might well be able to halve the time by alternating access to the post hole, while in (b), sixty workers obviously could not work effectively to drive the same spike concurrently.

2. Suppose an automobile assembly line has 15 stations, each of which requires six minutes for a given car. How long does it take to build a single car? How long does it take to build 10 cars? 100 cars? 1000 cars?

   Answer: It takes 90 minutes to build one car. In general, it takes $90+6(n-1) = 84 + 6n$ minutes to build n cars. Thus, it takes 144 minutes to build 10 cars, 684 minutes for 100 cars, and 6084 minutes for 1000 cars. As the number of cars becomes very large, the time per car approaches six minutes.

3. A conventional oven can bake one potato in one hour or six potatoes in one hour. A microwave oven can bake one potato in ten minutes or six potatoes in one hour. Explain these different behaviors in terms of the physical processes involved. What analogies can you draw with serial and parallel computers?

   Answer: A conventional oven maintains a constant temperature but may expend a variable amount of energy to do so, depending on the contents of the oven. It heats a potato by a relatively slow process, diffusion of heat from the surface into the interior of the potato. Because the temperature is held constant, the potatoes cook at the same rate regardless of the number of potatoes.

   A microwave oven heats a potato by a much faster process (radiation), but uses a fixed amount of energy, which is shared by however many potatoes may be in the oven. Thus, a single potato will cook rapidly, but a larger number of potatoes will cook more slowly (indeed, they may as well be cooked sequentially).

A microwave oven is somewhat like a fast uniprocessor, whereas a conventional oven is somewhat like a multiprocessor with relatively slow processors. And true to the analogy, much of the capacity of a conventional oven is wasted much of the time.

4. Suppose that a classroom of p students, each sitting at a desk, is to compute the sum of n numbers, each of which is written on an index card. In one unit of time a given student can either add two numbers and write the sum on a card, or pass a card to another student at an *adjacent* desk (i.e., at most an arm's length away). Assuming that $n \geq p$, what is the minimum time for computing the overall sum? Specify the initial distribution of cards and the arrangement of desks assumed in your answer.

   Answer: Despite its relatively simple statment, this question has a surprisingly complicated answer, and it depends on various assumptions one may make. For example, it seems reasonable to assume that the arrangement of desks is planar (i.e., we do not permit desks to be stacked vertically!). The usual arrangement of desks in a classroom is a two-dimensional array, but other arrangements are possible, including a one-dimensional array, a ring, or a tree. We have already assumed that a single student cannot compute and communicate simultaneously, so it seems reasonable to assume that a single student can neither send nor receive more than one card in one time step. Finally, the initial distribution of cards was left unspecified. Let us assume that the cards are distributed among the students as unformly as possible, with each student having either $\lceil n/p \rceil$ or $\lfloor n/p \rfloor$, so that all students participate (which may not actually yield the best possible solution unless n is sufficiently larger than p). Further remarks on this assumption are given below.

   Under the assumptions just made, the first step in summing the numbers is for all students to compute the sums of their share of the numbers, which takes time $\lceil n/p \rceil - 1$. Now the problem is reduced to that of combining these local results to arrive at the overall sum, and this process depends on the particular arrangement of desks.

   First, consider a one-dimensional array of p desks. The students on both ends of the array pass cards containing their totals to the next students toward the center, who receive the cards, add the new figures to their initial sums, and pass the resulting cards along to the next students, etc. The final answer ends up at the center desk if p is odd, or one of the two middle desks if p is even. The total time required for this phase is $2\lfloor (p+1)/2 \rfloor$. A ring arrangement of the p desks gives the same answer, since the added connection offers no advantage for this particular problem.

A two-dimensional arrangement of the desks works similarly, except that partial sums are first collected along rows, say, using the one-dimensional algorithm, and then the final sum is collected along the resulting column, again using the one-dimensional algorithm. If the array of desks is $p_r \times p_c$, where $p \leq p_r \cdot p_c$, then the total time required for this phase is $2(\lfloor(p_r+1)/2\rfloor + \lfloor(p_c+1)/2\rfloor)$, assuming that a "center" desk is occupied in each row and column and the students are seated contiguously. The final answer ends up at the "center" of the two-dimensional array.

Finally, consider a binary tree arrangement of the desks. Let k be the number of levels in a (possibly incomplete) binary tree with p nodes, i.e., k is the smallest integer such that $p \leq 2^k - 1$. Each leaf node sends its local sum to its parent, which adds them to its own total and then passes the new partial sum to its parent, and so on up to the root of the tree, where the final total ends up. The total time required for this phase is $4(k-1)$ if p is not a power of two, and $4(k-1)-2$ otherwise.

If we relax the assumption that the cards are uniformly distributed initially, then minor additional optimizations are possible. For example, a small gain can be achieved by giving fewer cards to the students near the ends of the 1-D array, and correspondingly more cards to the students toward the center, so that the former can complete their initial sums and communicate their results earlier, while the middle students compute their extra additions when they would otherwise be idle awaiting the sums propagating from the ends. Similar optimizations can be done for the 2-D array or binary tree; in the latter case, for example, the students near the leaves would have fewer cards than those near the root. In some situations, it may even be best if some students have no cards at all.

Roughly speaking, the times required for the 1-D, 2-D, and binary tree arrangements are $\mathcal{O}((n/p)+p)$, $\mathcal{O}((n/p)+\sqrt{p})$, and $\mathcal{O}((n/p)+\log_2 p)$, respectively. Note, however, that the binary tree arrangement is not scalable, in that an increasing distance between adjacent desks is required as p increases, which would eventually exceed "arm's length" for large enough p. Alternatively, a binary tree could be embedded within another arrangement, such as a 2-D array, but then additional communication time would be required due to the resulting dilation (i.e., some desks that are logically adjacent in the tree arrangement would not be physically adjacent in the grid arrangement, so cards passed between them would have to be routed through intermediate desks).

5. Suppose that four composers who live in four different cities agree to collaborate in composing a string quartet having four movements. Discuss the relative merits of the following possible divisions of labor among the composers, particularly with reference to their load balance and their communication requirements.

(a) Composer 1 writes the first movement, *allegro*.
Composer 2 writes the second movement, *andante*.
Composer 3 writes the third movement, *scherzo*.
Composer 4 writes the fourth movement, *presto*.

(b) Composer 1 writes the 1st violin part for all four movements.
Composer 2 writes the 2nd violin part for all four movements.
Composer 3 writes the viola part for all four movements.
Composer 4 writes the cello part for all four movements.

(c) Composer 1 writes all notes played on the first string for each of the four instruments.
Composer 2 writes all notes played on the second string for each of the four instruments.
Composer 3 writes all notes played on the third string for each of the four instruments.
Composer 4 writes all notes played on the fourth string for each of the four instruments.

Answer: Method (a) will likely require the least communication. The composers will need to communicate at the beginning to plan the composition, and perhaps again at the end to tidy up at bit, but in between each composer can write his assigned movement independently. Method 1 may have a poor load balance, however, as the four movements are likely to differ in tempo, duration, and complexity, so some composers may need much more time than others to complete their movements.

Method (b) will require much more communication in order for the music to have any melodic, rhythmic, or harmonic coherence. In particular, the composers are likely to need to communicate measure by measure. On the other hand, the load balance is likely to be better, since each composer participates in each movement, although there may still be some differences in complexity of the four instrumental parts.

Method (c) will require extremely frequent communication in order to ensure that the composition is even physically playable on the instruments, much less musically sensible. It is hard to imagine that this method could produce an aesthetically attractive result. Again, the load balance might be reasonably good, since each composer participates in each movement, although there may still be some differences in the relative use of the different strings.

6. What is the *average* distance (in hops) between any two nodes (i.e., the average over all pairs of *distinct* nodes) in each of the following networks? Your answer should be expressed as a reasonably simple function of the number of nodes p.

(a) 1-D mesh

Answer: The sum of the distances from node j to each other node is

$$\begin{aligned}
\sum_{i=1}^{j-1}(j-i) + \sum_{i=j+1}^{p}(i-j) &= \sum_{i=1}^{j-1} i + \sum_{i=1}^{p-j} i \\
&= \frac{(j-1)j}{2} + \frac{(p-j)(p-j+1)}{2} \\
&= j^2 - (p+1)j + \frac{p\,(p+1)}{2}.
\end{aligned}$$

Summing this value over all p nodes, we obtain the overall total distance

$$\sum_{j=1}^{p}\left(j^2 - (p+1)j + \frac{p\,(p+1)}{2}\right) =$$

$$\frac{p\,(p+1)(2p+1)}{6} - (p+1)\frac{p\,(p+1)}{2} + p\,\frac{p\,(p+1)}{2} = \frac{p\,(p-1)\,(p+1)}{3}.$$

Dividing by the total number of node pairs, $p\,(p-1)$, we obtain the average distance between nodes, $(p+1)/3$.

(b) 1-D torus (ring)

Answer: The answer depends on whether p is odd or even. If p is odd, then the sum of the distances from any given node to each other node is

$$2\sum_{i=1}^{(p-1)/2} i = 2\,\frac{((p-1)/2)\,((p+1)/2)}{2} = \frac{(p-1)(p+1)}{4}.$$

Dividing by the number of other nodes, $p-1$, we obtain the average distance, $(p+1)/4$. If p is even, then the sum of the distances from any given node to each other node is

$$\frac{p}{2} + 2\sum_{i=1}^{(p/2)-1} i = \frac{p}{2} + 2\,\frac{((p/2)-1)(p/2)}{2} = \frac{p^2}{4}.$$

Dividing by the number of other nodes, $p-1$, we obtain the average distance, $p^2/(4(p-1))$. Thus, in either case, the average distance in a ring is about $p/4$ for large p.

(c) 2-D square mesh

Answer: Assuming that the mesh is $\sqrt{p} \times \sqrt{p}$, we can use result from part (a) to conclude that the average distance in each dimension is $(\sqrt{p}+1)/3$, for an overall average total distance of $2(\sqrt{p}+1)/3$ for all pairs of nodes whose row and column indices are both distinct. The average distance between pairs with only one distinct row or column index is $(\sqrt{p}+1)/3$. The weighted average over all distinct pairs of nodes is therefore $2\sqrt{p}/3$.

(d) hypercube

Answer: The distances for any node vary from 1 to $\log(p)$, but each distance must be weighted by the number of nodes at that distance. If $k = \log(p)$, then there are

$$\binom{k}{j} = \frac{k!}{j!\,(k-j)!}\,,$$

nodes at distance j (i.e., the number of ways of choosing j bits out of k that differ from those of the given node). Thus, the average distance is

$$\frac{\sum_{j=1}^{k} j \binom{k}{j}}{p-1} = \frac{k\,2^{k-1}}{2^k - 1} \approx \frac{k}{2} = \frac{\log(p)}{2}\,.$$

This result is intuitively obvious, since the number of nodes whose distance is greater than $\log(p)/2$ is the same as the number of nodes whose distance is less than $\log(p)/2$, and they are all symmetrically placed about this middle distance.

7. For a 3-cube, draw pictures of three *edge-disjoint* spanning trees (i.e., no edge is used at the same level in more than one tree). All three trees should have the same root and minimum height.

   Answer: See Figure 1.

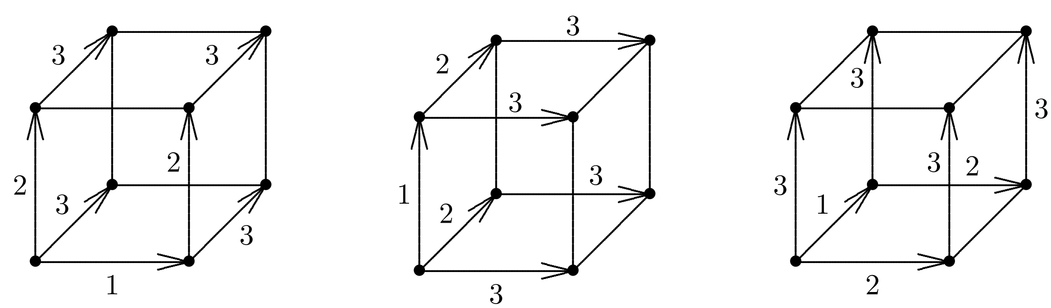

Figure 1: Three edge disjoint spanning trees for 3-cube.