Critical Path Scheduling...

(continued from page 9)

before Task j can begin. The digraph shown in the figure is for illustration only; the real digraph for such a project may be quite different.

Our goal is to construct an early-start/early-finish table (Table 1), and a late-start/late-finish table (Table 2). Table 1 gives for each task the earliest time that work on that task can begin and end. Table 2 gives for each task the latest time that work on that task can begin and end, given the constraint that the last task(s) are completed by the earliest possible completion time. Essentially, the late-start/late-finish table identifies those tasks that have some flexibility in their scheduling. To perform the entire job efficiently, each task must begin on or after its early-start time but on or before its late-start time.

Table 1

Task	T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)	T(8)
Early Start	0	0	0	13	10	25	43	43
Early Finish	13	10	14	22	25	43	47	50

Table 2

Task	T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)	T(8)
Early Start	3	0	36	16	10	25	46	43
Early Finish	16	10	50	25	25	43	50	50

Here are some hints to show how Table 1 was constructed. First, scan through the digraph in Figure 1 from left to right. Since tasks T(1), T(2), and T(3) have no predecessors, they can start at time 0, and their early-finish time is their early-start time (0 here) plus the time the task takes. How early can task T(6) start? It can not begin until both T(4) and T(5) are done. Hence, the earliest start time or T(6) is the maximum of the early finish times of T(4) and T(5), or 25 minutes. Other entries are found analogously. Observe that if each task begins right on its early-start time, and is completed in the time shown, the whole job can be completed by the largest time given in the table (50 minutes here). Notice that there is a path, T(2), T(5), T(6), T(8), such that the sum of the task times is equal to the early-finish time; such a path is called a "critical path." The tasks on this path require 50 minutes, so that 50 minutes is the best possible completion time.

Once Table 1 is complete, Table 2 can be filled in. This time, we scan through the digraph from right to left. Since the latest completion time of any task is 50 minutes, this is also the earliest completion time for the whole job. Since T(3), T(7), and T(8) have no tasks which must come after them, their late-finish times are all 50. The late-start time for Task 7 is

50(late-finish time) - 4(task time) = 46 minutes.

Other entries in Table 2 are found in a similar way. Examining the two tables, one can see that there is no flexibility in scheduling a task if and only if it lies on a critical path.

Scheduling Minibibliography

by Joseph Malkevitch

1. Coffman, E.G., *Computer & Job/Shop Scheduling*, Wiley, New York,1976. Excellent survey but now a bit dated.

2. COMAP, *For All Practical Purposes*, W.H. Freeman, 3rd Ed., 1994. An account of the critical path method and of the list-processing algorithm for scheduling machines is described. Some of the paradoxical behavior that occurs in scheduling theory is described.

3. Graham, Ronald, *The Combinatorial Mathematics of Scheduling,* Scientific American, 238(3), March, 1978, pg. 124-132. A very readable account of machine scheduling, and some of the paradoxical behavior scheduling theory sometimes produces (e.g., adding more machines can sometimes make things take longer).

4. Graham, Ronald, "Combinatorial Scheduling Theory", in Steen, L.A.(ed.), *Mathematics Today,* Springer-Verlag, 1978. A survey of elementary results about scheduling.

5. French, Simon, *Sequencing and Scheduling*, Wiley, New York, 1982. Technical but locally readable account of scheduling theory.

6. Lawler, E., "Recent Results in the Theory of Machine Scheduling", in *Mathematical Programming: The State of the Art,* Ed. A. Bachem, M. Grotschel, and B. Korte, Springer-Verlag, New York, 1983. A technical but relatively accessible survey as of about 1980.

(Continued on page 6)