In More Depth: Choosing the Right Mean

An inviting method of presenting computer performance is to normalize execution times to a reference computer, just as is done to obtain a SPEC ratio, and then take the average of the normalized execution times. However, if we average the normalized execution time values with an arithmetic mean, the result will depend on the choice of the computer we use as the reference. For example, in Figure 4.8.2, the execution times from Figure 4.4 are normalized to both A and B, and the arithmetic mean is computed. When we normalize to A, the arithmetic mean indicates that A is faster than B by 5.05/1, which is the inverse ratio of the execution times. When we normalize to B, we conclude that *B is faster by exactly the same ratio*. Clearly, both these results cannot be correct.

The difficulty arises from the use of the arithmetic mean of ratios. Instead, normalized results should be combined with the *geometric* mean. The formula for the geometric mean is

$$\sqrt[n]{\prod_{i=1}^{n} \text{Execution time ratio}_i}$$

where Execution time ratio_i is the execution time, normalized to the reference computer, for the ith program of a total of n in the workload, and

$$\prod_{i=1}^{n} a_i \text{ means the product } a_1 \times a_2 \times \ldots \times a_n$$

The geometric mean is independent of which data series we use for normalization because it has the property

$$\frac{\text{Geometric mean}(X_i)}{\text{Geometric mean}(Y_i)} = \text{Geometric mean}\left(\frac{X_i}{Y_i}\right)$$

meaning that taking either the ratio of the means or the mean of the ratios produces the same results. Thus the geometric mean produces the same relative result whether we normalize to A or B, as we can see in the bottom row of Figure 4.8.2. When execution times are normalized, only a geometric mean can be used to consistently summarize the normalized results. Unfortunately, as we show in the exercises, geometric means do not track total execution time and thus cannot be used to predict relative execution time for a workload.

The advantage of the geometric mean is that it is independent of the running times of the individual programs, and it doesn't matter which computer is used for normalization. The drawback to using geometric means of execution times is that they violate our fundamental principle of performance measurement—they do not predict execution time. The geometric means in Figure 4.8.2 suggest that for programs 1 and 2 the performance is the same for computers A and B. Yet, the arithmetic mean of the execution times, which we know is proportional to total execution time, suggests that computer B is 9.1 times faster than computer A! If we use total execution time as the performance measure, A and B would have the same performance only for a workload that ran the first program 100 times more often than the second program!

In general, no workload for three or more computers will match the performance predicted by the geometric mean of normalized execution times. The ideal solution is to measure a real workload and weight the programs according to their frequency of execution. If this can't be done, normalizing so that equal time is spent on each program on some computer at least makes the relative weightings explicit and predicts execution time of a workload with that mix. If results must be normalized to a specific computer, first summarize performance with the proper weighted measure, and then do the normalizing.

	Time on A	Time on B	Normalized to A		Normalized to B	
			A	B	A	B
Program 1	1	10	1	10	0.1	1
Program 2	1000	100	1	0.1	10	1
Arithmetic mean of time or normalized time	500.5	55	1	5.05	5.05	1
Geometric mean of time or normalized time	31.6	31.6	1	1	1	1

FIGURE 4.8.2 Execution times from Figure 4.4 normalized to each computer. The means are computed for each column. While the arithmetic means vary when we normalize to either A or B, the geometric means are consistent, independent of normalization.

Program	Floating-point operations	Execution time in seconds		
		Computer A	Computer B	Computer C
Program 1	10,000,000	1	10	20
Program 2	100,000,000	1000	100	20

FIGURE 4.8.3 Execution time and floating-point operations for two programs on three computers.

4.25 [5] <§4.3> You wonder how the performance of the three computers in Figure 4.8.3 would compare using other means to normalize performance. Which computer is fastest by the geometric mean?

4.26 [15] <§4.3> Find a workload for the two programs of Figure 4.8.3 that will produce the same performance summary using total execution time of the workload as the geometric mean of performance, as computed in Exercise . Give the workload as a percentage of executions of each program for the pairs of computers: A and B, B and C, and A and C.