A note on confidence intervals and multiple runs

There has been some confusion about how to get confidence intervals on some measure. Two relevant examples for cs115 and cs206 are the mean-time-in system of a train, or the volume of a blob being measured using Monte Carlo sampling.

Let T be one sample of a particular random variable; eg the time-in-system of *one* train, or a Boolean "true-or-false" result whether a vector $\mathbf{x}$ is inside or outside a d-dimensional blob. (In the latter case, given d-dimensional vector $\mathbf{x}$, $\phi(\mathbf{x})$ tells us if $\mathbf{x}$ lies inside the object of interest [in our case, the d-dimensional hypersphere].) We'll call this T a "point" sample: for MC sampling, calling $\phi(\mathbf{x})$ once, regardless of whether it returns "true" or "false"; for the train, it's the time-in-system of *one* train.

Say we do a total of $M = 10^9$ point-samples (ie., 10^9 trains, or 10^9 points from inside our hypercube), and S is the sum (not mean) of all the point samples. The mean is then $\bar{T} = S/M$ (and in the case of MC, we multiply by the volume of the hypercube H to get an estimate of the volume of the blob). The problem is that if we simply perform one big long run of $M = 10^9$ point-samples, we have no idea how precise $\bar{T}$ is. In statistical terms, we have no idea how much statistical "noise" is in $\bar{T}$. Formally, we don't know the *variance* of $\bar{T}$ if we only take one measurement of it. (ie., one measurement of the *mean*, even though the mean is computed from many point samples.)

There's an easy way to fix this: instead of doing one long run of M point-samples of T, split the M total point-samples into k smaller runs—called *batches*—of $n = M/k$ point-samples each. Each batch results in an independent statistical sample of $\bar{T}$. If, during batch i, the sum across all T's in the sample is S_i, then the *batch mean* of batch i is $\bar{T}_i(n) = S_i/n$. In other words, the set of batch means $\{\bar{T}_i(n)\}_{i=1}^k$ form k independent n-sample means, each being computed from $n = M/k$ point samples.

Given $\{\bar{T}_i(n)\}_{i=1}^k$, we can compute the total mean $\bar{T}(n,k) = \frac{1}{k}\sum_{i=1}^k \bar{T}_i(n)$. Note that $\bar{T}(n,k)$ is *exactly* the same number you'd get if you'd done only one batch of M point-samples, so you have not lost any information. What you have gained is that you can now compute the *sample variance* across the k batches $\bar{T}_i(n)$ where i ranges from 1 to k. (Note this is *not* the sample variance across point samples in one batch; it's the variance across batches of the batch means.) The sample variance is defined as

$$\tilde{\sigma}^2(n,k) = \frac{1}{k-1}\sum_{i=1}^{k}\left(\bar{T}_i(n) - \bar{T}(n,k)\right)^2 \tag{1}$$

The cool thing about a sample variance is that it scales linearly with the inverse of the sample size: if sample A has twice as many samples as B, then its variance will be half that of B's.

What this means is that once we compute $\tilde{\sigma}^2(n,k)$ above—the variance of the batch mean across batches—it tells us immediately how to estimate the variance of the mean across the complete M point-sample: it's just $\tilde{\sigma}^2(n,k)/k$, since it's composed of k batches n samples each. That is... the scaling of variance with batch size allows us to "magically" estimate the *variance of the mean* $\bar{T}(M)$*, even though we only have* one *sample of* $\bar{T}(M)$. So, if we define our final σ as

$$\sigma^2 = \tilde{\sigma}^2(n,k)/k, \tag{2}$$

then σ represents the standard deviation of the mean of then entire M point sample. Note that this estimate is independent of n and k separately, so long as $nk = M$. So if $M = 10^9$, it doesn't matter if you do 1 million batches of 1,000 points each, or 1,000 batches of 1 million points each, or any other combination of n and k such that $nk = M$: every one of them will give you (approximately) the same value of σ. (Try it!) This is exactly what we want: an estimate of the "error bars" for the full M point-sample run, that is independent of batch size so long as all the batches together form M point samples.[1]

From the estimate of σ above, we can compute "error bars" (or more formally, confidence intervals) on the final estimate. For example if you want 95% confidence, then the error bars must be 2σ. Translating back to the language of the project, if you want 4 digits of precision, we can interpret that as the "relative error" being 10^{-4}. In other words, at 95% confidence, we want $2\sigma/\bar{T}(n,k) < 10^{-4}$.

[1] Aside for the stats nerds: the variance of $\bar{T}(n,r)$'s goes up as n goes down, but it is compensated exactly by a decrease in $\sigma(n,k)$ as k is forced up to compensate.