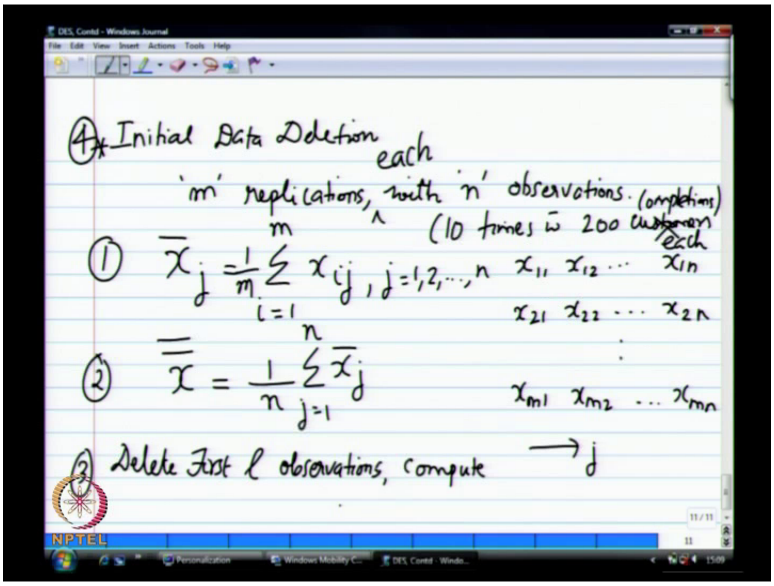
Performance Evaluation of Computer Systems

Prof. Krishna Moorthy Sivalingam
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture No. # 23
Simulations - III

(Refer Slide Time: 00:10)



This is again for whatever reason called initial data deletion. So, now what we do is we have… So, we run the same experiment m times. So, there are m replications with n observations, so in the case of m m 1, we with say until the five at cons 500 customer complete service run it. So, n say equal to 500 means I want 500 complete consumers to have completed their process at system right. So, that is why n is not time base, it is the number of observation that you make, number of sample that you collecting for whatever metric the too much. So, you have m replications each with n observation.

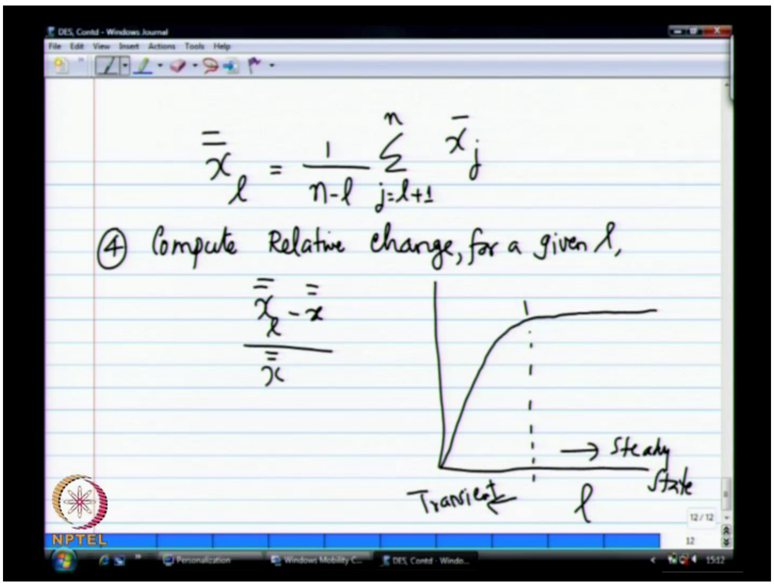
So, you know have like this big table of values right, this is first of replication first value 12 and so on up to 1, and then you have the second experiment of second replication. That is why set of values that you collected from your system so call, so this is like forced processing right. You are not really trying to delete at that instant, your processing this force pack and then trying to... See what should where relation simply drop of the steady state whether that transient state of the system. So, then now let us compute x j, which is this average. So, I am computing this is are j th observation may j goes long this axis.

(( ))

Yes no no no, each each with n observation. So, example if I say run it ten times, I run my simulation m m1 with 200 customers each 200 completions. I said 200 completion, so the first completion, second completion, that is when I do measurement completion. I will compute delay only one the customer finish the service. So that completion is when I start this metric, this particular metric at it is right. Other metrics utilization, I can complete any time, but delay I would complete only when there is the completion. So, my j basically goes along this access. So, j j is increasing order of number of observations. So, x j gone of the xij, and then we compute the overall average.

This because very looking for one value. I just want to do what is the average delay for that I got m n values choose from give may better overall system for metric. Only one value metric x is sorry x is in this case, whatever x is, x could be the average waiting service time. But every instant every observation that keep getting update say end of the first customer that average, wait average service time, average time in the system was a 1 and then goes to three and then and so on. So, each x is could be m mean also, now will come to earlier x j. You know x j is basically, this one when j equal to 1 and looking at average across all the m replications.

(Refer Slide Time: 05:44)



(( )) divided by (( )). Now, one is the value you want to find out. The delete so let l b whatever it is, so we delete first l observations. So, I can go from anywhere one time. So, what particular value of l they compute that is go to the next right (( )) no sorry they should be a bar there bar there (( )). So, if I ignore the first l observations, then I compute this x l bar. So, this is sort of like your running bit those. So, you are like in including a first l and computing what should be the mean of the remaining observation.

Then x relative change, which is for a given have, which is basically, so what is be always pass it on (( )) looking the first values. This is the overall mean this is the subset of the mean, so need not be right. So, I just should be a models actually magnitude sorry (( )) long here, so did not have that. Let us assume that there is might (( )), but essentially what we are looking for is the some point in time. Your excellent x bar should be closed each other at which means that anything will… So, when a x l bar equals x bar goes to x bar. You know that this is the steady state of the system. It anything below that there is too much variation that.

But I should and safely know that that is kind of what approach is trying to do? So, this is l and you plot this value here. This come change it is supposed sort of initially of changes will be higher. Some point in time your rate of change will be exploring down. So, the point at which this at the new of the curve, you would say this is the value that you want to delete. So, this is transient, so I am doing for every one l equals 1. My compute x l bar minus x bar or x l bar minus x bar 1. So, this is the replications, so replications is in fact this use to be sort of my assignment two last year, last I am started. This time I removed safe you that, otherwise yet actually just implement this and then start this similar are exclude that those values. So, now in this react some code that is one sync calls a deletions some of you met of seen that right. I am just doing is manually set deletion fifteen and just ignore five is what we normally do. Yes, we are doing batches of a first five batches, ten to remove. But this is little bit hopefully cleaner way of trying to find out, when the system is find pointing to steady state.

We know that tray we use a range that you know that something like n charts, something likes that which can be used to again get, what is the control that we are have? So, this will be because x bar will make other that we have extreme values. Then x bar will still will be the same, but it still a transient state. I am not looked at range based control, so that is looking at the range of values itself or is set.

So, the replication (( )) look at the range of values at we have the ranges of very large also that is also known indication of that there is wide way. That is right delay variation that is also transient state but, this would be, so again you have window of values same thing you range of looking at the mean just look at the the range of. So, if I keep changing this value l and in compute, simply the range of values for each of these and then the 1 with but, smallest range are what do you compute?

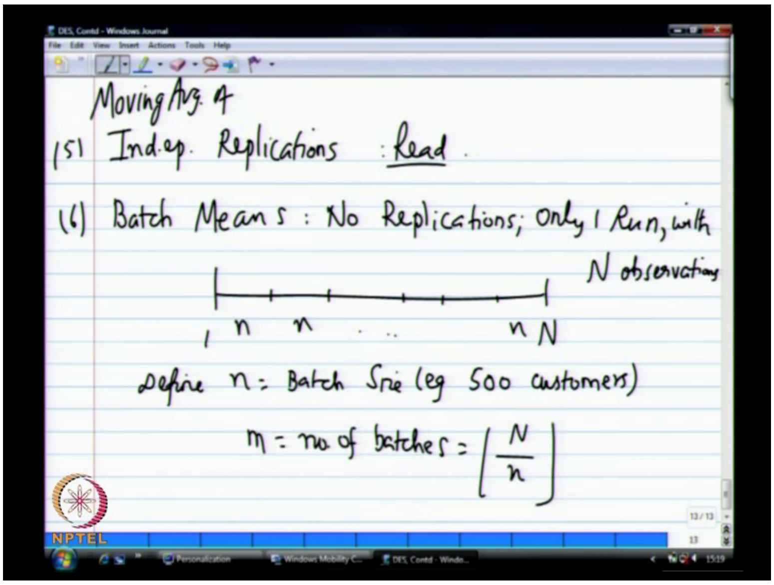
Range of the value your what a time (( )) will looking at the range control. So, variation (( )) same we can look at the range and we consider whether the ranges under control. Once we have range will under control, then we can (( )) then we can stress I am not.

(( )) goes of 2 infinitive, so this will be a n n taking (( )) up to n n. So, we lead did not come back to 0 (( )) process will be x l (( )) steady state. Then we are then what is happening (( )). So, your x bar is the actually including all the value right so.

In that case x as l double bar will be equal to x double bar, that is but again that because (( )) we hard to save that just like that moment. It should this relative change, but x bar is the including all the values and we are x l bar. Actually, sub set of the values (( )) it it could not that is saying that should (( )) actually of it be.

When l equal to with (( )) that will go to 0. Start the value of 1 was act (( )) 0 I equals to 1. The little (( )) it will be higher. It will be other way, the value is start from 1 and starts itself approaches its next every one (( )) asymptotically (( )) or that is the x l bar and the.

(Refer Slide Time: 12:12)

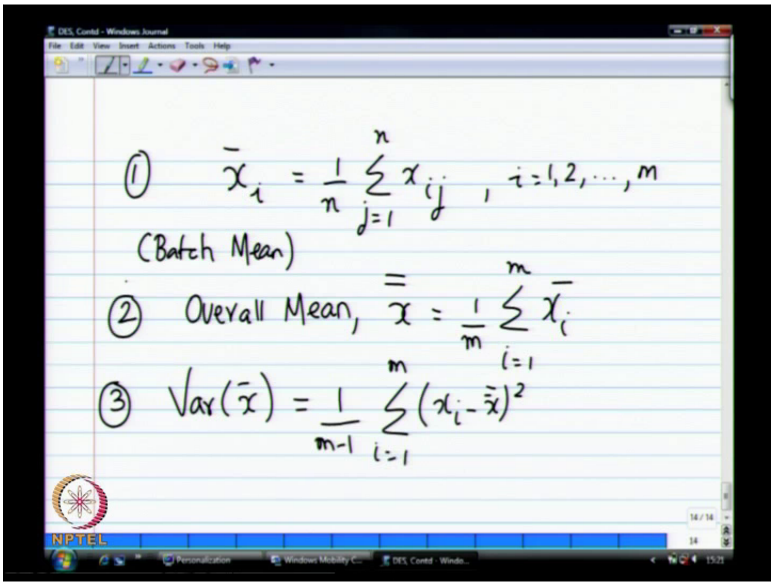


X and double bar approaching r will be x 1 x 1 bar under denominator, which case go to first one but, no sorry if you go to 0, if the x 1 bar equals plot. So, there are couple of other techniques, which is again used the same business of replications and so on, which I will kept for now. So, then you have your dependent moving average, so that two schemes are there.

One is based on independent applications and this is very similar. I just like you have to read that and other approach is vary run the simulation for the very long time but, you break it up in to smaller batches. And then look at batch by batch mean and then you try to converge based on the batch mean, which is what this is driver auto terminate actually try to do. It is based on batch means, where we have batch? You compute every five hundred customers, compute the delay, every final customer compute the mean delay and then you find that whether mean delay especially over all converging at what point. It does it start converging at we say at this point.

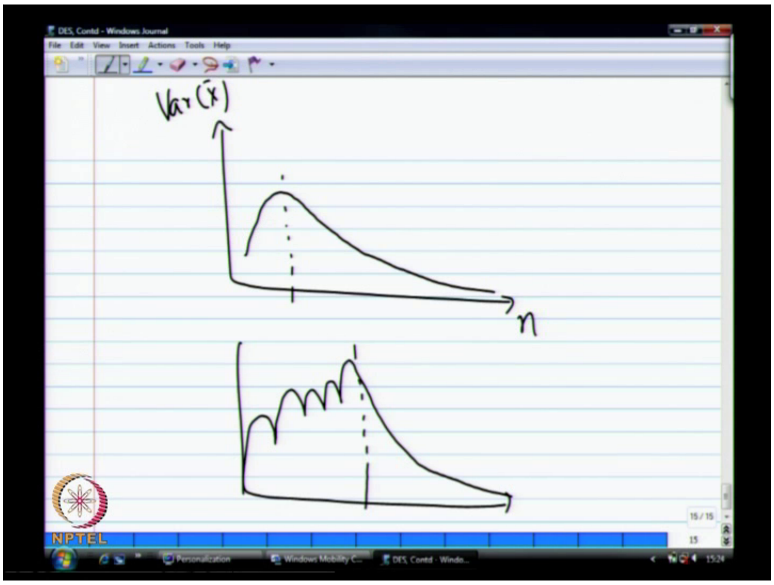
There is no need to look at that is your transient state and the same thing is also used for termination condition, where if you look at the variants across the difference means, that is going to be very small. You know that you are from models approaching steady state or there is need for running for the simulation much longer then. So, say in batch means what you do is you have the same set of n observations, you only running is there is no replications. Only one run then you have a total of n observations. Then you defined some n th to be a batch size in a break is happen to this is n and so on. So, your m is the number of batches. So, previously m was the number of replications, here m is the number of batches, which is simply then you look at inter batch means and use that (( )) so that is the batch means approach.

(Refer Slide Time: 15:16)



So, you compute the previously we had replication. So in this case, for batch one you compute the mean, the batch two, you will compute the mean and then use the this different batch mean to decide, whether there is we are has to when you stop counting the observation. When you start counting the observation in the lead once behave before that was this batch move, you want to let be just to put this rest of this down. So, your batch mean is basically, so this is x i bar like before so this is 1 over run. So, this is so here is the replication. So, I go from every five hundred customers compute the mean. We are not computing the overall mean like within before. Now, what the running that just running to this is just restart and then right compute for thus batches of a time. So, this is also called batch mean then you compute the variants right it in this means.

(Refer Slide Time: 17:29)

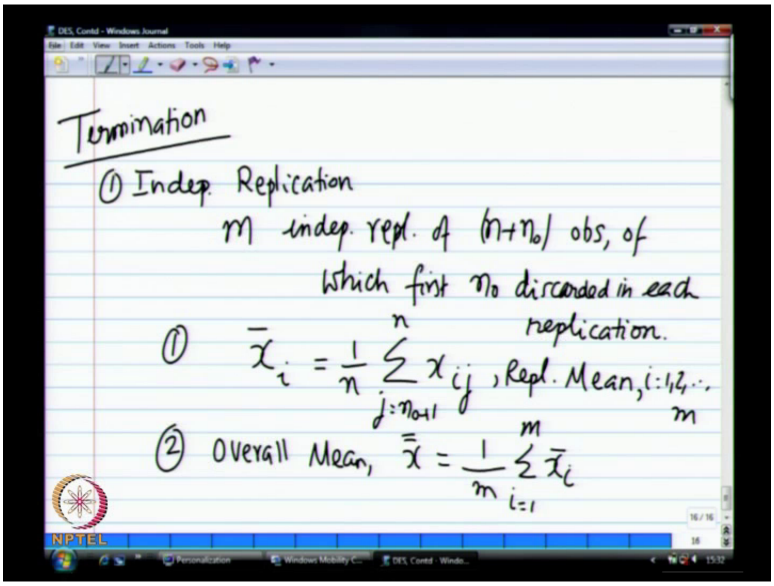


So, for a given value of n compute recomputed the variance between the difference in terms of using the batch means and then you plot again with increasing value of n. Now, you have a plot of so this is n and this is the variance of a x bar for the particular value of n. So, the point at which your variance start decreasing is technically, where transient state has finished and then steady state start.

So, basically will be going towards, if we are means all converging will be starting towards 0 variance, if you variants keep increasing then that is still where the slot of transient state system and overall point of time as you start decreasing and going towards 0. Then that is technically, where so it in idle case it should be something like this I guess, when start the decreasing and going toward 0, this is what you would like to see and say that at this point, this graph essentially your transient state finishes and steady state start again this.

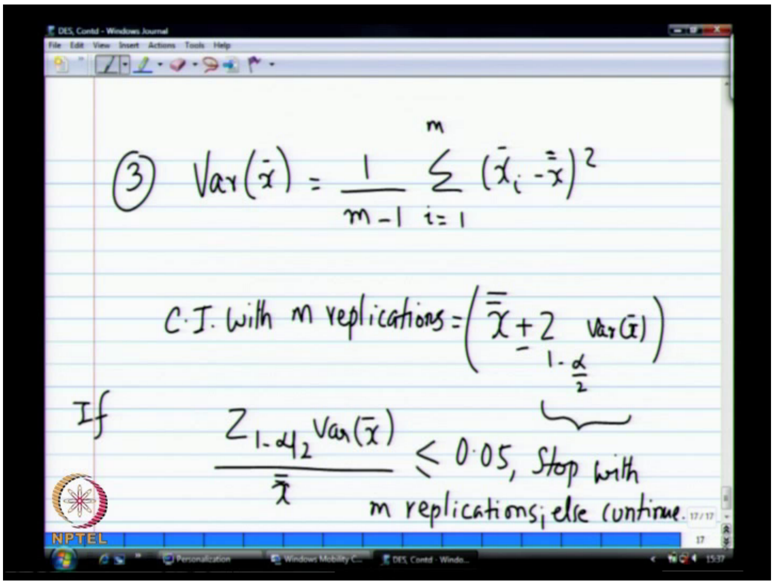
We have to way from that is the basically those possible that we could be jumping up and down, have several of this little loops very come down, the again go up, come down, go up. Finally, go to on the 0, but this is sort of ideally, where if I want to use this is the expected idle behavior system need not follow the particular behavior is possible that with they are so much of it is you have scenario like this very go up in come down. But again go up come down the variants keep changing, and finally at some point really start work here. So, this is where you should be having a transient state, this is still as this where transient state n, but if we simply say where it decreasing then I just cannot use that becomes a local (( )). So, that the question is what is the best batch size has to use compute to say that because you want delete the initial data. So, depending on the batch size, batch is very small then the depending on the batch size, you find out your, if your when is n equals to number of observation and this only one batch (( )) in which case there is no variants look at along. So, you want to essentially look at computing there.

(Refer Slide Time: 19:40)



So each batch (( )) if this is for each value of n, this is the value of n the number of value means, a batch you trying to find out the optimal value of when, so here I again we can do independent of replications are batch means. This also me thing else called regeneration. So, have m is independent replication with different seed of n plus n not observations of which approached n not are discarded. So, this is the default step between and therefore then we compute the batch the replication rise. May this is replication mean, so for replication x i this is what the normally do, where a run in going compute the mean rights such your x i bar, then you compute over all mean.

(Refer Slide Time: 22:04)



So, just we like do before one replication for an application mean, then compute the overall mean, like your run and then I compute the confidence interval. So of course, I compute the variance the usual, now the question is the which (( )) do by, what do by control. I want to basically able to stop my simulation at some point in time done. So, two parameters to control 1 is the number of replications. I want this variance essentially to be very small going to 0 that is essentially what I want.

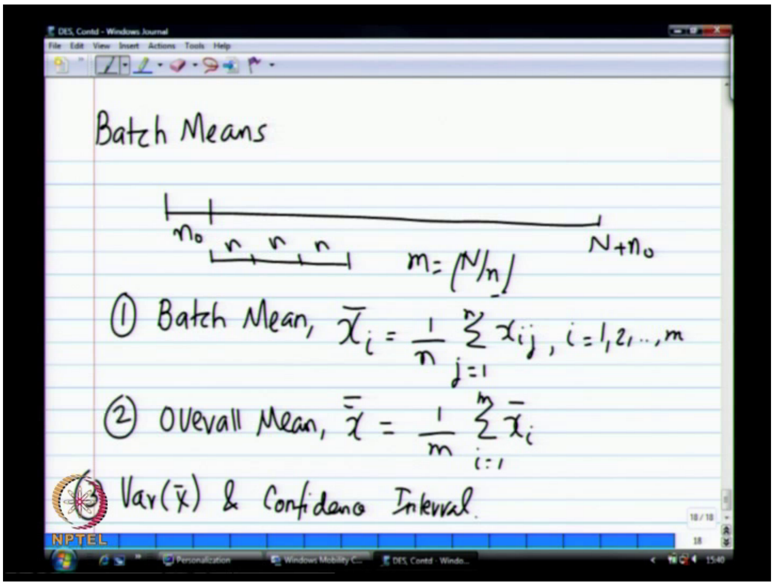
So that, I know that mean volume concluding is more or less representative of the system expected value, the other is to control the number of observation in each replication. So, this is how long should in an each application for, which is easier with this approach double of replicated. I can fix ends say run each for 1 million the completions compute values and then you start computing. So, run for first replication, second replication, third replication and so on. Keep computing this variants of x bar and then you terminate either when the variants x bar goes to 0 might never go to 0.

So, possible that it may never go to 0 in which case use you tell the system, you have to have some sort of specification on what should be the confidence interval that you are looking for, either you would say I stopped after ten replications and the confidence rate. You can compute the confidence interval with m replications could be this like within before, so x we can simply state that this is the confidence interval and then say I am done, whether you take it or not. So, this is not really termination condition, your simply telling that after x number of replications have from this conclusion or what the driver auto terminate in that system placed do is, it tries to compute the ratio of this interval, this is a confidence interval.

So, this divided by x bar that should be some specified value that you want. I want to say the 95 percent confidence, the width of the interval or one half of the width of the interval is no more than five percent of the mean. I can specify that is my termination condition, so that keep running the application until I meet that particular value. So, if I have is specification, so is if this is less than 0 point five (( )). So, the interval is less than five percent of the mean.

Therefore, you know that mean plus or minus five percent, this the confidence interval for whatever that your specifying rate for given value of alpha is specified. So, if you look at the driver auto terminate is specify 95, and if specify 0.05 that is a required with that you are looking for. If this interval values in that mean are in that range that you are specifying, then you stop else continue with number of replications that with the independent replication of the system.

(Refer Slide Time: 27:03)



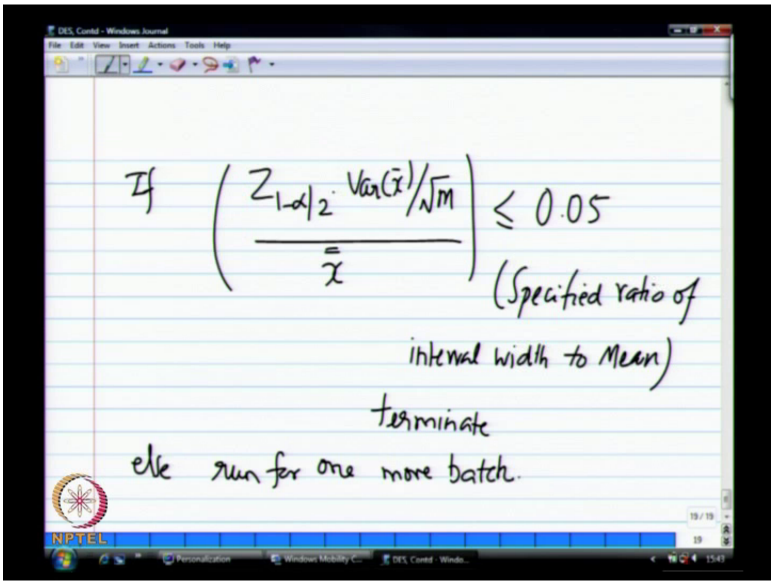
So, you need specify the stopping criteria, you have to specify what variable for what metric that you are measuring and what confidence interval that you are looking for, and what is the width of the interval that you desired achieve. If all this three or met when you say that and (( )) stop simulation that the point that is of the with m in a varying the number of replications. Assuming the each replication has enough samples to obtain some reasonable samples. So, what actually are react some tries to do is the way (( )) that the batch means the approach, where you simply run the simulation longer and longer and longer until this termination condition is met. But this auto correlation between the data is the problem.

So that, you have to ensure you have to auto correlation and index is that very small or auto co variants. So, here again I am running the system for, so n not is the initial data, which I am ignoring then from here onwards there is total of n plus n not observations and then this I am breaking up in to batches also (( )). So, I compute the batch mean, I compute the overall mean, compute the variants between the batch means and then again compute this same confidence interval based on the overall mean and the batch way.

So, we compute the batch mean, so m is simply n by so you will keep running for five hundred more observations. Keep increase the number of batches, one more batch compute x i bar, then compute the overall mean and then compute the variance of the means and the confidence interval. So, run it for five hundred ignore the first five hundred say that (( )) ignore first five hundred, then next five hundred, third five hundred and so on.

Finally, like the previous case might termination condition should be such that might the interval with the confidence interval, with ratio of that to the mean. If that is within your specify requirement, then you say I am done that is the. But that assume that is system will going to steady state, assume that with moving convergence. Sometimes you never converge try m m 1 with very large with roe greater than 1, system will not converge your queue will be especially infinite queue.

(Refer Slide Time: 30:31)



So, just keep building and your delay values will be increasing and so on. It never get to steady state, so there you can simply say system between not converge again we have. We can say that standard for some amount of time variants never. So, then again you would compute this, so this is where they may stored the square root of m last m m (( )) and here again choosing the value of n is (( )) with ten completion, two hundred completion, five hundred completion.

So, you again if you get all the data, then you can choose different values of an find out, where your auto covariance less and then use that as approved really (( )) and then report the mean, because your never reporting the mean value. So, that value of n will also make it different but, there we simply say I thing cubistically (( )) five minutes completion, this good enough to compute the mean for a particular system.

So, that is why if you look added we think you can specify the number of in your driver are terminate specify 500 is the first parameter saying number of completions, which is specifying, what the batch (( )) you can vary that. And so more larges value of when then more number of batches will have to be run take it steady state. So that, end of all this bottom line is that you are your simulation values output from simulation values they did can be wide range of values possible to depending on how is specify your end, your m, your width of the ninety five percent, 99 percent this ten percent deviation from the mean.

So, all those thing will finally, will be give you one value, so you should never stress the number that comes from is simulation as the number for the system. It is just the representative number, and it divide mostly trends is what we look for, it will try to compare two different system with how much randomness. And system bottom line is you cannot say the delay with this system is going to be exactly two seconds or two millisecond just in approximate indication of what the system performance (( )) standard deviation.

So, may square little go here, now they you there is no for if you look at the book at the stable, where a run varies from 1 to 2, 50, 6, 5, 12, and so on. Then they are also computing the co variant and then wherever that is the very small value that is saying that this is an approach appropriate value for this value (( )). We do not simply just assume that (( )) and openly nobody I will ask, so what you choose for the batch value? So, that is that little bit of arbitrary enough is there at it is.

(( )) that is for (( )) so you know you can peak a result (( )) you want putting small batch number you get better results, your system is better than the other. You can do that but, some but, has to be questioning you saying that, what is this put the value of, what is the batch size, if you used and does not make some.

If you take, then we try to calculate the confidence interval on the standard deviation. How do you feel it will, what (( )) this is for trying to terminate. So, I am trying to terminate, so if I just run one replication and if I am just computing one value of x that is also there your simply it reporting that I rand it for see, where you value will value varied depending upon the number of batches, number of customer completion let you see.

And I will keep calculating the confidence interval, you can that is the first approach (( )) be saw, but there termination condition is just simply arbitrary, you are saying that termination that the across the entire sample you saying that for the entire sample like, reduce the same termination condition not in terms of batches. So, you are batches basically size one. Batch size basically one what they, where there you will find that between, if it is single between successful values, there will be certain amount of correlation between successive values of your mean that your trying to compute that is what try to avoid by going in to larger batches. So that, you strain to say that if I go for 1024 customers at a time, then the batch mean compute between two successive batches there is not that much of correlation (( )), where is on a for customer (( )) there will be correlation between two successive customers leaving the system.

But, to find out the optimum number of customer in system so like first (( )) optimal number of customer in a batch system can handle. So, if you set look calculated delay like, 1 0 4 by 2 and next it case directly 1 0 2 4 big window then just (( )) no this batches or optimizing for the number of customers.

(( )) keep my increasing number of customers my delay should not you know exceed some particular point. So, one can calculate that mean delay, you should have this particular value. If I this batch means of approach, my suppose, my batch is 1 0 2 4. So, first batch will be 1000 before and next batch two, calculate only after thousand twenty four second thousand twenty four batch is mean is (( )) calculated. Yes, suppose my optimum number is somewhere between fifteen that six sixteen hundred, but it will directly give my 1024 or 24 it something like that no no no (( )) in your again taking of the batch size something being up. So then you can run if you that is what this is post facto, you can set on run the score for varying values of n. Such that then you find out various way, which is optimum value of n that will you are looking for can you do that and then you based on that will report a batch mean and corresponding confidence interval for that value of n.

So, what the stages that even the mean depends upon the value of n that achieved and usually we fix the n, if you want to be really making sure that there is correlation does not exist between batches, then you choose one particular value of n and for that you can report. But most of the time it one making that much of different in a win in such as small range, you one see that particular like five told 1025.

Probably, see one here to there is the big jump in to (( )) and that to really depend upon, how the system behavior is, it is the very complex system that you try to simulate it mostly just visual thinking that there is not that much of correlation between the successive. I am same thing with making assumption about exponential that Poisson and so on. Those are all this convenient as assumption that we make. So, here two we just say that this particular simulation run, and reasonably confident that the values that representing order of individual system.

We can ask some termination condition yes not do not for all system, which is known the theoretical queues at least you can calculate theoretical value, which is m m 1. You can say that this is approaching with a theoretical values (( )) have theoretical value then widely (( )) the simulation.

To verify, so when many times see in some cases it is say it is used as the verification. But many times we are running simulation almost finally we do not have really closed form solutions even if it is closed form. Closed form is how much of approximation that is not really capture by the theory itself. So, where is simulation can capture that better so it is only thought, where is the sometimes theory gives you so not always possible that (( )).

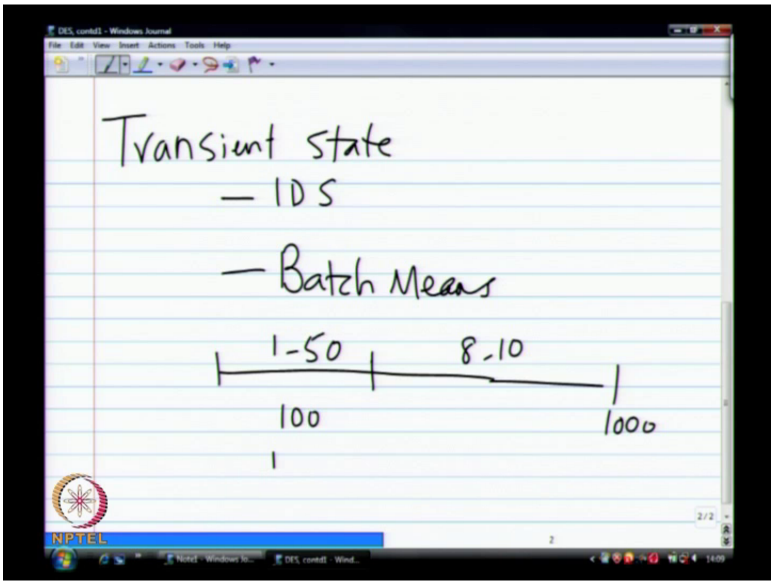
Only the set of (( )) batch size (( )) termination condition occur (( )) if you make your batch size very small (( )), your effective value of the batch mean that your depending on a small number of samples. So therefore, it you are since you are computing the overall mean it is probably. But your each batch mean is only very small set of samples. Therefore, it might not be fully representative of the system final delay that your (( )) the batch mean itself that the termination.

(( )) that termination condition we say, so we have a smaller batch size then that (( )) product will be less than 0.05 this termination addition, smaller batch size, so fast smaller batch size you are liking to the batch size is only appearing here. But in (( )) this between this increases your number of batches will proportionally increase and so how what will that of effective overall the...

(( )) within a smaller set of samples, so if you look at smaller sample in the variation between subsequence batches can be very large (( )) varying the tries of the batch will affect, so need to find out some optimal batch size, which is what I set complete take that we thing that thousand sample we could enough. So, if you want to say what is say average (( )) of student on campus again get everybody D P S are just take some hundred randomly select a student and then say that is average D P S, I think this is more or less but, you see I keep replicating the next 100, 200 and so on. So, ideally you should be saying thing five different hundred student batches, I find that the variance is more or less small across the five batches. If I good enough this is the mean of a particular class of you know particular this mean for I I T student at large that is what people try to do in their trying to do your obedient (( )) polling all those things if you go to see and then and so on.

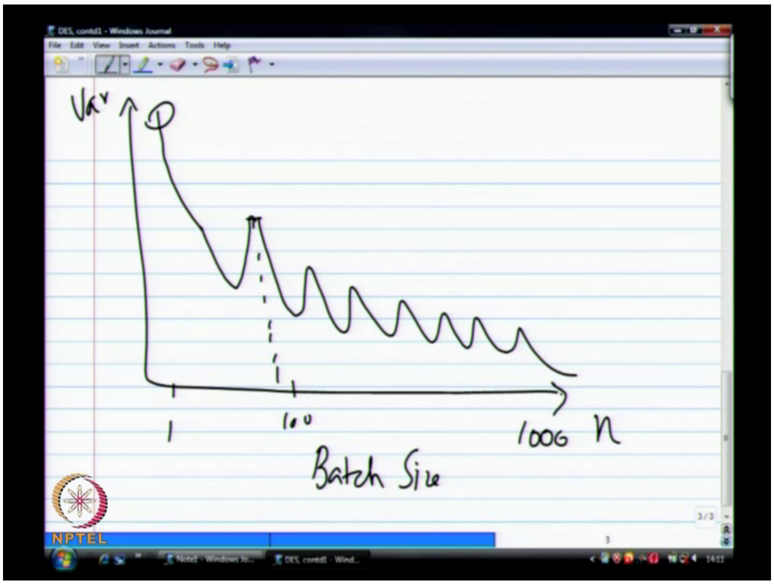
Then they poll a thousand people and say Obama (( )) got a rating of x there is as nobody as everybody there my do that over some (( )) thousand batches. But they feel that is good enough of a sample (( )) and approach that is fully that statistic focus to how they handle that, but in this case at what will happening? If you batches very small to the variance between batches will be large, if you want to capture a large enough sample size that between batches more or less representative values are there (( )). So, I went a look at those two methods (( )) solve last time for the initial data for that transient state. So, we saw this there are two or three schemes, but we covered primarily ideas there is the moving average scheme, which is very similar, which have skip and there is the batch means scheme. So, ideas with replication and batch means no replication.

(Refer Slide Time: 40:40)



So, I implemented both of them, and then I just look at numbers, so way generated numbers was I had about 1000 numbers total just random numbers, and first 100 numbers varied between. So, this is some sort of it is not exactly representative you know we can use this with m m 1 system, if you want to do, so I had 100 numbers, which was with between 1 0 1 to 50. So, there is more amount of variation then the next set of numbers, which as 8 and 10 (( )) varied in this way. So that, if you want to see that there is some sort of sharp transition from the transient state in the steady state, and so this is I just identify program, which I hope is correct. So, when I look at the th batch means, so in the batch mean idea is that you break it up in to different batch sizes last time we can of (( )) saying over is the batch cannot tell us, what are the transient state is?

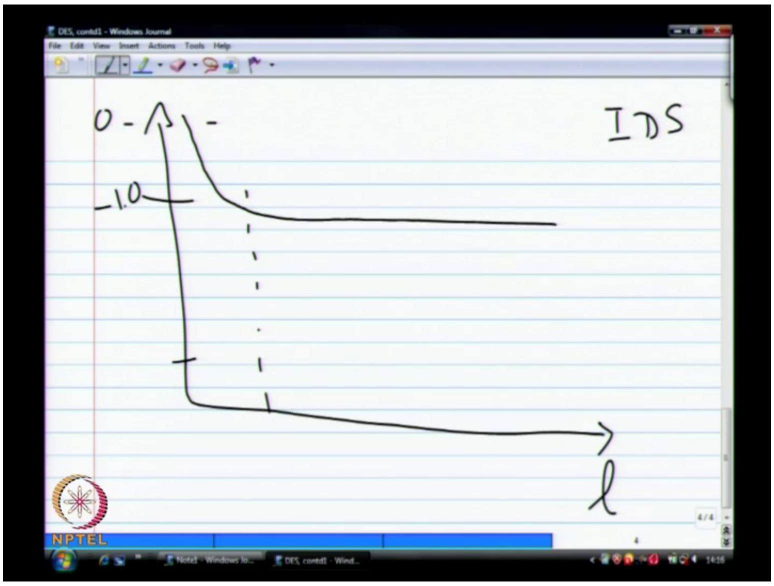
(Refer Slide Time: 41:43)



So, idea is that wherever there is definite decrease in the variance, it cannot weakly defined, wherever there is the different decrease in the variance that batch size for with it variance the change occur is considered to be the tran up to that point is that transient state. So, remember we had batch size is varying from saying that i this case I had say 1 to about 1000 which is can also see 1000 is almost just only one batch. So, I went from 2 to 500 batches or 336 batches and so on and a plotted the corresponding variance. So, this is a batch size. This diagram this e p s file is there and more (( )), now I could not load the ideas file more (( )) flows on mean. So, it definitely sort of you know start decreasing away start decreasing then that trend, if you see their again, there is an increase, the next goes up (( )) and then it is coming of (( )) you know keeps going like this have where you see so but, the only thing the noticeable.

Because, I know the values hundred you notice that the after say here it is decreasing and then increasing variance. But around this point after this exactly this is (( )) 100 also then it decreases, but never really comes back to the (( )). So, this is sort of like a does not h, if I exclude this maximum, then this is where the (( )), this is, and this we are supposed in decide that hundred is your, I am sorry scale for this particular system is started 40 or 35 are something like that, because of this depends some set of input values (( )). And it finally, stabilize that 20 and then itself keep the range that (( )). So, that is the so again is an approximation this is advantage (( )) we need to know all the data values this is actually make your decision. So, that is the batch means and neither the same thing for I d s, so in i d s, if you look added intuitively what you are trying to do is you are completing the overall mean.

(Refer Slide Time: 43:42)

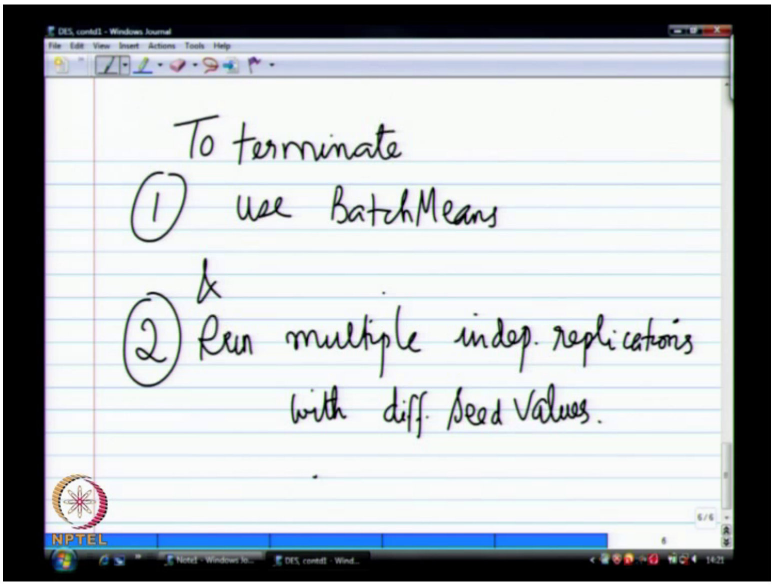


And then as you start compressing to us smaller set of values, what I did here then your overall mean will always be more than your actual mean. That is the way that my data set it could be the other way on varied initial could be small values over all then the values started increasing, which case overall when you mean will be less.

So, when I simply plotted without that magnitude, just with the signed consider, then actually started of you know some minus value something to 0, then it comes down to minus 1.1 and so on. So, this is the I D S method. If an I D S method, I am using this parameter l. So, what I do is the exclude the first tell elements, and look at the mean of the next n minus l elements. That is why x l bar and then compute the range rate of change respect to x bar overall mean in my case happen to be larger than the in, because of the fact that this values 8 to 10 was that steady state values, initial values are varying from 1 to 50.

So, really can of hot to say, so if we just plotted that value in this case would negative of course, because x bar always greater than x bar. But the nice thing was that it is sort of stabilized. So n in because of the fact that I did induce my data was cooked up. So that, I have very little variation after the 100 samples, it was all just 8, 9, 10 and so on. Therefore, that the means around wide effects include the first 50 half as 100 values all the values are between the ranges 8 to 10. Therefore, the mean will be sort of more or less stable rate of changes going to be less given, if I add out at most sample.

(Refer Slide Time: 52:15)



(( )) simulation that is correct, so we just want to exclude be want I wanted drop that first 100 values only things they did not know, where that sort of transition occur. This case it is a very sharp transition I could have been little bit more I could of made my number random generated. So, such that I have you know 1 to 100 is 0 to fifty and then I say being to 50 to 30 did not 20 and so on. But any way it was depending on a set of values you will get some sort of indication. So that, is just wanted (( )) there because in the book you simply said it those from 0 to positive, when that is not necessary like case because it depends on the values of your x bar, unless we using magnitude.

(( )) value bit (( )) we just remove the your going to remove, we are right basically, what you want to do is want to delete, first tell values you want to find the value of l, where the mean is sort of stabilizing. The change mean is of the change in mean is sort of less as you from this point on wards, because your smaller window val of values such operating and that is assuming it is again two things. Consider 1 is I am say my case as a simply had individual numbers plotted, I could also have compute the running average which is watch (( )).