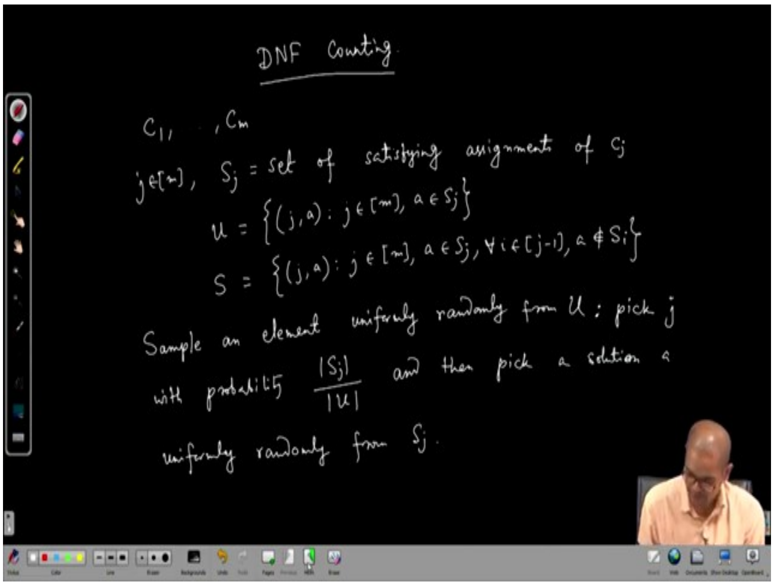
Selected Topics in Algorithm
Prof. Palash Dey
Department of Computer Science and Engineering
Indian Institute of Technology – Kharagpur

Lecture – 28
DNF Counting (Continued)

Welcome in the last class, we have seen an algorithm for DNF Counting and we claim that it is an FPRAS. So, we will do the analysis in today's class.

(Refer Slide Time: 00:37)

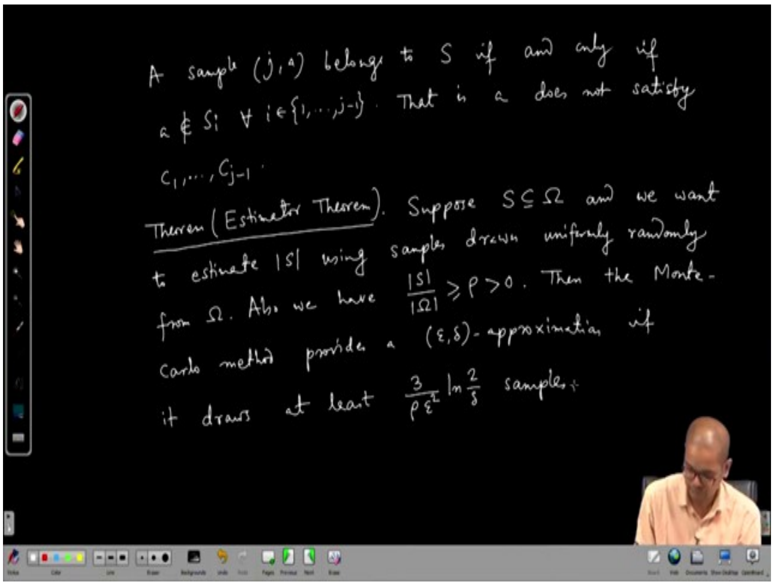


So, DNF counting; so, let us recall we designed two sets so, the clauses was $C_1, \ldots, C_m$ for $j \in [m]$. We defined S_j is the set of satisfying assignments of the clause C_j and then we define this set U which is $(j,a)\, j \in [m], a \in S_j$ this set. And you will find another set S which has which is one to one correspondence to the solution set of this DNF formula. This is j, a such that $(j,a)\, j \in [m], a \in S_j$ and for all $i \in \{1, \ldots, j-1\}$, a does not belong to S_i.

And the idea is we sample an element from u uniformly at random and check whether it belongs to S or not. How to sample U, sample element from U? For that so, sample an element uniformly randomly from U. What was the recipe? Pick j with probability proportional to S_j, by sum of cardinality S_j and sum of cardinality S_j is cardinality U.

So, pick a j proportional to S_j first and then pick a solution a uniformly randomly from S_j. Now, once a sample is picked from U, we need to check whether it belongs to yes or not that is easy.

(Refer Slide Time: 04:18)



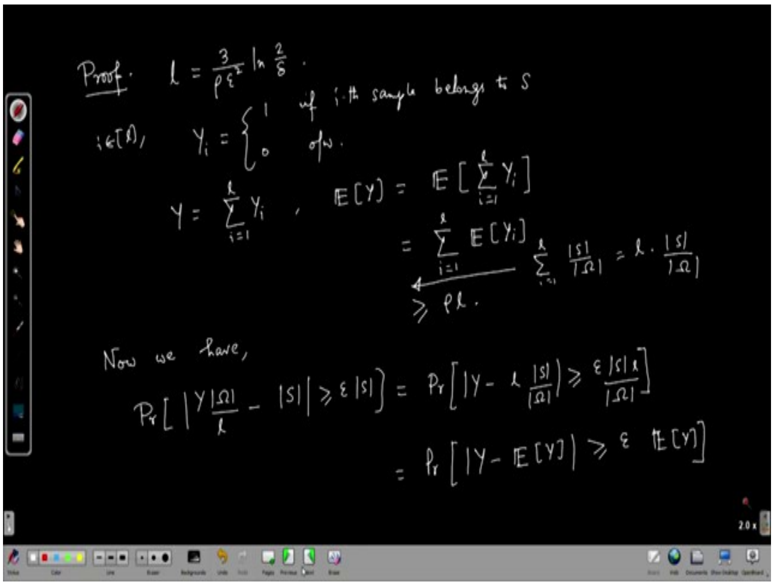
So, checking so, a sample j, a belongs to S if and only if a does not belong to S_i for all $i \in \{1, \ldots, j-1\}$. That means this assignment a does not satisfy the clauses $C_1, \ldots, C_{j-1}$ that is a does not satisfy $C_1, \ldots, C_{j-1}$ which can be checked very easily. So now, next we show that it is indeed and the output is indeed an epsilon delta approximation. So, for that we first prove a general estimator theorem and we will apply this theorem in our setting.

So, theorem is called estimated theorem. So, suppose S is a subset of some sample space and we want to estimate size of S using samples drawn uniformly randomly from Ω. And also suppose we have size of S is not too small compared to the size of Ω, it is bounded away from 0. The size of S by size of Ω this is greater than equal to ρ which is greater than 0.

It is bounded away from 0, it is not very close to 0 like our naive approach for DNF counting, where we try to sample from all set all assignments. Then the Monte Carlo method provides epsilon delta approximation. So, what is the Monte Carlo method? You draw l samples and see how many of them belongs to S? Suppose k of them belongs to S, output size of S as $\frac{k}{l}|\Omega|$, suppose cardinality omega is known.

Then this output is an (ϵ, δ) approximation if it draws at least $\frac{3}{\rho \epsilon^2} \ln\left(\frac{2}{\delta}\right)$ samples.

(Refer Slide Time: 09:40)



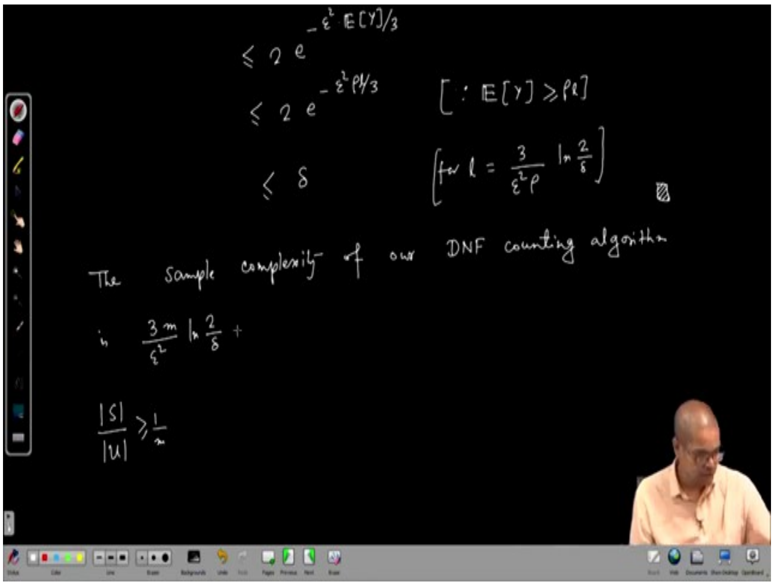
Proof, so that means l, l is the number of samples we draw. So, $l = \frac{3}{\rho\epsilon^2} \ln\left(\frac{2}{\delta}\right)$ and Y i let us define for $i \in [l]$, Y_i is 1 if i-th sample belongs to S and 0 otherwise. And define $Y = \sum_{i=1}^{l} Y_i$ it is the number of samples that fall into S. So, $E[Y] = \sum_{i=1}^{l} E[Y_i]$. Now, apply linearity of expectation, an expectation of Y_i is probability that i sample belongs to S and that is at least rho.

So, this is ρl. Now, we have probability that what is the estimate? Y many samples has fallen in S. So, $Y\frac{|\Omega|}{l}$, l many samples has been drawn. So, this is the estimate and this differs from size of S is absolute difference is greater than equal to ϵ size of S.

What is this probability? We write probability this is $|Y - l\frac{|S|}{|\Omega|}|$. This is greater than equal to $\epsilon|S|\frac{l}{|\Omega|}$. Now, see that Y is a sum of Bernoulli random variables and this is in Chernoff bound form. So, this here and line can be inserted this is summation expectation of Y is size of S by size of omega, i = 1 to l this is l times size of S by size of omega.

So now, it is it can be clearly seen that this is or let me write this is probability that mod of Y minus expectation of Y. Y deviates from it is expectation by more than $\epsilon E[Y]$. This is a two sided version.

(Refer Slide Time: 14:27)

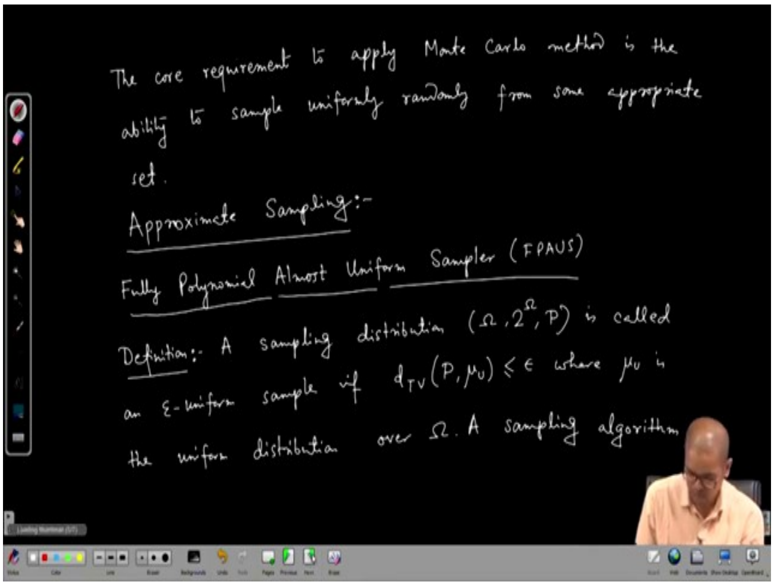


This is less than equal to $2e^{-\epsilon^2 E[Y]/3}$. So, this is $2e^{-\epsilon^2 \rho/3}$ because expectation of Y is greater than equal to ρl. And now, if we pick l equal to this that means for $l = \frac{3}{\epsilon^2 \rho} \ln\left(\frac{2}{\delta}\right)$ this is less than equal to delta. So, what is this concludes the proof of estimator theorem.

So, the sample complexity of our DNF counting algorithm is $\frac{3}{\epsilon^2 \rho}$, ρ is the bound what is the size of favourable outcome as compared to the size of sample space. Now, here for DNF counting this turns out to be this is size of S by size of U from U we are sampling so, U is like omega and S is the favourable outcome and this is greater than equal to $\frac{1}{m}$.

So, ρ is like $\frac{1}{m}$ so, this is put $\rho = \frac{1}{m}$. So, $\frac{3m}{\epsilon^2} \ln\left(\frac{2}{\delta}\right)$ so, this is the sample complexity of our DNF counting algorithm. So, our next one.

(Refer Slide Time: 17:34)

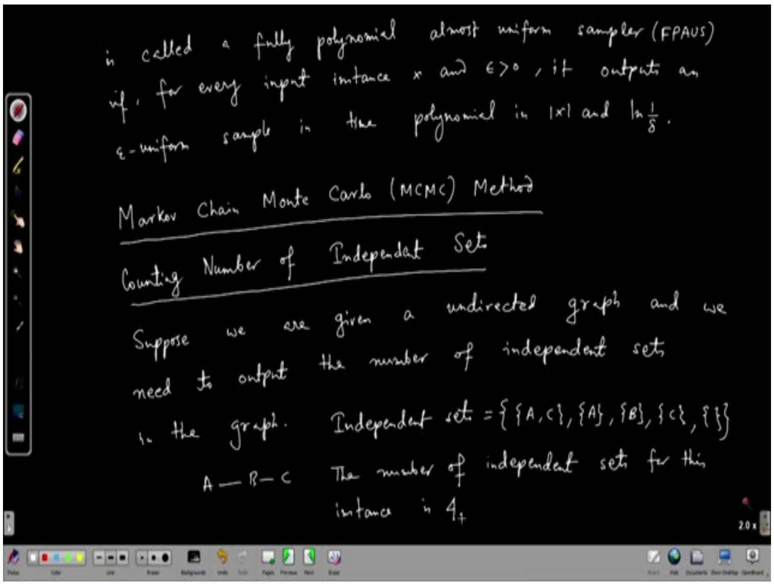


What we do is that we have seen that we to apply Monte Carlo algorithm, Monte Carlo method what we need is the ability to sample uniformly from some distribution. So that is the core thing of Monte Carlo method. So, the core requirement to apply Monte Carlo method is the ability to sample uniformly randomly from some appropriate set namely omega. So, this is our next problem that is approximate sampling.

So, often we will not be able to do draw uniform samples but we will see that almost uniform or close to uniform samples are often enough. So that is our next objective, approximate sampling. So, what is approximate sampling? So, let us call that fully polynomial almost uniform sampler FPAUS. Definition so, our sampling distribution 2^{Ω} that means set of all events 2^{Ω} and P is a probability space is called an epsilon uniform sample.

If the total variational distance between this probability P and the uniform distribution μ_U is at most epsilon where μ_U is the uniform distribution over Ω. A sampling algorithm so, this is called these are ϵ uniform samples.

(Refer Slide Time: 22:39)



A sampling algorithm is called a fully polynomial almost uniform sampler if FPAUS if for every input instance X and ϵ greater than 0 it outputs and epsilon uniform sample in time polynomial in size of X and $\ln\left(\frac{1}{\delta}\right)$. So, next what we will do is that? We will see how Markov Chain can be used effectively to design FPAUS and the corresponding technique is called Markov Chain Monte Carlo method.

So, our next topic is Markov Chain, Monte Carlo method is often abbreviated as MCMC method. So, we see this method on the problem of counting number of independent sets. So, what is this problem? What is an independent threat? So, suppose we are given an undirected graph and we need to output the number of independent sets in the graph, not necessarily maximum independent sets.

For example, if I take a graph A, B, C then the independent sets, independent set is a subset of vertices having no age bit in between them. So, for example A and C form an independent set and that is the maximum independent set but these are not the only independent set. For example all the singleton sets A, B, C they also form independent sets and also the empty set of vertices also form a independent set.

So, the number of independent sets for this instance is 4. So, in the next class we will design FPAUS and for independent sets which can sample independence uniformly at random. And using that we will see how this can be used to estimate the number of independent sets in a graph. So, we will stop here today.