Approximate Counting

Andreas-Nikolas Göbel

National Technical University of Athens, Greece

Advanced Algorithms, May 2010

Outline

1. The Monte Carlo Method
   - Introduction
   - DNFSAT Counting
2. The Markov Chain Monte Carlo Method
   - Introduction
   - From Sampling to Counting
   - Markov Chains and Mixing Time
   - Coupling of Markov Chains
   - Other Mixing Time Bounding Methods
3. Permanent
   - Permanent

Example: Estimating π

- Chose a point, (X, Y), in a 2×2 square centered at $(0, 0)$.
    - Or equiv chose Y and X independently from $[-1, 1]$.
- $Z = \begin{cases} 1 & \mathrm{if}(X, Y) \in \text{Unit Circle} \\ 0 & \textit{otherwise} \end{cases}$
- $Pr(Z = 1) = \frac{\pi}{4}$ the ratio of the area of the cicle to the area of the square.
- We run m times and let $W = \sum_{i=1}^{m} Z_i$.
- $\mathbb{E}[W] = \frac{m\pi}{4}$ and $W' = (4/m)W$ is a natural estimate for π.
- By Chernoff bound ($Pr(|X - \mu| \geq \delta\mu) \leq 2e^{\mu\delta^2/3}$, where X is the sum of independent poisson trials) we have: $Pr(|W' - \pi| \geq \varepsilon\pi) \leq 2e^{-m\pi\varepsilon^2/12}$

(ε, δ)-approximation and FPRAS

Definition ((ε, δ)-approximation)

A randomized algorithm gives an (ε, δ)-approximation for the value V if the output X satisfies:

$$Pr(|X - V| \leq \varepsilon V) \geq 1 - \delta.$$

Therefore if we choose $m \geq \frac{12 \ln(2/\delta}{\pi\varepsilon^2}$ we have an (ε, δ)-approximation for π.

Definition

A fully polynomial randomized approximation scheme for a problem is a randomized algorithm for which, given an input x and any parameters $0 < \varepsilon, \delta < 1$, the algorithm outputs an (ε, δ)-approximation to $V(x)$ in time polynomial in $1/\varepsilon$, $\ln \delta^{-1}$ and the size of the input x.

Outline of the Monte Carlo Method

Obtain an efficient approximation for a value V:

Find an efficient Process to generate a sequence of of independent and identically distributed random samples with $\mathbb{E}[X_i] = V$.

Get enough samples for an (ε, δ)-approximation for V.

The nontrivial task here is to Generate a good sequence of samples.

The Monte Carlo method is also called Monte Carlo Simulation.

A little about counting problems

- In counting problems we are interested in finding the number of different solutions for the input.
- For example in #SAT we are interested in counting the number of satisfying assignments of a given boolean formula in conjunctive normal form.
- The class of counting problems that can be solved within poly-time is FP

  The output is a number and not a yes/no answer as in decision problems
- The class that contains the problems of counting the solutions of NP problems is called $\#\mathrm{P}$.

A little about counting problems (cont.)

- $\#\mathrm{P} = \{f \mid f(x) = \mathrm{acc}_M(x)\}$, where M is a NPTM and $\mathrm{acc}_M(x) =$ number of accepting paths of M on input x.
- With an a-la-cook proof we can get that $\#$SAT is a complete problem for $\#\mathrm{P}$.
- It is interesting the fact that counting versions of problems in P may also be complete for $\#\mathrm{P}$.
    - examples: #BIPMATCHINGS, #DNFSAT, #MONSAT, #IS, #BIS.
- Note that these hard to count easy to decide problems are $\#\mathrm{P}$ complete under the poly-time Turing reduction and $\#\mathrm{P}$ is not closed under poly-time Turing reduction.
- On the other hand $\#\mathrm{P}$ is closed under poly-time many one reduction (parsimonious or karp).

A little about counting problems (concl.)

- Furthermore there are three degrees of approximability within problems of $\#\mathrm{P}$ [DGGJ'00]:
  - Solvable by an *FPRAS*:
    #PM, #DNFSAT, ...
  - AP-interreducible with #SAT:
    #SAT, #IS, #IS $|_{\mathrm{deg}(25)}$ ...
  - An Intermediate Class (AP-Interreducible with #BIS)

Note that if the counting versions of NP complete problems have an FPRAS this would imply an unexpected class collision $(\mathrm{NP} = \mathrm{RP})$.

#DNFSAT: A first approach

- Given a #DNFSAT formula F consider the following algorithm:

```
1  $X := 0$
2  For $k = 1$ to $m$ do:
       a  Generate a random assignment for the $n$ variables, chosen
          uniformly at random from all $2^n$ possible assignments
       b  If the random assignment satisfies the formula: $X := X + 1$
3  Return $(X/m)2^n$.
```

- If $X = \sum_{i=1}^{m} X_i$, where X_i independent random variables that take value 1 with probability $c(F)/n$
- By linearity of expectations: $\mathbb{E}[Y] = \frac{\mathbb{E}[X]2^n}{m} = c(F)$, where $c(F) = \#$ sat assingns.

A first approach (concl.)

- The previous approach gives an (ε, δ)-approximation of $c(F)$ when $m \geq \frac{3 \cdot 2^n \ln(2/d)}{\varepsilon^2 c(F)}$

A first approach (concl.)

- The previous approach gives an (ε, δ)-approximation of $c(F)$ when $m \geq \frac{3 \cdot 2^n \ln(2/d)}{\varepsilon^2 c(F)}$
- The above algorithm is polynomial to the size of the input (n) only if $c(F) \geq 2^n/\mathrm{poly}(n)$
- We have no guarantee of how dense $c(F)$ is in our sample space
- If $c(f)$ is polynomial in n then with high probability we must sample an exponential number of assignments before finding the fist satisfying one.

Fixing the sample space

- A sat assignment of $F = C_1 \vee C_2 \ldots C_t$ needs to satisfy at least one of the clauses.
- If clause C_i has l_i litterals there are exactly 2^{n-l_i} sat assigns.
- If SC_i is the set of assigns that sat C_i we will use as sample space the following: $U = \{(i, a) \mid 1 \leq i \leq t \quad \& \quad a \in SC_i\}$.
- $|U| = \sum_{i=1}^{t} |SC_i|$ and we want to compute $c(F) = \left|\bigcup_{i=1}^{t} SC_i\right|$.
- An assignment can satisfy more than one clause, thus we need to define a subset $S \subseteq U$ with size $c(F)$.

The Algorithm

- We provide the following algorithm for sampling

```
1. $X := 0$
2. For $k := 1$ to $m$ do:
     a. With probability $|SC_i|/|U|$ choose, uniformly at random, an assignment $a \in SC_i$
     b. If $a$ is not in any $SC_j$, $j < i$, then $X := X + 1$.
3. Return $(X/m)|U|$
```

- The above algorithm in order to estimate $c(F)$ uses $S = \{(i, a) \mid 1 \leq i \leq t, a \in SC_i, a \notin SC_j \quad \text{for} \quad j < i\}$.
  - That is for each sat assign we get exactly one pair, the one with the smalest clause index number.
- Then we estimate the ratio $|S|/|U|$ by sampling uniformly at random from U.

FPRAS for #DNFSAT

- How to uniformly sample from U:
  - We first choose the first coordinate i.
  - The i-th clause has $|SC_i|$ sat assigns, therefore we should chose i with probability proportional to $|SC_i|$, that is we chose i with probability $|SC_i|/|U|$.
  - Then we chose a sat assign uniformly at random from SC_i, that is we chose the value "T" or "F" independently and uniformly at random for each variable not in clause i.
- $Pr((i,a) \text{ is chosen }) = Pr(a \text{ is chosen } \mid i \text{ is chosen})$ $= \frac{|SC_i|}{|U|} \cdot \frac{1}{|SC_i|} = \frac{1}{|U|}$, which gives a uniform distribution.
- This algorithm is an FPRAS when $m = \lceil (3t/\varepsilon^2) \ln(2/\delta) \rceil$.

FPRAS for #DnfSat (concl.)

This algorithm is an FPRAS when $m = \lceil (3t/\varepsilon^2)\ln(2/\delta) \rceil$.

- A sat assign of F sats at most t clauses, therefore there are at most t elements (i, a) in U, corresponding to each C_i
- therefore $\frac{|S|}{|U|} \geq \frac{1}{t}$, that is the probability that each random chosen element belongs to S is at least $1/t$. $(\mathbb{E}[X] \geq 1/t)$
- $Pr(\big|\mathbb{E}[Y] - |S|\big| \geq \varepsilon\mathbb{E}[Y]) =$
  $Pr(\big|\mathbb{E}[X] - |S|m\big| \geq \varepsilon\mathbb{E}[X]m) \leq$
  $2e^{-\varepsilon^2\mathbb{E}[X]m/3} \leq \delta$

Markov Chains Reminder

- MC is a stohastic process that has states and transition probabilities.
- The transition probabilities are memoryless, i.e. they depend only on the current state of the MC.
- An ergodic (irreducible, finite and aperiodic) Markov Chain converges to a unique stationary distribution π.
  - That is the probability of a state in the MC is given by π, and it is independent from the initial state.

Overview of the MCMC method

- Define an ergodic Markov Chain with states the elements of the Sample Space.
- This MC must converge to the required Sampling Distribution.
- From any starting state X_0, and after a sufficient number of steps r the distribution of X_r will be close to the stationary.
- We use as almost independent samples $X_r, X_{2r}, X_{3r} \ldots$.
- The efficiency of MCMC method depends on:
  - How large r must be to have a good samples.
  - How fast (computationally) can we traverse between the states of the MC.

Variation Distance and Approximate Samplers

Definition (Variation Distance)

The variation distance between two probability distributions π and π' on a countable state space S is given by:
$\|\pi - \pi'\| = \frac{1}{2}\sum_{x \in S} |\pi(x) - \pi'(x)|$.

- $\|\pi - \pi'\| = \max_{A \subseteq S} |\pi(A) - \pi'(A)|$

Definition (FPAUS)

An almost uniform sampler is a randomized algorithm that takes as input x and a tolerance δ, and produces a random variable $Z \in \Omega(x)$, such that the probability distribution of Z is within variation distance ε of the uniform distribution on $\Omega(x)$. An almost uniform sampler is said to be fully polynomial if it runs in poly-time in $|x|$ and $\ln \delta^{-1}$.

Notice that the above definition can be generalized for any desired distribution.

An Example: Proper Colorings of a Graph

Theorem

Suppose we have an AUS for $k-$colorings of a graph, which works for graphs G with max degree $\Delta < k$; and suppose that the sampler has time complexity $T(n, \delta)$ (n is the number of vertices in G). Then we may construct a (ε, δ)-approximation for the number of $k-$colorings of a graph, which works for graphs with max degree bounded by Δ, and which has time complexity $\mathcal{O}\left(\frac{m^2}{\varepsilon^2} T\left(n, \frac{\varepsilon}{6m}\right)\right)$.

The idea of the proof will be presented on the whiteboard.

Markov Chain with Uniform distribution

- We need a MC with uniform stationary distribution.
- We perform a random walk in the graph of the state space.
- We add self loops to break the periodicity of MC.
- Lemma:
  For a finite space Ω and neigborhood structure $\{N(x) \mid x \in \Omega\}$ let $N = \max_{x\in\Omega} |N(x)|$. Let $M \geq N$. If the following MC is irreducible, aperiodic then the sationary distribution is the uniform distribution.

$$P_{x,y} = \begin{cases} 1/M & \text{if } x \neq y \text{ and } y \in N(x), \\ 0 & \text{if } x \neq y \text{ and } y \notin N(x), \\ 1 - N(x)/M & \text{if } x = y. \end{cases}$$

Markov Chain for the k-colorings

- For our example we will use the following Markov Chain: At each step choose a vertex v u.a.r. and a color c u.a.r. Recolor v with c if the new coloring is proper, otherwise the state of the chain remains unchainged
- This chain obviously satisfies the requirements of the previous lemma.
- We will show that the above MC is “rapidly mixing”, that is the t-step distribution closely approaches to the stationary distribution in polynomial time (of n), provided $k \leq 2\Delta + 1$.

Mixing Time

Definition

Let π be the stationary disrtibution of a Markov Chain with state space S. Let p_x^t be the distribution of the state of the chain starting at x after t steps. We define:
$\Delta_x(t) = \|p_x^t - \pi\|$.

Definition (Mixing Time)

We define $\tau_x(\varepsilon) = \min\{t \mid \Delta_x(t) \leq \varepsilon\}$ and $\tau(\varepsilon) = \max_{x \in S} \tau_x(\varepsilon)$. That is $\tau_x(\varepsilon)$ is the first step t at which the variation distance between p_x^t and the stationary distribution is less than ε, and $\tau(\varepsilon)$ is tha maximum of these values over all states x.

A chain is called rapidly mixing if $\tau(\varepsilon)$ is polynomial in $1/\varepsilon$ and the size of the problem.

The main idea

- In order to show that a chain is rapidly mixing consider the following.
- We have two copies of the same Markov Chain one of them already in the sationary distribution.
- The other starts at a state x.
- We then prove that after a short period of time they reach the same state.
- Additionally we have defined the two chains properly so that the remain in the same state right after.

Coupling

Definition (MC coupling)

A coupling of a Markov chain M_t with a state space S is a Markov chain $Z_t = (X_t, Y_t)$ on the state space $S \times S$ such that:

$$\begin{aligned} Pr(X_{t+1} = x' \mid Z_t = (x, y)) = Pr(M_{t+1} = x' \mid M_t = x); \\ Pr(X_{t+1} = y' \mid Z_t = (x, y)) = Pr(M_{t+1} = y' \mid M_t = y). \end{aligned}$$

That is, a coupling consists of two copies of the MC M running simultaneously. They are not necessarily in the same state of make the same move, instead each copy behaves exactly like the original chain.
We will use couplings that:

1. bring the two copies to the same state
2. keep them in the same state by having the two chains make identical moves once they are in the same state.

Coupling Lemma

Coupling Lemma

Let $Z_t = (X_t, Y_t)$ be a coupling for a Markov Chain M. Suppose that there exists a T such that, for every $x, y \in S$,
$Pr(X_T \neq Y_T \mid X_0 = x, Y_0 = y) \leq \varepsilon$
Then $\tau(\varepsilon) \leq T$.

That is, for any initial state, the variation distance between the distribution of the state of the chain after T steps and the stationary distribution is at most T.

Proof on board.

FPAUS for k-colorings (I)

- Consider the case of k-colorings where $k > 2\Delta + 1$
- We remind the MC on the colorings of G:
  At each step chose a vertex v u.a.r. and a color c u.a.r.
  Recolor v with c if the new coloring is proper, otherwise let the state unhanged.
- We will define a coupling of this MC.
- Let D_t be the set of vertices that have different colors in the two chains of the coupling at time t with $|D_t| = d_t$.
- Let A_t be the set of vertices that have the same color in the two chains at time t.
- Define $d'(v)$ to be the neigbours of v in D_t if $v \in A_t$.
- Similarly $d'(w)$ the neigbours of w in A_t if $w \in D_t$.

FPAUS for k-colorings (II)

- Note that $\sum_{v \in A_t} d'(v) = \sum_{w \in D_t} d'(w) = m'$.
- Coupling: If an vertex $v \in D_t$ is chosen to be recolored, we chose the same color for both chains.
- The vertex v will have the same color in both chains whenever the color chosen is different from any color on any of the neigbors of v in both copies of the MC.
- There are $k - 2\Delta + d'(v)$ such colors.
- The probability that $d_{t+1} = d_t - 1$ when $d_t > 0$ is at least: $\frac{1}{n} \sum_{v \in D_t} \frac{k-2\Delta+d'(v)}{k} = \frac{1}{kn}((k - 2\Delta)d_t + m')$.

FPAUS for k-colorings (III)

- Coupling: If a vertex $v \in A_t$ is chosen to be recolored we use the following:
- If the two vertices have one neighbour with different colors wlog assume v has color 1, and the neigbours have colors 2,3. We recolor v with 3 in the first copy and 2 in the second copy. (d_t doesn't increase)
- General case, id there are $d'(v)$ differently colored vertices around v we can couple the colors so that at most $d'(v)$ color choices cause d_t to increase. (explain)
- the probability that $d_{t-1} = d_t + 1$ is at most: $\frac{1}{n}\sum_{v \in A_t} \frac{d'(v)}{k} = \frac{m'}{kn}$.
- After some calculations (board) we prove that: $\tau(\varepsilon) \leq \frac{n(k-\Delta)}{k-2\Delta} \ln(\frac{n}{\varepsilon})$

Path Coupling

- We will explain the intuition of Path coupling with the problem #IS (it works for max deg ≤ 4).
- We start witha coupling for pairs of states thad differ in just one vertex.
- Then we extend this to a general coupling over all pairs of states.
- This technique is powerfull because it is often much easier to analyze the situation where the two states differ in a small way, than to analyze all possible ways of states.
- The extention of the coupling is a chain of states $Z_0 \dots Z_{d_t}$ where $Z_0 = X_t$ and $Y_t = Z_{d_t}$, an each successive Z_i is obtained from Z_{i-1} by either removing a vertex from $X_t \setminus Y_t$ or adding a vertex from $Y_t \setminus X_t$.
- The previous can be done for example by first removing all vertices in $X_t \setminus Y_t$ one by one and then add all the vertices in $Y_t \setminus X_t$ one by one.

Canonical Paths, CFTP

- Canonical Paths
  - View the MC as an undirected gaph with vertex set Ω and edge set $E = \{\{x, y\} \in \Omega^2 \mid P(x, y) > 0\}$.
  - For each ordered pair (x, y) we specify a canonical path γ_{xy} in the graph.
  - We choose a set of paths that avoid teh creation of edges that carry a heavy burden of paths
  - intuitively we might expect a MC to be rapidly mixing if it contains no “bottlenecks”.
- Coupling from the Past
  - We use “algorithmic coupling” to obtain sample from the exact stationary distribution.

Definition and History

- The permanent for a $n \times n$ zero one matrix is deifined by:

$$\operatorname{per}(A) = \sum_{\pi} \prod_{i=1}^{n} A_{1,\pi(i)}$$

  where the sum is over all permutations π of $\{1, 2, \ldots, n\}$.
- The best deterministic algorithm runs in time $\mathcal{O}(n2^n)$
- Although the determinant can be computed in poly time by gaussian elimination.
- It is equivalent to #BIPMATCHINGS, if A is the adjacency matrix.
- Valiant has shown that it is $\#\mathrm{P}$-complete.

FPRAS for the Permanent

- An FPRAS was given by Jerrum, Sinclair and Vigoda '02.
- It is based in a Markov Chain monte carlo method.
- The sample space of the MC consists of all perfect and near-perfect Matchings (matchings with two uncovered vertices).
- The problem is that near-perfect mathcings may outnumber the pm's by more than a polynomial factor.
- Solution: a weighting of the near perfect matchings in the stationary distribution so as to take acount the position of the holes (not matched vertices).
- Each hole pattern has equal aggregated weigt so the PM's are not dominated too much
- The mixing time of the chain is bounded by Canonical Paths Method

An alternative estimator (Simple Approach)

- The Laplace's expantion formula for the Permanent: $\mathrm{per}(A) = \sum_{j=1}^{n} a_{1j}\mathrm{per}(A_{1j})$
- The algorithm is the following:

```
If $n = 0$ then $X_A = 1$.
$W := \{j \mid a_{1j} = 1\}$.
If $W = \emptyset$ then $X_A = 0$.
else chose $J$ u.a.r. from $W$
$X_A = |W| X_{A_{1J}}$.
```

- For this estimator it holds that:
  $\mathbb{E}[X_A] = \mathrm{per}(A)$
  $\mathbb{E}[X_A^2] = \mathrm{per}^2(A)n!$. (equality for the upper triangular)
- The important result here is that for any function $\omega(n)$

$$Pr_{A_n}\left(\frac{\mathbb{E}[X_A^2]}{(\mathbb{E}[X_A])^2} > n\omega(n)\right) \to 0$$

That is the number of trials is bounded by $\mathcal{O}(n\omega(n)/\varepsilon^2)$ with high probabilty.