Coupon Collector and MAXSAT

Design and Analysis of Algorithms
Andrei Bulatov

Coupon Collector

Coupon collector. Each box of cereal contains a coupon. There are n different types of coupons. Assuming all boxes are equally likely to contain each coupon, how many boxes do you need to open before you have at least 1 coupon of each type?

Claim.

The expected number of steps is $\Theta(n \log n)$.

Proof

Phase j is the time between j and j + 1 distinct coupons.

Let X_j be the number of steps you spend in phase j.

Let X be the number of steps in total, i.e. $X_0 + X_1 + \ldots + X_{n-1}$.

$$E[X] = \sum_{j=0}^{n-1} E[X_j] = \sum_{j=0}^{n-1} \frac{n}{n-j} = n \sum_{i=1}^{n} \frac{1}{i} = nH(n)$$

Maximum 3-Satisfiability

MAX-3SAT.

Instance

A 3-CNF

Objective

Find an assignment that satisfies as many clauses as possible.

Remark: NP-hard search problem.

Simple idea:

Flip a coin, and set each variable true with probability ½, independently for each variable.

Maximum 3-Satisfiability: Analysis

Lemma

Given a 3-SAT formula with k clauses, the expected number of clauses satisfied by a random assignment is 7k/8.

Proof

Consider random variable $Z_j = \begin{cases} 1 & \text{if clause } C_j \text{ is satisfied} \\ 0 & \text{otherwise.} \end{cases}$

Let Z be the number of clauses satisfied by a random assignment

$$\begin{aligned} E[Z] &= \sum_{j=1}^{k} E[Z_j] \\ &= \sum_{j=1}^{k} \Pr[\text{clause } C_j \text{ is satisfied}] \\ &= \frac{7}{8}k \end{aligned}$$

The Probabilistic Method

Corollary

For any instance of 3-SAT, there exists a truth assignment that satisfies at least a 7/8 fraction of all clauses.

Proof

Random variable is at least its expectation some of the time.

Probabilistic method:

We showed the existence of a non-obvious property of 3-SAT by showing that a random construction produces it with positive probability!

Maximum 3-Satisfiability: Analysis

Can we turn this idea into a 7/8-approximation algorithm? In general, a random variable can almost always be below its mean.

Lemma

The probability that a random assignment satisfies ≥ 7k/8 clauses is at least 1/(8k).

Proof

Let p_j be probability that exactly j clauses are satisfied; let p be probability that at least 7k/8 clauses are satisfied.

$$\tfrac{7}{8}k = E[Z] = \sum_{j \geq 0} j p_j = \sum_{j < 7k/8} j p_j + \sum_{j \geq 7k/8} j p_j$$

$$\leq \left(\tfrac{7k}{8} - \tfrac{1}{8}\right) \sum_{j < 7k/8} p_j + k \sum_{j \geq 7k/8} p_j \leq \left(\tfrac{7}{8}k - \tfrac{1}{8}\right) \cdot 1 + kp$$

Maximum 3-Satisfiability: Analysis

Johnson's algorithm

Repeatedly generate random truth assignments until one of them satisfies at least 7k/8 clauses.

Theorem

Johnson's algorithm is a 7/8-approximation algorithm.

Proof

By previous lemma, each iteration succeeds with probability at least 1/(8k). By the waiting-time bound, the expected number of trials to find the satisfying assignment is at most 8k.

Maximum Satisfiability

Extensions.

Allow one, two, or more literals per clause.

Find max weighted set of satisfied clauses.

Theorem [Asano-Williamson 2000]

There exists a 0.784-approximation algorithm for MAX-SAT.

Theorem [Karloff-Zwick 1997, Zwick+computer 2002]

There exists a 7/8-approximation algorithm for version of MAX-3SAT where each clause has at most 3 literals.

Theorem [Håstad 1997]

Unless $P = NP$, no ρ-approximation algorithm for MAX-3SAT (and hence MAX-SAT) for any $\rho > 7/8$.

Chernoff Bound

Design and Analysis of Algorithms
Andrei Bulatov

Chernoff Bounds: Above Mean

Theorem

Suppose $X_1, \ldots, X_n$ are independent 0-1 random variables. Let $X = X_1 + \ldots + X_n$. Then for any $\mu \geq E[X]$ and for any $\delta > 0$, we have

$$\Pr[X > (1+\delta)\mu] < \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu}$$

Intuition: sum of independent 0-1 random variables is tightly centered on the mean

Chernoff Bounds: Below Mean

Theorem

Suppose $X_1, \ldots, X_n$ are independent 0-1 random variables. Let $X = X_1 + \ldots + X_n$. Then for any $\mu \le E[X]$ and for any $0 < \delta < 1$, we have

$$\Pr[X < (1-\delta)\mu] < e^{-\delta^2 \mu / 2}$$

Remark.

Not quite symmetric since it only makes sense to consider $\delta < 1$.

Load Balancing

Load balancing.

Instance

System in which m jobs arrive in a stream and need to be processed immediately on n identical processors.

Objective

Find an assignment that balances the workload across processors.

Centralized controller. Assign jobs in round-robin manner. Each processor receives at most $\lceil m/n \rceil$ jobs.

Decentralized controller. Assign jobs to processors uniformly at random.

How likely is it that some processor is assigned "too many" jobs?

Load Balancing (cntd)

Analysis.

Let X_i be the number of jobs assigned to processor i.

Let $Y_{ij} = 1$ if job j assigned to processor i, and 0 otherwise.

We have $E[Y_{ij}] = 1/n$

Thus, $X_i = \sum_j Y_{ij}$, and $\mu = E[X_i] = m/n$

Chernoff bounds with $\delta = c - 1$ yields $\Pr[X_i > c\frac{m}{n}] < \left(\frac{e^{c-1}}{c^c}\right)^{m/n}$

Let $\gamma(n)$ be number x such that $x^x = n$, and choose $c = e\,\gamma(n)$.

$$\Pr[X_i > c\frac{m}{n}] < \frac{e^{(c-1)\frac{m}{n}}}{c^{c\frac{m}{n}}} < \left(\frac{e}{c}\right)^{c\frac{m}{n}} = \left(\frac{1}{\gamma(n)}\right)^{e\gamma(n)\frac{m}{n}} < \left(\frac{1}{\gamma(n)}\right)^{2\gamma(n)\frac{m}{n}} = \frac{1}{n^{\frac{2m}{n}}}$$

Union bound implies that with probability at least 1 – 1/n no processor receives more than $e \cdot \gamma(n)\frac{m}{n} = \Theta(\log n / \log\log n \cdot \frac{m}{n})$ jobs.

Load Balancing: Many Jobs

Theorem

Suppose the number of jobs $m = 16n \ln n$. Then on average, each of the n processors handles $\mu = 16 \ln n$ jobs.

With high probability every processor will have between half and twice the average load.

Proof

Let X_i , Y_{ij} be as before.

Applying Chernoff bounds with $\delta = 1$ yields

$$\Pr[X_i > 2\mu] < \left(\frac{e}{4}\right)^{16 \ln n} < \left(\frac{1}{e^2}\right)^{\ln n} = \frac{1}{n^2}$$

Load Balancing: Many Jobs

Proof

$$\Pr[X_i < \tfrac{1}{2}\mu] < e^{-\frac{1}{2}\left(\frac{1}{2}\right)^2 (16 \ln n)} = \frac{1}{n^2}$$

Union bound implies that every processor has load between half and twice the average with probability at least 1 - 2/n.

QED