Számítástudomány gyakorlat

Péntek 08:25-09:55, LD-5-202
7. feladatsor

Az NP témakör egyik kérdése az, hogy vissza tudjuk-e vezetni a megoldás keresését eldöntési kérdések megválaszolására. A SAT esetében ha nincsen túl sok klóz, akkor a Lovász Lokális Lemma (LLL) garantálja hogy létezik megoldás, tehát ha csak ilyen „ritka” SAT problémákkal foglalkozunk akkor nincs is mire visszavezetni a keresést.

Vegyünk egy k-SAT formulát, amiben minden klóz legfeljebb d másik klózzal osztozik valamelyik változóján, és legyen $p = 2^{-k}$. Az LLL szerint ha $p(d+1)e \le 1$, akkor biztosan kielégíthető a formula. De meg is tudunk találni egy ilyen megoldást? Ezt válaszolta meg pozitív értelemben Moser és Tardos, akik 40 évvel az egzisztenciális LLL után adták az alábbi konstruktív, véletlen algoritmust.

Algorithm 1 Moser-Tardos „újrasorsoló” (resampling) algoritmusa

1: **input** klózok halmaza: C
2: inicializáljuk az összes Boole változót uniform véletlen módon $\{$Legyen a napló ℓ kezdetben üres$\}$
3: $S \leftarrow \varnothing$ (S jelöli azokat a klózokat amiket már ellenőriztünk és biztosan ki vannak elégítve)
4: **while** $S \neq C$ **do**
5:     **válasszunk** egy $c \in C \setminus S$ klózt és **ellenőrizzük** hogy ki van-e elégítve
6:     **if** kielégítve $\{\ell \leftarrow \ell 0$ – a napló végére 0-t írunk$\}$
7:         **update** $S \leftarrow S \cup \{c\}$
8:     **else** $\{\ell \leftarrow \ell 1$ – a napló végére 1-est írunk$\}$
9:         **újrasorsol** a c-beli változókat új uniform véletlen értékre állítjuk (remélhetőleg kijavítva c-t)
10:         **update** $S \leftarrow S \setminus \Gamma^+(c)$ ($\Gamma^+(c)$ jelöli azokat a klózokat akiknek van közös változójuk c-vel)
11: **end while**

A napló ℓ nem játszik szerepet az algoritmus futásában csak az analízisében.

1. Legyen egy $k-$SAT feladat megadva a klózok C halmazával az $x_1, x_2, \ldots, x_n$ Boole változókon. Futtasuk a Moser-Tardos algoritmust. Tegyük fel, hogy az 5: sorban mindig a lexikografikusan első klózt választjuk.

a)$^{\text{HF}}$ Legyen $b \in \{0,1\}^*$. Mutasd meg, hogy az algoritmus futása során minden ciklus végén igaz, hogy $\Pr(\ell = b) \le p^{\|b\|}$, ahol $\|b\|$ a Hamming súlyt, azaz az egyesek számát jelöli. Tipp: érdemes egy erősebb állítást bizonyítani: $\forall y \in \{I, H\}^n$ esetén $\Pr(\ell = b \text{ és } x = y) \le p^{\|b\|} 2^{-n}$.

b)* Bizonyítsd be, hogy az algoritmus várható értékben $\mathcal{O}(|C|)$ újrasorsolást végez mielőtt talál egy megoldást. Tipp: mutasd meg, hogy az algoritmus során potenciálisan előfordulható „megengedett” naplók ℓ számára a következő teljesül:

$$\#\{\ell \text{ potenciális napló, } \|\ell\| = r \text{ és } \ell \text{ végén 1 áll}\} \le \binom{|C| + r(d+1)}{r} \le \left(d + 1 - \frac{1}{2} + \frac{|C|}{r}\right)^r e^r.$$

A második egyenlőtlenség egy egyszerű binomiális becslés: $\binom{n}{k} < \left(\frac{en}{k} - \frac{e}{2}\right)^k$ minden $0 < k < n$ esetén, ezért elég ha az első egyenlőtlenséget bizonyítod! Érdemes először meggondolni, milyen hosszú lehet egy legfeljebb r darab egyest tartalmazó napló ami előfordulhat az algoritmus során. Ebből kijön az állítás, ugyanis annak a valószínűsége, hogy legalább r újrasorsolást végez az algoritmus az unió korláttal becsülhető.

A kurzus honlapján (`http://gilyen.hu/teaching/Szamtud_2023.html`) elérhetőek a (javított) feladatsorok és az órával kapcsolatos egyéb tudnivalók. A házi- és csillagos feladatokat a következő gyakorlat előtt tudjátok beadni, illetve csillagos feladatokat egészen addig amíg azokat „le nem lőjük” előadáson vagy gyakorlaton.