NP-kompletnost

Postoje problemi koji se ne mogu rešiti algoritmom čija vremenska složenost je O(P(n)), gde P(n) jeste polinom od dimenzije ulaza n. Važno je znati prepoznati takve probleme i za njih naći približna rešenja.
Tema ovog poglavlja su problemi za koje se ne zna da li su klasi P, specijalnije, fokus je na tzv. NP-kompletnim problemima.
Neformalno rečeno, NP klasa je klasa svih problema odlučivanja koji mogu biti rešeni Nedeterminističkim algoritmom za Polinomijalno vreme.
Za sada niko nije uspeo da pronađe NP problem koji nije u klasi P. Sam problem utvrđivanja odnosa klasa P i NP poznat je u literaturi kao **"problem P=NP "** .
Za problem X se kaže da je NP-težak ako je svaki problem iz klase NP polinomijalno svodljiv na X.
Za problem X se kaže da je NP-kompletan ako je

- X pripada klasi NP
- X je NP-težak .

Kako je S.A.Cook 1971. godine dokazao da postoje NP-kompletni problemi, to kriterijum NP kompletnosti (a koji se češće koristi u ovom kursu) za problem X glasi:

Za problem X se kaže da je NP-kompletan ako je

- X pripada klasi NP
- postoji NP-kompletan problem Y koji je polinomijalno svodljiv na X.

  Primeri problema Y (sa predavanja, iz knjige):

1. pokrivač grana
2. dominirajući skup
3. 3 SAT
4. problem klike
5. 3-obojivost

Korisni termini:

klika
: kompletan podgraf; za graf G=(V,E), klika je podgraf H u kome su svi čvorovi povezani među sobom granama iz E.

pokrivač grana
: za graf G=(V,E), pokrivač grana je skup čvorova takvih da svaka grana iz E je susedna bar jednom čvoru iz skupa

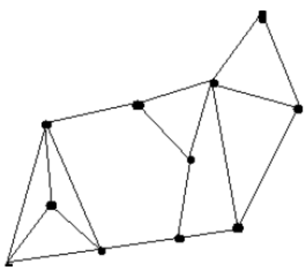
problem klika
: za dati neusmeren graf G=(V,E) i prirodan broj k ustanoviti da li G sadrži kliku veličine bar k

SAT problem
: za dati Bulov izraz S ustanoviti da li je zadovoljiv (tj. ustanoviti da li postoji dodeljivanje nula i jedinica promenljivama izraza tako da izraz ima vrednost 1)

3 SAT problem
: uprošćenje SAT problema; Bulov izraz u KNF se sastoji od klauza u kojoj svaka klauza ima tačno tri literala

3-obojivost
: za neismereni graf G=(V,E) ustanoviti da li se G može obojiti sa 3 boje (pridruživanje boja čvorovima ima pravilo da se svakom čvoru pridruži neka boja, a da su susednim čvorovima uvek pridružene različite boje)

1. a) Da li se može pronaći klika veličine 5 za graf sa slike? Sami obeležite čvorove grafa.
   b) Dokazati da je problem klika NP kompletan.



REŠENJA:

a) Za dati graf ne možemo pronaći kliku veličine 5. Ne postoji ni bar 5 čvorova stepena barem 4 (da bi to bilo moguće).

b) Teorema 11.7 iz udžbenika: Problem klika je u klasi NP, jer se za svaki pretpostavljeni podskup od *k* čvorova za polinomijalno vreme može proveriti da li je klika.

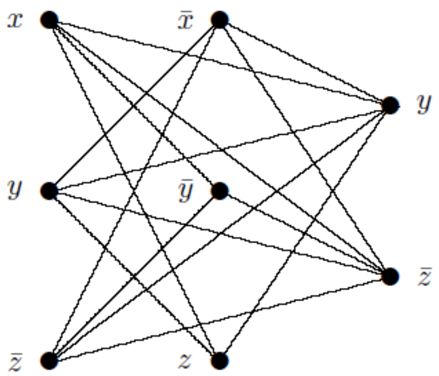
Pokazaćemo potom da se problem SAT može svesti na problem klika.

Neka je *E* proizvoljni Bulov izraz u KNF, $E = E1\ E2 \ldots Em$. Posmatrajmo, na primer, klauzu $Ei = (x + y + z + w)$.

Noj pridružujemo "kolonu" od četiri čvora, označena literalima iz *Ei* (bez obzira što se neki od njih možda pojavljuju i u drugim klauzama). Drugim rečima, graf *G* koji konstruišemo imaće po jedan čvor za svaku pojavu bilo koje promenljive. Ostaje pitanje kako povezati ove čvorove, tako da *G* sadrži kliku veličine bar *k* akko je izraz *E* zadovoljiv. Primetimo da se vrednost *k* može izabrati proizvoljno, jer je potrebno svesti problem SAT na problem klika, odnosno rešiti problem SAT koristeći algoritam za rešavanje problema klika. Naravno, algoritam za rešavanje problema klika mora da radi za svaku vrednost *k*. Ovo je važna fleksibilnost, koja se često koristi u dokazima NP-kompletnosti. U ovom slučaju za *k* ćemo izabrati vrednost jednaku broju klauza *m*.

Grane u grafu *G* mogu se zadati na sledeći način. Čvorovi iz iste kolone (odnosno čvorovi pridruženi literalima iz iste klauze) ne povezuju se granama.Čvorovi iz različitih kolona su skoro uvek povezani: izuzetak je slučaj dva čvora od kojih jedan odgovara promenljivoj, a drugi komplementu te iste promenljive. Primer grafa koji odgovara izrazu

$E = (x + y + \bar{z})\ (\bar{x} + \bar{y} + z)\ (y + \bar{z})$
prikazan je na slici . Jasno je da se *G* može konstruisati za polinomijalno vreme.



Tvrdimo da *G* ima kliku veličine bar *m*, akko je izraz *E* zadovoljiv.
<=: Uočimo da zbog konstrukcije maksimalna klika ne može imati više od *m* qvorova, nezavisno od *E*. Pretpostavimo da je izraz *E* zadovoljiv. Tada postoji takvo dodeljivanje vrednosti promenljivim, da u svakoj klauzi postoji bar jedan literal sa vrednošću 1. Čvor koji odgovara tom literalu prik uljučuje se kliki (ako ima više takvih literala, bira se proizvoljan od njih). Dobijeni podgraf jeste klika, jer jedini način da dva čvora iz različitih kolona ne budu povezana je da jedan od _ih bude komplement drugog, što je nemoguće, jer je svim izabranim literalima dodeljena vrednost 1.
=>: Obrnuto, pretpostavimo da *G* sadrži kliku veličine bar *m*. Klika mora da sadrži tačno jedan čvor iz svake kolone (jer po konstrukciji čvorovi iz iste kolone nisu povezani). Odgovarajućim literalima dodeljujemo vrednost 1. Ako na ovaj način nekoj promenljivoj nije dodeljena vrednost, to se može učiniti na proizvoljan način. Izvedeno dodeljivanje vrednosti promenljivima je neprotivrečno: ako bi nekoj promenljivoj *x* i njenom komplementu $\bar{x}$, uključenim u kliku, istovremeno bila dodeljena vrednost 1, to bi značilo da odgovarajući čvorovi (po konstrukciji) nisu povezani što je suprotno pretpostavci da su oba čvora u kliki.

2. Da li sledeći algoritam za utvrđivanje da li graf ima kliku veličine *k* jest polinomijalne vremenske složenosti:

- generisati skup od po tačno *k* čvorova (ima ih $O(n^k)$)
- proveriti kompletnost svakog podgrafa indukovanog tim podskupom
  REŠENJE:
  Klasa $O(n^k)$ je polinomijalne slozenosti po n , ali ne i po vrednosti k (koja nije fiksirana vrednost i koja je, takođe, ulazni parametar algoritma; npr. moze k=n/2),
  te ovo nije algoritam polinomijalne slozenosti, jer eksponencijalno zavisi od k

3. Zadat je regularni graf (graf čiji svi čvorovi imaju isti stepen) i prirodan broj *k*. Dokazati NP-kompletnost problema koji utvrđuje da li ovaj graf sadrži kliku veličine *k*. REŠENJE:
Moze se svesti opšti klika problem na klika problem za regularne grafove.
Neka je G=(V,E) proizvoljan graf i k proizvoljan broj.
Kontruise se pravilan graf H takav da graf G ima kliku velicine vecu ili jednaku k ako i samo ako H ima kliku velicine vecu ili jednaku k.
Neka je n broj cvorova grafa G .
Neka je d maksimalni stepen grafa G ako je taj broj paran, inace maksimalni stepen grafa G plus jedan (tj. vrednost d je uvek parna).
Za svaki cvor v grafa G stepena d(v)<d dodaje se d-d(v) novih cvorova i povezuju se sa v . Ukupan broj novih cvorova je

$\sum_{i=1..n}(d - d(v_i)) = dn - \sum_{i=1..n} d(v_i) = dn - 2|E|$

Ovaj broj je paran, jer je broj d paran.
Cilj nam je da povezemo nove cvorove tako da svi imaju stepen d.
Nove klike za sada nisu napravljene (zasto?). Skup novih cvorova se, dalje, deli na dva dela sa jednakim brojem cvorova (zasto je to moguce?). Svaki cvor iz jedne grupe povezimo sa tacno d-1 cvorova iz druge.
Tako dobijen graf je regularan i on ima kliku velicine veću ili jednaku k ako i samo ako graf G ima kliku velicine veću ili jednaku k .
Time je klika problem sveden na klika problem za regularnene grafove. Kako je klika problem NP-kompletan problem, sledi da je i klika problem za regularne grafove NP-kompletan.

4. Neka je E Bulov izraz u konjuktivnoj normalnoj formi u kojem se svaka promenljiva pojavljuje tačno jednom u pozitivnoj i tačno jednom u negativnoj formi. Opisati polinomijalni algoritam za određivanje da li je ovakav izraz zadovoljiv ili dokazati da je ovaj problem NP-kompletan. Pozitivna i negativna forma neke promenljive je na primer x i ¬x.
Uputstvo: Dati problem ima polinomijalno rešenje.

Polinomijalni algoritam može biti opisan na sledeći način:

(1) Ukoliko su tokom primene algoritma izbrisane sve klauze, onda je polazni iskaz zadovoljiv.

(2) Ukoliko postoje klauze x i ¬x, onda dati iskaz nije zadovoljiv.

(3) Ukoliko se neka promenljiva pojavljuje u samo jednoj klauzi, onda ta klauza može da bude izbrisana; idi na korak (1).

(4) Ako neka klauza C sadrži i pozitivnu i negativnu formu neke promenljive (npr. x i ¬x), onda ta klauza može da bude izbrisana; idi na korak (1).

(5) Dve klauze koje sadrže pozitivnu i negativnu formu neke promenljive mogu da budu zamenjene njihovom rezolventom po toj promenljivoj; idi na korak (1).

Navedeni algoritam je korektan i zaista ima polinomijalnu složenost

5. *k*−SAT problem je problem određivanja da li je zadovoljiv zadati logički iskaz u konjuktivnoj normalnoj formi, takav da svaki konjunkt ima tačno *k* literala (varijabli ili njihovih negacija). Poznato je da je 2-SAT problem rešiv u polinomijalnom vremenu. Svesti 2-SAT problem na 4-SAT problem. Da li opisana redukcija problema znači da je i 4-SAT problem rešiv u polinomijalnom vremenu?

REŠENJE:
k-SAT problem je problem odredjivanja da li je zadovoljiv zadati logički iskaz u konjunktivnoj normalnoj formi, takav da svaki konjunkt ima tačno *k* literala.
Neka je *E* proizvoljna instanca 2-SAT problema. Ona je oblika *E* = *E*1 *E*2... *En* pri čemu je svaki konjunkt, odnosno klauza *Ei* disjunkcija dva literala. Ideja je da svaku klauzu *Ei* zamenimo odredjenim brojem klauza takvih da svaka sadrži **tačno četiri literala.**
Neka je *Ei* = $x + y$ proizvoljna klauza iz *E* i neka je *E0* izraz dobijen iz izraza *E* tako što je klauza *Ei* zamenjena izrazom *Ei0*, gde je
$Ei0 = (x + y + a + b)(x + y + a + \bar{b})(x + y + \bar{a} + b)(x + y + \bar{a} + \bar{b})$
(*a*, *b* su nove promenljive tj. nisu sadržane u instanci *E*).
Treba dokazati: *E* je zadovoljiv akko je *E0* zadovoljiv.

(=>:) Ako je *E* zadovoljiv, onda po definiciji zadovoljivosti postoji interpretacija u kojoj je vrednost logickog izraza *E* jednaka 1.
Zbog osobine konjunkcije, to znaci da je vrednost svake klauze *Ej* jednaka 1.
Dakle, i vrednost uocene klauze *Ei* je 1, pa je vrednost bar jednog od literala *x*, *y* jednaka 1.
Zato je vrednost *Ei0*jednaka 1. To znaci da je i *E0* zadovoljiv, jer je vrednost svih klauza u *E0* jednaka 1.

(<=:) Ako je *E0* zadovoljiv, onda po definiciji zadovoljivosti postoji interpretacija u kojoj je vrednost logickog izraza *E0* jednaka 1.
Zbog osobine konjunkcije, to znaci da je vrednost svake klauze iz *E0* jednaka 1.
Dakle,i vrednost klauze *Ei0* je 1, pa je vrednost bar jednog od literala *x*, *y* jednaka 1. (U suprotnom, za $x = y = 0$ izraz *Ei0* se svodi na izraz $(a + b)(a + \bar{b})(\bar{a} + b)(\bar{a} + \bar{b})$, koji je uvek netacan, a to je kontradikcija sa polaznom pretpostavkom.)
Zato je vrednost *Ei* jednaka 1. To znaci da je i *E* zadovoljiv, jer je vrednost svih klauza u *E* jednaka 1.

Kako je Ei proizvoljno izabrana klauza, ovaj postupak (kao i dokaz) mozemo ponoviti za svako i = 1, n, pa je polazni izraz zadovoljiv akko je zadovoljiv dobijeni izraz, tj. polazni 2-SAT izraz je zadovoljiv akko je zadovoljiv 4-SAT izraz.
Opisana polinomijalna redukcija (svaka klauza se obilazi tacno jednom, i zamenjuje sa 4 nove klauze duzine 4) 2-SAT problema (koji je resiv u polinomijalnom vremenu) na 4-SAT problem ne znaci da je i 4-SAT problem resiv u polinomijalnom vremenu. Stavise, poznato je da je 4-SAT problem NP-kompletan. 4-SAT problem bi bio resiv u polinomijalnom vremenu ako bismo uspeli da ga svedemo na 2-SAT problem u polinomijalnom

vremenu.
(Teorema: Ako je L1 polinomijalno svodljiv na L2 i ako postoji algoritam polinomijalne slozenosti za L2, onda postoji i algoritam polinomijalne slozenosti za L1.)

6. *kSAT* problem je problem odredjivanja da li je zadovoljiv zadati logički iskaz u konjuktivnoj normalnoj formi, takav da svaki njegov konjunkt ima tačno k literala (varijabli ili njihovih negacija).
(a) Da li važi $1SAT \in P$?
(b) Da li važi $1SAT \in NP$?
Neka je X proizvoljan problem.
(c) Da li iz ($X \in P$ i $X \in NP$) sledi $P = NP$?
(d) Da li iz ($X \in P$ i $X \notin NP$) sledi $P \neq NP$?
(e) Da li iz ($X \in P$ i $X \notin NP$) sledi $P = NP$?
(f) Da li iz ($X \notin P$ i $X \in NP$) sledi $P \neq NP$?
(g) Da li iz ($X \notin P$ i $X \notin NP$) sledi $P = NP$?

REŠENJE
(a) Da bismo dokazali $1SAT \in P$, moramo dokazati da postoji polinomijalni algoritam koji rešava problem $1SAT$. Koristimo sledeću lemu:
Lema: Logički izraz $S = L1 \wedge \ldots \wedge Ln$, gdje su $L1, \ldots, Ln$ literali, je zadovoljiv (tj. nije kontradikcija) *akko* ne postoje $i, j \in \{1, \ldots, n\}$ takvi da je $Li = p$, a $Lj = \neg p$ za neko iskazno slovo p.
Dokaz:
=>: Dokaz izvodimo kontrapozicijom. Neka postoje $i, j \in \{1, \ldots, n\}$ takvi da je $Li = p$, a $Lj = \neg p$ za neko iskazno slovo p. Neka je $S0$ logicki izraz koji se dobija kada iz S izostavimo literale Li i Lj . Tada je:
S *ekv* $Li \wedge Lj \wedge S0$
ekv $p \wedge \neg p \wedge S0$
ekv $\bot \wedge S0$
ekv $\bot$
(*ekv* je relacija ekvivalencije nad logickim izrazima koja se uvodi sa: $S1$ *ekv* $S2$ *akko* je $S1$, $S2$ tautologija).
Dakle, S nije zadovoljiv.
<= :Neka ne postoje $i, j \in \{1, \ldots, n\}$ takvi da je $Li = p$, a $Lj = \neg p$ za neko iskazno slovo p. Vaˇzi tačno jedna od sledećih mogućnosti:
1. U izrazu S učestvuje n različitih iskaznih slova. Tada je jasno da postoji valuacija iskaznih slova u kojoj će vrednost svakog literala izraza S biti $\top$ (ako je Li oblika $\neg pi$, slovu pi dodelimo vrijednost $\bot$, inače, dodelimo mu vrednost $\top$). Dakle, S je zadovoljiv.
2. U izrazu S učestvuje $k < n$ iskaznih slova, tj. postoje $i, j \in \{1, \ldots, n\}$ takvi da je $Li = Lj$. U ovom sluˇcaju višestrukom primenom tautologije $p \wedge p \Leftrightarrow p$, izraz S svodimo na njemu ekvivalentan izraz S'' sa k literala i k razlicitih iskaznih slova. Analogno kao pod 1. postoji valuacija t takva da je $t(S'') = \top$, ali vazi S *ekv* S'' pa je $t(S) = t(S'') = \top$. Dakle, S je zadovoljiv.

Koristeci ovu lemu sastavljamo sledeci algoritam za odredjivanje da li je izraz S zadovoljiv:
1. korak: Smestimo sve literale izraza S u niz dimenzije n.
2. korak: Proverimo da li u tom nizu postoje elementi takvi da je jedan negacija drugoga. Ako postoje, algoritam daje odgovor "ne", inace, algoritam daje odgovor "da".

Slozenost prvog koraka je $O(n)$, a drugog $O(n^2)$ (svaki od n elemenata niza uporedimo sa $n - 1$ ostalih elemenata), pa je zato ovaj algoritam polinomijalan. Dakle $1SAT \in P$.

(b) Neka se u izrazu S javlja k razlicitih iskaznih slova. Postoji 2^k razlicitih nacina na koje tim slovima mozemo dodeliti vrednost. Nedeterministicki polinomijalni algoritam koji resava problem zadovoljivosti izraza S glasi ovako:
Od 2^k nacina dodele vrednosti slovima, *ND*-izborima bira se jedan (za svako iskazno slovo, jedan *ND*-izbor pridruzuje mu jednu od dve moguce istinitosne vrednosti) i proverimo kolika je vrednost izraza S pri toj valuaciji.
Ta provera vrsi se u $O(n)$ koraka, primenom istinitosnih tablica za logicke veznike $\wedge$ i $\neg$. Ako je vrednost izraza S jednaka $\top$, algoritam staje i daje odgovor "da".
Ako je izraz zadovoljiv onda postoji niz *ND*-izbora posle kojeg ce algoritam dati odgovor "da", a inace ne postoji takav niz.
Dakle, postoji i nedeterministicki polinomijalni algoritam koji resava $1SAT$ problem, pa vazi $1SAT \in NP$.
(Naravno, mogli smo i (i to uvek, ne samo u ovom slucaju) da primenimo vec poznati deterministicki polinomijalni algoritam za konstrukciju nedeterministickog polinomijalnog algoritma.)

(c) Iz ($X \in P$ i $X \in NP$) ne mozemo izvesti cinjenicu da za svaki problem $Y \in NP$ vazi $Y \in P$. Kada bi X problem bio NP-tezak, onda bismo to mogli da izvedemo. Dakle, kako je X proizvoljan problem, iz ($X \in P$ i $X \in NP$) ne sledi $P = NP$.

(d) Iskaz "$X \in P$ i $X \notin NP$" je za svako X netacan, jer je $P \subseteq NP$, pa je zato iskaz "iz ($X \in P$ i $X \notin NP$) sledi $P \neq NP$ " formalno tacan, na osnovu definicije semantike Tarskog, na osnovu koje je: Iskaz "iz A sledi B" je tacan akko je iskaz A netacan ili iskaz B tacan.
(Napomena: implikacija " iz ($X \in P$ i $X \notin NP$) sledi $P \neq NP$ " je tacna, ali to jos uvek ne znaci da vazi $P \neq NP$, jer ne postoji problem X za koji vazi $X \in P$ i $X \notin NP$).

(e) Odgovor je "da" analogno kao pod (d)

(f) Odgovor je ”da”, jer ako postoji element skupa NP, koji ne pripada skupu P, onda se skupovi P i NP razlikuju (bar po tom elementu).

(g) Slicno kao pod (d, a), cinjenica (X $\notin$ P i X $\notin$ NP) nam nije dovoljna da zakljucimo da svaki Y $\in$ NP mora pripadati skupu P. Zato iz (X $\notin$ P i X $\notin$ NP) ne sledi P = NP.

7. Dat je prirodan broj *k* i neusmeren graf G=(V,E) čiji svi čvorovi imaju paran stepen. Dokazati da je NP-kompletan problem koji ustanovljuje da li u G postoji skup C sa ne više od k čvorova tako da svaka grana iz E je susedna bar jednom od čvorova iz skupa C.

REŠENJE
Dokaz se zasniva na redukciji sa običnog problema pokrivač grana.
Neka je $G = (V,E)$ proizvoljni neusmereni graf. Neka je *U* skup čvorova neparnog stepena u *G*. Kako je ukupna suma stepena svih čvorova parana, to je i |U| paran broj (vidi sličan zadatak sa vežbi)
Uvedimo novi graf dodavanjem tri nova čvora *x*, *y*, i *z*, međusobno povezana u trougao. Dodatno, povežimo čvor *x* sa svim čvorovima iz *U*.
Tako dobijamo novi graf *G’*, čiji svi čvorovi imaju paran stepen (jer |U| je paran broj, te su parni brojevi i: d(x)=2+|U|, d(y)=2, d(z)=2, a svaki čvor neparnog stepena iz U je dobio +1 granu, te je i njihov stepen paran broj).
Dakle, pokažimo da ako graf G’ ima pokrivač grana, onda uklanjem nekih grana možemo dobiti i pokrivač za graf G.
Tvrđenje: Graf *G’* ima pokrivac grana velicine *k* ⇔ *G* ima pokrivac grana velicine *k* - 2.
<=:
od proizvoljnog pokrivača *G* veličine *k*-2 se dodavanjem cvorova *x* i *y* dobija pokrivac grana grafa *G’* velicine *k*. Zato sto su novododate grane grafa G’ (tj. grane (x,y), (x,z), (y,z), (x,w) gde w je iz U) susedne čvorovima x, y.
=>: ako u *G’* postoji pokrivac grana velicine *k*, onda taj pokrivac mora da sadrzi bar dva od tri cvora *x*, *y* i *z* da bi bile pokrivene novododate grane (tj. grane (x,y), (x,z), (y,z), (x,w) gde w je iz U)
Tada je i skup od *k* čvorova koji se dobija zamenom ta dva čvora sa *x*, *y* takođe pokrivač grana *G’*
Pošto *x*, *y* ne pokrivaju ni jednu granu iz *G*, njihovim uklanjanjem iz skupa dobija se pokrivač grana grafa *G* veličine k-2.