محمدمهدی مرادی

Subject: ۴۰۳۱۰۶۶۸۱ Date:

-1

$A \in P$ و $P \subseteq NP \Rightarrow A \in NP$ (I)

A تهی نیست پس رشته‌ای وجود دارد که می‌پذیرد (x)

$\Sigma^* \neq A$ پس رشته‌ای وجود دارد که نمی‌پذیرد (y)

حالا باید یک زبان $L \in NP$ را در زمان چندجمله‌ای به A reduce کنیم

چون $P=NP$ پس $L \in P$ پس TM قطعی M وجود دارد تا در زمان چندجمله‌ای

عضویت یک رشته را در L تصمیم بگیرد

$$w \in L \xrightarrow{f} f(w) \in A$$

f را اینگونه می‌سازیم:

اگر $w \in L$ آنگاه $f(w)=x$ و اگر $w \notin L$ آنگاه $f(w)=y$

بررسی عضویت w در L زمان چندجمله‌ای دارد تولید خروجی هم $O(1)$

$w \in L \Rightarrow f(w) \in A$ و $w \notin L \Rightarrow f(w) \notin A$

پس A یک NP-complete است

DAT

Subject: Date:

۲- الف) TM زبان L_1 می‌شود M_1 و L_2 می‌شود M_2

P:

M' را اینگونه می‌سازیم:

M_1 را روی w اجرا می‌کنیم. اگر بپذیرد پس M' می‌پذیرد اگر نه M_2 را روی w اجرا می‌کند و اگر بپذیرد پس M' می‌پذیرد اگر نه M' نمی‌پذیرد

$O(n^{k_1}+n^{k_2}) \Rightarrow P \Rightarrow L_1 \cup L_2 \in P$

NP:

به صورت تصادفی M_1 یا M_2 را روی w اجرا می‌کنیم اگر بپذیرد، می‌پذیریم

$O(\max(n^{k_1}, n^{k_2})) \Rightarrow P \Rightarrow L_1 \cup L_2 \in NP$

ب) مانند بخش الف فقط به جای or اینجا and داریم

P:

پس اگر M_1 بپذیرد و M_2 بپذیرد آنگاه M' می‌پذیرد اگر نه نمی‌پذیرد

$O(n^{k_1}+n^{k_2}) \Rightarrow P \Rightarrow L_1 \cap L_2 \in P$

NP: M_1 را روی w اجرا می‌کنیم اگر روی یک شاخه پذیرفته شد M_2 را روی w اجرا می‌کنیم و اگر پذیرفت M' هم می‌پذیرد اگر نه نمی‌پذیرد

$O(n^{k_1}+n^{k_2}) \Rightarrow P \Rightarrow L_1 \cap L_2 \in NP$

DAT

CamScanner

Subject: Date:

ج) یک for از $i=0$ تا n می‌زنیم و $x=w[0:i]$ $y=w[i+1:n-1]$ P:

M_1 را روی x اجرا ← پذیرفت ← M_2 را روی y اجرا ← پذیرفت

⇐ M' می‌پذیرد و اگر for تمام نشود بدون اینکه M' بپذیرد ← رد می‌کند

$O(n \times (n^{k_1} + n^{k_2})) \Rightarrow P \Rightarrow L_1 . L_2 \in P$

NP: همان الگوریتم P فقط for ندارد و i را به صورت غیرقطعی انتخاب

می‌کنیم $O(n^{k_1} + n^{k_2}) \Rightarrow P \Rightarrow L_1 . L_2 \in NP$

د) M' را اینگونه می‌سازیم: $(|w|=n)$ P:

یک گراف جهت دار با $n+1$ گره داریم که از ۰ تا n نام‌گذاری شده‌اند

دو تا for می‌زنیم $i=0$ ، $i<n-1$ و $j=i+1$ $j<n$

substring $w[i+1:j]$ را به M می‌دهیم اگر پذیرفت یال از i به j

رسم می‌کنیم

حالا روی این گراف DFS می‌زنیم (با شروع از ۰) اگر به n رسید w را

می‌پذیریم $O(n^2 \times n^k + n^2) \Rightarrow P \Rightarrow L^* \in P$

DFS

DAT

Subject: Date:

NP: مانند بخش P است با چند تا تغییر

یک for می‌زنیم $i=0$ $i<n$

به صورت غیر قطعی i بعدی را حدس می‌زنیم که از i قبلی بزرگتر باشد

substring $w[i_{old}+1 : i_{new}]$ را به M می‌دهیم اگر رد کرد این حدس غلط

میشه و ادامه نمی‌دیم اگر نه $i=i_{new}$ تا وقتی که $i=n$ که اینجا

رشته را می‌پذیریم

$O(n \times n^k) \Rightarrow P \Rightarrow L^* \in NP$

Subject: Date:

۳ـ الف) با انجام بحث verification اثبات می‌کنیم: Subset-sum $\in NP$

فقط کافی است که $\sum y_i$ را حساب کنیم ($\{y_1, \dots, y_k\} \subseteq S$) و اگر برابر t بود ← accept

جمع کردن هم تو اوردر چند جمله‌ای است

حالا برای اثبات NP-complete می‌یابیم مسئله 3CNF-SAT رو به SUBSET-SUM کاهش می‌دهیم (در زمان چند جمله‌ای)

$x_1, \dots, x_m$ ← لیترال $c_1, \dots, c_l$ ← clause

اعداد را اینگونه می‌سازیم ($m+l$ رقمی)

اعداد لیترال‌ها (m رقمی):

برای هر x_i، دو عدد y_i (نشان دهنده True) و z_i (نشان دهنده False) داریم:

رقم i اُم y_i و z_i یک است

رقم $m+j$ y_i یک است اگر $x_i \in c_j$

// // // z_i // // $\bar{x}_i \in c_j$

بقیه ارقام صفر

DAT

Subject: Date:

اعداد clause (l رقمی)

برای هر C_i دو عدد a_i و b_i داریم:

فقط رقم $m+i$ هر دو یک است و بقیه ارقام صفر

ساخت t:

مقدار یک را در ارقام اول تا m اُم قرار می‌دهیم و از $m+1$ تا

$m+l$ مقدار ۳ را قرار می‌دهیم

در مجموع $2(m+l)$ عدد ساختیم که هرکدام $m+l$ رقم دارند

$$\Rightarrow O(2(m+l)^2) = O((m+l)^2) \longrightarrow P$$

با این ریدکشن ثابت شد $SUBSET-SUM \in NP$

DAT

Subject: Date:

ب) در الف مبنا ۱۰ بود و ما اعدادی ساختیم که m+l رقم داشتند

برای نشان دادن آن در مبنای unary در worst-case به 10^{m+l}

عدد یک نیاز داریم که به وضوح زمان این ریدکشن در اورد نمایی

است پس UNARY_SUM $\notin$ NP-complete

حالا اثبات می کنیم عضو P است:

یک مجموعه A داریم که در ابتدا $A=\{0\}$

برای هر x در S داریم:

$$A = A \cup \{a+x \mid a \in A \wedge a+x \leq t\}$$

در آخر اگر عدد t در مجموعه بود ورودی را می پذیریم اگر نه رد می کنیم

در بدترین حالت کنام ۰ تا t را دارد پس: $|A| \leq t+1$

چون مبنا یک است و اگر طول کل رشته ورودی ($\langle S, t \rangle$) n باشد

$n \leq t \Rightarrow |A|$ از $O(n)$ است و تعداد بروزرسانی A هر به اندازه $|S|$ است

که $O(n)$ است پس در کل ما داریم به اندازه $t|S|$ پردازش انجام می دهیم

$$\Rightarrow O(n^2) \rightarrow P \rightarrow \text{UNARY_SUM} \in P$$

DAT

Subject: Date:

۴- نوار M را به ۲k بخش (track) تقسیم می‌کنیم. برای هر k نوار:

یک بخش دارای محتوای نوار i ام است

بخش دیگر علامت # را دارد که نشان می‌دهد هد نوار i ام کجاست و بقیه

خانه‌های این بخش خالی هستند

در اول کار ورودی را در بخش اول می‌گذاریم و خانه اول بخش دوم را #

می‌گذاریم و برای k-1 نوار دیگر بخش اول خالی و بخش دوم (هد)

روی موقعیت استارت است

حالا M از چپ‌ترین # تا راست‌ترین # می‌رود و محتوای نوار را ذخیره

می‌کند حالا با استفاده از تابع انتقال T محتوای جدید و موقعیت هدها را

آپدیت می‌کند (هدها یکی به راست یا چپ می‌روند)

حالا فرض کنیم که T روی یک ورودی در m مرحله متوقف می‌شود

در مرحله i ام ماشین M باید نوار خود را بپیماید که طول این متناسبا فاصله

هدها است که در worste-case $O(i)$ است پس

$$\sum_{i=1}^{m} O(i) = O(m^2)$$

DAT

پس اگر T در $O(m)$ بپذیرد M در $O(m^2)$ می‌پذیرد همان ورودی را

Subject: Date:

۵- الف) برای بخش verification یک mapping $f: V(G_2) \to V(G_1)$

داریم که در زمان چند جمله ای چک می کند که f یک به یک باشد و برای

هر یال (u,v) در G_2 حتما $(f(u),f(v))$ در G_1 باشد

پس مسئله هم ریختی گراف $\in NP$

حالا طبق اسلایدهای درس می دانیم: $CLIQUE \in NP\text{-complete}$

کاهش: G_1 را همان G قرار می دهیم و G_2 را یک گراف کامل با k رأس

می سازیم که در $O(k^2)$ ساخته می شود

صحت کاهش:

پیدا کردن زیرگرافی در G_1 که با گراف کامل k رأسی یک ریخت باشد

دقیقا معادل پیدا کردن k رأس در G_1 است که همه به هم متصل باشند

پس داریم:

مسئله هم ریختی گراف $\in NP\text{-complete}$

DAT

Subject: Date:

ب) برای بخش verification یک مجموعه از رئوس S داریم که در زمان چند جمله‌ای چک می‌کنیم که: $|S| \leq k+1$ ۲- $\forall (u,v) \in E \mid u \in S \text{ or } v \in S$

پس: $\in NP$ مسئله پوشش رأسی

حالا طبق اسلایدها که اثبات شد داریم:

برای تبدیل 3CNF-SAT با m لیترال و l clause برای هر متغیر

یک گسبت شامل x_i و $\bar{x}_i$ متصل با یک یال می‌سازیم برای هر clause

هر یک گراف کامل ۳ رأسی می‌سازیم سپس رئوس متناظر را وصل می‌کنیم

در نهایت مقدار k را برابر $m+2l$ می‌گذاریم

مجموع گره‌ها $2m+3l$ می‌شود و مجموع یال‌ها $m+6l$

$\Rightarrow O(2m+3l+m+6l) = O(m+l) \rightarrow P$

گراف حاصل یک پوشش رأسی به اندازه k دارد $\Longleftrightarrow$ فرمول اولیه ارضاپذیر باشد

$\Rightarrow$ مسئله پوشش رأسی $\in NP\text{-complete}$

DAT

Subject: Date:

هـ) برای بخش verification داریم

می‌توانیم با یک پیمایش در زمان چند جمله‌ای بررسی کنیم که دنباله رئوس

یک مسیر ساده در G است که از a شروع و به b ختم شود و طول حداقل k

داشته باشد پس: $LPATH \in NP$

طبق اسلایدها می‌دانیم: $HAMILTONIAN-PATH \in NP-complete$

کاهش:

فرض کنید $G(V,E)$ و دو رأس u و v برای مسیر همیلتونی داریم

می‌گذاریم $a=u$ و $b=v$ k هم می‌شود $|V|-1 \Longleftarrow O(1)$

پس G یک مسیر همیلتونی دارد $\Longleftrightarrow$ یک مسیر ساده از u به v با طول حداقل

$|V|-1$ داشته باشد

$\Longrightarrow LPATH \in NP-complete$

DAT

Subject: Date:

$\alpha \in N$

i اُمین TM

E تمامی (M_i, α) را تولید می‌کند و برای هر جفت یک TM به اسم

$M_{i\alpha}$ درست می‌کند و خروجی می‌دهد

نحوه کارکرد $M_{i\alpha}$ روی رشته w:

M_i و T_α را همزمان روی w اجرا می‌کند

اگر قبل از تمام نشدن کار T_α، M_i رشته را پذیرفت ← accept

اگر نپذیرفت ← reject

اگر هم T_α متوقف شد بدون اینکه M_i جواب دهد ← reject

decider بودن خروجی:

چون T_α در زمان محدود کارش را تمام می‌کند حتما $M_{i\alpha}$ در زمان

محدود خروجی می‌دهد پس تصمیم پذیر است

مرز بالا A_{EP} ؟

اگر A_{EP} پس یک TM M_A وجود دارد که در زمان چند جمله‌ای تصمیم می‌گیرد

E تمامی جفت‌ها را تولید می‌کند و در یکی از آنها M_A را داریم

DAT

این کار یک معادل سازی است

Subject: Date:

و وقتی E ماشین $M_{A\alpha}$ را دارد می سازد (فرض کنیم که M_A حداکثر در زمان $|x|^{\alpha}$ کارش تمام است) M_A قبل از T_α تمام می شود و accept و reject می دهد پس $M_{A\alpha}$ همان رفتار M_A را دارد که A را می پذیرد

پس مرز بان دلخواهی در P حداقل یک بار توسط یکی از ماشین های تولید شده از E پوشش داده می شود

DAT