TEOREMA (di Cook).

SAT è NP-completo.

Dimostrazione.

Dobbiamo mostrare che SAT $\in$ NP e che ogni altro problema NP è riducibile a SAT in tempo polinomiale.

Per quanto riguarda il primo punto ci limitiamo ad osservare che il non determinismo della MdT ci permette di seguire contemporaneamente cosa avviene quando assegnamo i valori vero o falso ai vari atomi (ed è ciò che permette di ottenere il risultato in tempo polinomiale).

Passiamo quindi direttamente al secondo (e più difficile) problema.

Sia M una MdT non deterministica che accetta un generico $L \in NP$ e sia $p(n)$ un polinomio che limita il numero di passi richiesti da M per accettare una stringa di ingresso; assumiamo inoltre che $p(n) \geq n$.

Ciò che dobbiamo mostrare è che esiste una funzione calcolabile in tempo polinomiale che trasforma ogni stringa di ingresso w di M in una formula δ_w in CNF tale che w è accettata da M se e solo se δ_w è soddisfacibile.

Se w è una stringa di ingresso, poniamo $t = p(|w|)$.

Sappiamo che se M accetta la stringa w lo fa in non più di t passi.
Quindi per determinare se M accetta w dobbiamo solo far girare la macchina per al più t passi e controllare se la configurazione finale è una <u>configurazione terminale</u>.

Poiché ad ogni passo M può al più muoversi di <u>uno</u> quadrato a destra o a sinistra del quadrato esaminato, dopo t passi la macchina starà esaminando un quadrato che si trova, al più, a distanza t – a destra o a sinistra – del quadrato di partenza.
Poiché abbiamo scelto un polinomio tale che $p(n) \geq n$, avremo che $t \geq |w|$ e quindi per tutta la durata del calcolo è sufficiente considerare una porzione di nastro che contiene $2t+1$ quadrati.
Tutto il processo di calcolo è perciò contenuto in una tabella $t \times (2t+1)$ [t configurazioni successive di un segmento lungo $2t+1$].

Per rendere più meccanico ed uniforme il nostro modo di procedere per verificare l'accettazione di una stringa assumiamo ancora che questa venga determinata dal fatto che la MdT va a finire in uno stato particolare prescelto q_u (questa è una piccola variante del modello base di MdT che abbiamo già visto) ed inoltre assumiamo che se la MdT è arrivata in questo stato prima di t passi allora ripeteremo la configurazione finale nella nostra rappresentazione in modo da potere esaminare sempre la configurazione numero t per sapere se la stringa è stata accettata o meno.

Le "informazioni-base" di cui abbiamo bisogno sono le seguenti:

1. La MdT M si trova nello stato q_u esaminando il quadrato j-esimo al passo k-esimo del calcolo. (e quindi nella riga k-esima della nostra tabella).

2. Il simbolo S_i si trova nel quadrato j-esimo al passo k-esimo del calcolo.

Indichiamo tali situazioni elementari rispettivamente con i simboli $\varsigma_{h,j,k}$ e $\sigma_{i,j,k}$

Tali simboli saranno gli atomi del nostro calcolo.

L'insieme degli atomi è quindi dato da:

$$A = \{ \varsigma_{h,j,k}, \sigma_{i,j,k} \mid 1 \leq h \leq m ; \; 0 \leq i \leq r ;$$

stati della MdT — simboli

$$1 \leq j \leq 2t+1 ; \; 1 \leq k \leq t \}$$

posizione occupata sulla porzione di nastro (e su ogni riga della tabella)

passo del calcolo (riga della tabella)

Costruiremo adesso una formula δ_u (che o è già in CNF o è ad essa trasformabile con le tecniche standard del calcolo proposizionale) tale che $\delta_u^v = 1$, con v che assegna i valori "naturali" agli atomi e cioè:

$$v(\varsigma_{h,j,k}) = \begin{cases} 1 & \text{se } M \text{ si trova nello stato } q_h \text{ esaminando il quadrato } j\text{-esimo al passo } k\text{-esimo del calcolo} \\ 0 & \text{altrimenti} \end{cases}$$

$$v(\sigma_{i,j,k}) = \begin{cases} 1 & \text{se il simbolo } S_i \text{ si trova nel posto } j\text{-esimo della riga } k\text{-esima della tabella} \\ 0 & \text{altrimenti.} \end{cases}$$

Introduciamo adesso una abbreviazione:

$$\nabla\{x_e \mid 1 \le e \le \ell\} = \left(\bigwedge_{1 \le e < f \le \ell} (\neg x_e \vee \neg x_f) \right) \wedge \left(\bigvee_{1 \le e \le \ell} x_e \right)$$

dove le x_e, $1 \le e \le \ell$, sono formule.

$\nabla\{x_e\}$ è una formula che è vera in una certa interpretazione se e solo se <u>esattamente una sola</u> delle formule presenti <u>è vera</u> in quella interpretazione.

In generale le x_e sono formule generiche, nel caso in cui sono atomi, la ∇, come si vede facilmente è una formula in forma normale congiuntiva.
Calcoliamo l'ordine di grandezza della lunghezza di ∇ in quest'ultimo caso.
∇ è formata da una clausola di due letterali per ogni coppia di indici e, f tali che $1 \le e < f \le \ell$ e da una clausola di ℓ letterali.
Quindi avremo $\frac{\ell(\ell-1)}{2}$ stringhe di lunghezza 3 ($\neg x_e$, $\neg x_f$ e barra di separazione / tra le clausole) ed una stringa di lunghezza $\ell+1$, quindi

$$\left| \nabla\{x_e \mid 1 \le e \le \ell\} \right| = \frac{\ell(\ell-1)}{2} \cdot 3 + (\ell+1)$$

che è dell'ordine di grandezza di ℓ^2.

Esempio 1 (con due variabili).

$$\nabla\{x_e \mid 1 \le e \le 2\} =$$

$$= \underbrace{\bigwedge_{1 \le e < f \le 2} (\neg x_e \vee \neg x_f)}_{\parallel} \wedge \underbrace{\bigvee_{1 \le e \le 2} x_e}_{\parallel} =$$

$$= (\neg x_1 \vee \neg x_2) \wedge (x_1 \vee x_2)$$

Esempio 2 (con tre variabili)

$$\nabla\{x_e \mid 1 \le e \le 3\} =$$

$$= \underbrace{\bigwedge_{1 \le e < f \le 3} (\neg x_e \vee \neg x_f)}_{\parallel} \wedge \underbrace{\bigvee_{1 \le e \le 3} x_e}_{\parallel} =$$

$$= (\neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2) \wedge (x_1 \vee x_2 \vee x_3)$$

Come facciamo a tradurre in formule del calcolo proposizionale il computo effettuato dalla MdT?

Spezzeremo tutto il processo in una serie di passi più elementari e "tradurremo" questi in formule costruite a partire dagli atomi di ~~Aff~~.

Per prima cosa dobbiamo partire dalla configurazione iniziale del nastro. Si ha che:

(1) "Nella configurazione iniziale il contenuto del nastro di $\mathcal{M}$ corrisponde a quello della prima riga della tabella: $\mathcal{M}$ si trova nello stato q_1 esaminando il simbolo S_0 immediatamente a sinistra del primo simbolo della stringa di ingresso x"

"codifica":

$$\bigwedge_{0<j\leq t+1} \sigma_{0,j,1} \wedge \bigwedge_{0<j\leq z} \sigma_{x_j,t+j+1,1} \wedge \bigwedge_{0<j\leq t-z} \sigma_{0,t+z+j+1,1} \wedge q_{1,t+1,1}$$

- $0<j\leq t+1$: i primi $t+1$ simboli sono S_0
- $0<j\leq z$: dal quadrato $t+2$ al quadrato $t+z+1$ c'è scritta la stringa x
- $0<j\leq t-z$: dal quadrato $t+z+2$ al quadrato $2t+1$ c'è il simbolo S_0
- $q_{1,t+1,1}$: $\mathcal{M}$ si trova nello stato q_1 esaminando il $(t+1)$-esimo quadrato

Sempre nel primo rigo della tabella.

la lunghezza della stringa è dell'ordine di t.

Si ha inoltre che:

(2) "In ciascun passo del calcolo la MdT M si trova in un unico stato esaminando un solo quadrato"

"codifica":

$$\bigwedge_{1\le k\le t} \nabla\{ S_{h,j,k} \mid 1\le h\le m\ ,\ 1\le j\le 2t+1 \} \qquad (*)$$

Osservazioni

- $S_{h,j,k}$ esprime il fatto che M si trova nello stato q_h esaminando il quadrato j-esimo nel passo k-esimo del calcolo.
- l'operatore ∇ ci assicura che uno solo dei $S_{h,j,k}$ è vera per un k fissato, cioè ad ogni passo del computo, facendo variare gli altri indici in modo da considerare tutti gli stati possibili e tutti i quadrati accessibili possibili.
- Considerando infine la congiunzione su tutti i passi k del calcolo otteniamo la (2).

In base a quanto osservato prima sulle formule di tipo ∇ possiamo concludere che la (*) ha una lunghezza dell'ordine di t^3

Dobbiamo poi avere che:

(3) "Ciascun elemento della nostra tabella deve contenere esattamente un simbolo."

Usando ancora una volta l'operatore ∇ si ha che (una quanta volta applicato a σ)

"codifica":

$$\bigwedge_{1\leq k\leq t} \bigwedge_{1\leq j\leq 2t+1} \nabla\{\sigma_{i,j,k} \mid 0\leq i\leq r\}$$

Per k e j fissati $\nabla\{\sigma_i\}$ ci assicura che esattamente un simbolo si trova nel quadrato j-esimo della riga k-esima.
Per ottenere la (3) dobbiamo considerare la congiunzione degli enunciati riferentisi ad ogni elemento della tabella.

L'ordine di grandezza della lunghezza della stringa al crescere di t è dato da t^2
(r è il numero dei simboli ed è fissato in partenza).

TC8

Vediamo adesso come si può "codificare" il passaggio da una configurazione all'altra della MdT M. Date anche le convenzioni aggiuntive fatte ciò che si deve "codificare" si riassume in:

(4) "Ciascuna configurazione nel calcolo, dopo la prima, o è identica alla precedente o è ottenuta dalla precedente applicando una delle quadruple di M".

Suddivideremo il problema in cinque sottoproblemi.

i) M <u>non</u> sta esaminando il quadrato j-esimo al passo k-esimo del computo.

Ciò è "codificato" da:

$$\text{NOTHEAD}(j,k) = \bigvee_{0 \le i \le r} (S_{i,j,k} \wedge \bar{S}_{i,j,k+1}) \wedge \bigwedge_{1 \le h \le m} \neg S_{h,j,k}$$

È facile verificare che NOTHEAD(j,k) assume il valore 1 se e solo se la i) è verificata.

ii) M sta esaminando il j-esimo quadrato
sia al passo k-esimo sia al passo (k+1)esimo
del calcolo esaminando lo stesso simbolo
e trovandosi nello stesso stato

"codifica":

$$IDENT(j,k) = \bigvee_{1 \le h \le m} \bigvee_{0 \le i \le r} \left(\rho_{h,j,k} \wedge \sigma_{i,j,k} \wedge \rho_{h,j,k+1} \wedge \sigma_{i,j,k+1} \right)$$

IDENT(j,k) assume il valore 1 se e solo se
la (ii) è verificata

iii) Si passa dal k-esimo al (k+1)-esimo passo del computo applicando una quadrupla di "stampa"

$$\{ q_{i_a} S_{j_a} S_{k_a} q_{l_a} \mid a = 1, 2, \ldots, \bar{a} \}$$

"codifica"

$$A(j,k) = \bigvee_{1 \le a \le \bar{a}} \left(\varphi_{i_a, j, k} \wedge \sigma_{j_a, j, k} \wedge \sigma_{k_a, j, k+1} \wedge \varphi_{l_a, j, k+1} \right)$$

$A(j,k)$ assume il valore 1 se e solo se la iii) è verificata.

iv) Si passa dal k-esimo al (k+1)-esimo passo del computo applicando una quadrupla di spostamento a destra:

$$\{ q_{i_b} S_{j_b} R\, q_{\ell_b} \mid b = 1, 2, \dots, \bar{b} \}$$

"codifica"

$$B(j,k) = \bigvee_{1 \le b \le \bar{b}} \left(\varsigma_{i_b, j, k} \wedge \sigma_{j_b, j, k} \wedge \sigma_{j_b, j, k+1} \wedge \varsigma_{\ell_b, j+1, k+1} \right)$$

per $j \neq 2t+1$

e $B(j,k) = \square$ (doppia vuota) per $j = 2t+1$

(N.B. la prima definizione applicata a $j = 2t+1$ porterebbe ad esaminare un quadrato posto al di là della porzione di nastro considerata).

v) Si passa dal k-esimo al $(k+1)$-esimo passo del computo applicando una quadrupla di spostamento a sinistra:

$$\{ q_{i_c} S_{j_c} L q_{l_c} \mid c = 1, 2, \dots, \bar{c} \}$$

"codifica"

$$\mathcal{C}(j,k) = \bigvee_{1 \le c \le \bar{c}} \left(\varrho_{i_c, j, k} \wedge \bar{\sigma}_{j_c, j, k} \wedge \bar{\sigma}_{j_c, j, k+1} \wedge \varrho_{l_c, j-1, k+1} \right)$$

per $j \neq 1$

e $\mathcal{C}(j,k) = \square$ (clausola vuota)

per $j = 1$

(anche in questo caso, un particolare valore di j ci porterebbe fuori dalla porzione di nastro considerata).

È chiaro che il funzionamento della MdT M riassunto nell'enunciato (4) è "codificato" da

$$\bigwedge_{1\le k<t}\ \bigwedge_{1\le j\le 2t+1} \left(\text{NOTHEAD}(j,k) \vee \text{IDENT}(j,k) \vee A(j,k) \vee B(j,k) \vee @(j,k)\right)$$

Poiché la lunghezza di ciascuno dei cinque disgiunti non varia al variare degli indici e poiché la quantificazione limitata è effettuata sui due indici k e j che variano rispettivamente tra 1 ed t e tra 1 e $t+1$, possiamo concludere che la lunghezza di tutta la formula è dell'ordine di t^2.

Infine dobbiamo considerare esplicitamente il fatto che

(5) "la t-esima configurazione è una configurazione terminale, il che vuol dire che, nella convenzione qui usata, M si trova nello stato q_m".

"Codifica":

$$\bigvee_{1 \le j \le 2t+1} \varphi_{m,j,t} \qquad (*)$$

Ordine di grandezza della lunghezza della (*): t.

Assumiamo adesso che δ_n sia la congiunzione di tutte le formule di "codifica" fin qui considerate (i cinque passi base).

È chiaro, per come queste ultime sono state costruite, che se M accetta n allora δ_n è soddisfacibile.

Viceversa assumiamo che δ_u sia soddisfacibile. Questo vuol dire che esiste una assegnazione σ tale che $\delta_u^\sigma = 1$.

Ma questo vuol dire che ciascuno dei suoi cinque congiunti è vero; procedendo a ritroso possiamo mostrare che M accetta u.

In particolare

- la (3) ci assicura che possiamo ricostruire la tabella
- la (2) che vi è un unico stato ed un unico quadrato esaminato in ciascun rigo (questo ci permette di ricostruire <u>una</u> configurazione di M).
- la (1) ci permette di individuare la configurazione iniziale.
- la (4) le azioni delle quadruple
- la (5) la configurazione finale.

Possiamo quindi ricostruire il processo di calcolo e quindi u è accettata da M.

che esiste una funzione calcolabile in tempo polinomiale che trasforma ciascuna stringa x nella formula corrispondente δ_x lo si vede facilmente considerando la semplicità dei vari pezzi che compongono δ_x.

Clay Mathematics Institute

| [home] |

Millennium Prize Problems

P versus NP

The Hodge Conjecture

The Poincaré Conjecture

The Riemann Hypothesis

Yang-Mills Existence and Mass Gap

Navier-Stokes Existence and Smoothness

The Birch and Swinnerton-Dyer Conjecture

Announced 16:00, on Wednesday, May 24, 2000
Collège de France

Statement from the Directors and Scientific Advisory Board

Rules for the CMI Millennium Prize Problems

Historical Context

Press Release

Press Contact: +33 (0)1 44 27 12 72 (Veronique Lemaitre, France)

+1-609-275-7444 (Rachel Ingbar, U.S.)

webmaster@claymath.org

---

www.claymath.org

Clay Mathematics Institute

Historical Context

David Hilbert was born in Königsberg, Germany, on January 23, 1862. He obtained his doctorate from the University of Königsberg in 1885, where he remained until 1895 when he took up the professorship at Göttingen he was to hold until his death in 1943.

Hilbert's contributions to mathematics have been vast and far-reaching. His early work was concerned with the theory of invariants, while later he moved into the foundations of geometry and the theory of algebraic number fields. At the turn of the century Hilbert's research efforts broadened yet further, encompassing potential theory, the calculus of variations and various areas of mathematical physics. In his later years Hilbert became primarily involved with the foundations of mathematics, and he is now remembered as one of the greatest mathematicians of the twentieth century. Indeed, it is staggering how many deep results and profound conjectures Hilbert produced across the wide spectrum of his mathematical interests. He also wrote monumental texts on the foundations of mathematics, geometry, logic and algebraic number theory. By the end of the nineteenth century Hilbert's achievements had already lifted him to a point from which he dared to chart out the most promising avenues for research in the twentieth century. In 1900 Hilbert gave life to his vision through the formulation of twenty-three problems that he presented at the International Congress of Mathematicians in Paris. These problems have inspired and guided the minds of mathematicians throughout the last century. Out of the original twenty-three problems eight were of a purely investigative nature. To date twelve of the remaining fifteen have been completely resolved. Quite remarkably, only one problem, the so-called Riemann Hypothesis remains as mysterious and challenging as ever, being now widely regarded as the most important open problem in pure mathematics.

The CMI Millennium Prize Problems are not intended to shape the direction of mathematics in the next century. Rather these problems focus attention on a small set of long-standing mathematical questions, each central to mathematics, that also have resisted many years of serious attempts by experts to solve them. The Riemann hypothesis is one of Hilbert's original questions.
