MEASURING COMPLEXITY BY USING REDUCTION TO SOLVE P VS NP AND NC & PH

KOBAYASHI KOJI

1. Abstract

This article describes about that NC and PH is proper (especially P is not NP) by using problem reduction. If L is not P, we can prove P is not NP by using difference between logarithm space reduction and polynomial time reduction. Like this, we can also prove that NC is proper by using AL0 is not NC1. This means L is not P. Therefore P is not NP. And we can also prove that PH is proper by using P is not NP.

2. P is not NP if L is not P

Definition 1. We will use the term "L", "P", "$P-Complete$", "NP", "$NP-Complete$", "FL", "FP" as each complexity classes. These complexity classes also use Turing Machine (TM) set that compute target complexity classes problems. "$f\circ g$" as composite TM that accepting configurations of g are starting configurations of f.

Theorem 2. $L\subsetneq P\rightarrow P\subsetneq NP$

Proof. To prove it by using contraposition $P=NP\rightarrow L=P$.

As we all know that if $P=NP$ then all NP can reduce $P-Complete$ under FL.

$P=NP\rightarrow\forall A\in P-Complete,B\in NP\exists C\in FL\left(A\circ C=B\right)$

This is correct even if NP reduce by any FP.

$P=NP\rightarrow\forall D\in P-Complete,E\in NP,F\in FP\exists G\in FL\left(D\circ G=E\circ F\right)$

If $P=NP$, all NP can reduce $\{1\}$ under some FP.

$P=NP\rightarrow\forall D\in P-Complete\exists G\in FL\left(D\circ G=\{1\}\right)$

This means $L=P$. Therefore, this theorem was shown. □

3. NC is proper

We use circuit problem as follows;

Definition 3. We will use the term "AC^i", "NC^i" as each complexity decision problems classes. "FAC^i" as function problems class of "AC^i". These complexity classes also use uniform circuits family set that compute target complexity classes problems. "$f\circ g$" as composite circuit that output of g are input of f. In this case, we also use complexity classes to show target circuit. For example, $A\circ BB$ when A is circuits family and BB is circuits family set mean that $a\circ b\mid a\in A,b\in B\in BB$. Circuits family uniformity is that these circuits can compute FAC^0.

Theorem 4. $NL\leq_{AC^0}NC^2$

Proof. Mentioned [1] Theorem 10.40, all NC^2 are closed by FL reduction. This reduction is validity of (c_1, c_2) transition function. Transition function change $O(1)$ memory and keep another memory. Therefore this validity can compute AC^0 and we can replace FL to FAC^0. □

Theorem 5. *AC^i has Universal Circuits Family that can emulate all AC^i circuits family. That is, every AC^i has $AC^i - Complete$.*

Proof. To prove this theorem by making universal circuit family $A^i \in AC^i$ that emulate circuit family $\{C_j\} \in AC^i$ by using "depth circuit tableau". Universal circuit $U_j \in A^i$ have partial circuit $u_{k,d}$ that emulate all C_j gates $g_{k \in n}$ (include input value) and connected wires $w_{p,q}$ from g_p output to g_q input in every depth d. ($w_{p,p}$ always exist)

$u_{v \in n,d}$ have inputs from all $u_{u \in n,d-1}$ and g_u information that mean
a) validity of $u_{u,d-1}$
b) $u_{u,d-1}$ output (true if g_u output true)
c) existence of $w_{u,v}$ (true if $w_{u,v}$ is exists)
d) negation of $w_{u,v}$ (true if $w_{u,v}$ include not gate)
e) gate type of g_v (Or gate or And gate)
and outputs to $u_{w \in n,d+1}$ that mean
A) validity of $u_{v,d}$
B) $u_{v,d}$ output
These $u_{v,d}$ compute output like this;
If $u_{u,d-1}$ a) or c) input false then $u_{v,d}$ ignore $u_{u,d-1}$.
If $u_{u,d-1}$ a) and c) input true then $u_{v,d}$ A) output true and $u_{v,d}$ B) output g_k value that compute from e), b), d). b), d) include another $u_{w \in n,d-1}$ b), d).
If all a) input false then $u_{k,d}$ A) output false.
If all c) input false then $u_{k,d}$ A) output false.
And depth 0 circuit compute additional condition;
If $u_{k,0}$ is C_j input then $u_{k,0}$ A) output true and $u_{i,d}$ B) output C_j input value, else $u_{k,0}$ A) output false.

This U_j that consists of u emulate C_j. We can make every u in FAC^0, so that A^i in AC^i.

Therefore, this theorem was shown. □

Theorem 6. $NC^i \subsetneq NC^{i+1}$

Proof. To prove it using reduction to absurdity. We assume that $NC^i = NC^{i+1}$. It is trivial that $NC^i = AC^i = \cdots = NC^{2i}$.

Mentioned above 5, all AC^i can reduce $AC^i - Complete$ under AC^0. Therefore if $NC^i = NC^{i+1}$ then all NC^{2i} can reduce $AC^i - Complete$ under AC^0.

$NC^i = NC^{i+1} \rightarrow \forall A \in AC^i - Complete, B \in NC^{2i} \exists C \in AC^0 \, (A \circ C = B)$

All $NC^i \circ NC^i$ is in NC^{2i}. Therefore above is correct even if NC^i is $NC^i \circ NC^i$.

$NC^i = NC^{i+1} \rightarrow \forall D \in AC^i - Complete, E, F \in NC^i \exists G \in AC^0 \, (D \circ G = E \circ F)$

All NC^i can reduce $\{1\}$ under some NC^i.

$NC^i = NC^{i+1} \rightarrow \forall D \in AC^i - Complete \exists G \in AC^0 \, (D \circ G = \{1\})$

This means $AC^0 = AC^i$. But this contradict contradict $AC^0 \subsetneq NC^1 \subset AC^i$.

Therefore, this theorem was shown than reduction to absurdity. □

4. P is not NP

Theorem 7. $P \neq NP$

Proof. Mentioned above 2, $L \subsetneq P \to P \subsetneq NP$. And mentioned above 6, $L \subset NC^i \subsetneq NC^{i+1} \subset P$. Therefore $P \subsetneq NP$. □

5. PH is proper

Theorem 8. $\Pi_k \subsetneq \Pi_{k+2}$

Proof. To prove it using reduction to absurdity. We assume that $\Pi_k = \Pi_{k+2}$. It is trivial that $\Pi_k = \Pi_{k+2} = \cdots = \Pi_{2k}$.

Mentioned [2] Theorem 6.26, $QSAT_k'$ are $\Pi_k - Complete$ under polynomial time reduction. All Π_k can reduce $\Pi_k - Complete$ under FP. Therefore if $\Pi_k = \Pi_{k+2}$ then all Π_{2k} can reduce $\Pi_k - Complete$ under FP.

$\Pi_k = \Pi_{k+2} \to \forall A \in \Pi_k - Complete, B \in \Pi_{2k} \exists C \in FP \left(A \circ C = B\right)$

All $\Pi_k \circ \Pi_k$ is in Π_{2k}. Therefore, if $\Pi_k = \Pi_{k+2}$ then above is correct even if Π_k is $\Pi_k \circ \Pi_k$.

$\Pi_k = \Pi_{k+2} \to \forall D \in \Pi_k - Complete, E, F \in \Pi_{2k} \exists G \in FP \left(D \circ G = E \circ F\right)$

All Π_k can reduce $\{1\}$ under some Π_k.

$\Pi_k = \Pi_{k+2} \to \forall D \in \Pi_k - Complete \exists G \in FP \left(D \circ G = \{1\}\right)$

This means $FP = \Pi_k$. But this contradict contradict $FP \subsetneq NP \subset \Pi_k$ mentioned above7.

Therefore, this theorem was shown than reduction to absurdity. □

Theorem 9. $\Delta_k \subsetneq \Sigma_k, \Sigma_k \neq \Pi_k$

Proof. Mentioned [2] Theorem 6.12,

$\Sigma_k = \Pi_k \to \Sigma_k = \Pi_k = PH$

$\Delta_k = \Sigma_k \to \Delta_k = \Sigma_k = \Pi_k = PH$

This contraposition is,

$\left(\Sigma_k \subsetneq PH\right) \vee \left(\Pi_k \subsetneq PH\right) \to \Sigma_k \neq \Pi_k$

$\left(\Delta_k \subsetneq PH\right) \vee \left(\Sigma_k \subsetneq PH\right) \vee \left(\Pi_k \subsetneq PH\right) \to \Delta_k \neq \Sigma_k$

From mentioned above 8,

$\Sigma_k \subsetneq \Pi_{k+1} \subset PH$

Therefore, $\Delta_k \neq \Sigma_k, \Sigma_k \neq \Pi_k$.

Mentioned [2] Theorem 6.10,

$\Sigma_k \subset \Sigma_{k+1}, \Pi_k \subset \Pi_{k+1}, \forall k \geq 1 \left(\Delta_k \subset \left(\Sigma_k \cap \Pi_k\right) \subset \left(\Sigma_k \cup \Pi_k\right) \subset \Delta_{k+1}\right)$

Therefore, $\Delta_k \subsetneq \Sigma_k, \Sigma_k \neq \Pi_k$. □

Theorem 10. $\Pi_k \not\subset \Sigma_k, \Sigma_k \not\subset \Pi_k$

Proof. To prove it using reduction to absurdity. We assume that $\Pi_k \subset \Sigma_k$. This means that all $\overline{\Sigma_k} = \Pi_k$ is also Σ_k.

$\Pi_k \subset \Sigma_k \to \forall A \in \Sigma_k \left(\overline{A} \in \Pi_k \subset \Sigma_k\right)$

Mentioned [2] Theorem 6.21, all Σ_k are closed under polynomial time conjunctive reduction. We can emulate these reduction by using Π_1. That is,

$\exists B \in \Sigma_k \forall C \in \Sigma_k \exists D \in \Pi_1 \left(B \circ D = C\right)$

Therefore,

$\Pi_k \subset \Sigma_k$

$\to \exists B \in \Sigma_k \forall C \in \Sigma_k \exists D \in \Pi_1 \forall A \in \Sigma_k \left(B \circ D = C\right) \wedge \left(\overline{A} \in \Pi_k \subset \Sigma_k\right)$

$\to \exists B \in \Sigma_k \forall C \in \Sigma_k \exists D \in \Pi_1 \left(B \circ D = C\right) \wedge \left(\overline{B} \in \Sigma_k\right)$

$\to \exists B \in \Sigma_k \forall C \in \Sigma_k \exists D \in \Pi_1 \left(B \circ D = C\right) \wedge \left(B \in \Pi_k\right)$

Therefore $\Sigma_k \subset \Pi_k$ because $B \circ D \in \Pi_k$. But this means $\Sigma_k = \Pi_k$ and contradict $\Sigma_k \neq \Pi_k$ mentioned above 9. Therefore $\Pi_k \not\subset \Sigma_k$.

We can prove $\Sigma_k \not\subset \Pi_k$like this.
Therefore, this theorem was shown than reduction to absurdity. $\square$

Theorem 11. $\Delta_k \subsetneq \Pi_k$

Proof. To prove it using reduction to absurdity. We assume that $\Delta_k = \Pi_k$.
Mentioned [2] Theorem 6.10,
$\Sigma_k \subset \Sigma_{k+1}$, $\Pi_k \subset \Pi_{k+1}$,$\forall k \geq 1\,(\Delta_k \subset (\Sigma_k \cap \Pi_k) \subset (\Sigma_k \cup \Pi_k) \subset \Delta_{k+1})$
Therefore
$\Delta_k = \Pi_k$
$\rightarrow \Delta_k = \Pi_k \subset (\Sigma_k \cap \Pi_k) \subset \Sigma_k \subset (\Sigma_k \cup \Pi_k) \subset \Delta_{k+1}$
$\rightarrow \Pi_k \subset \Sigma_k$
But this result contradict mentioned above 10.
Therefore, this theorem was shown than reduction to absurdity. $\square$

Theorem 12. $\Sigma_k \subsetneq \Delta_{k+1}, \Pi_k \subsetneq \Delta_{k+1}$

Proof. To prove it using reduction to absurdity. We assume that $\Sigma_k = \Delta_{k+1}$.
Mentioned [2] Theorem 6.10,
$\forall k \geq 1\,(\Delta_k \subset (\Sigma_k \cap \Pi_k) \subset (\Sigma_k \cup \Pi_k) \subset \Delta_{k+1})$
Therefore
$\Sigma_k = \Delta_{k+1}$
$\rightarrow \Delta_k \subset (\Sigma_k \cap \Pi_k) \subset \Pi_k \subset (\Sigma_k \cup \Pi_k) \subset \Sigma_k = \Delta_{k+1}$
$\rightarrow \Pi_k \subset \Sigma_k$
But this result contradict mentioned above 10. Therefore $\Sigma_k \subsetneq \Delta_{k+1}$.
We can prove $\Pi_k \subsetneq \Delta_{k+1}$ like this.
Therefore, this theorem was shown than reduction to absurdity. $\square$

References

[1] Michael Sipser, (translation) OHTA Kazuo, TANAKA Keisuke, ABE Masayuki, UEDA Hiroki, FUJIOKA Atsushi, WATANABE Osamu, Introduction to the Theory of COMPUTATION Second Edition, 2008

[2] OGIHARA Mitsunori, Hierarchies in Complexity Theory, 2006

[3] MORITA Kenichi, Reversible Computing, 2012