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

KOBAYASHI KOJI

1. Abstract

This article prove that NC and PH is proper (especially P is not NP) by using reduction difference. We can prove that NC is proper by using AL0 is not NC. This means L is not P. We can prove P is not NP by using reduction difference between L and P. And we can also prove that PH is proper by using P is not NP.

2. NC is proper

We use circuit problem as follows;

Definition 1. 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 $a \circ b \mid a \in A, b \in B \in BB$. "$R(A)$" as subset of reversible NC that include A. Reversible mean that $\left(R(A) \circ (R(A))^{-1}\right)(x) = x$. Circuits family uniformity is that these circuits can compute FAC^0.

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

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 3. *AC^i has Universal Circuits Family that can emulate all AC^i circuits family. That is, every AC^i has $AC^i - Complete$ under FAC^0.*

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 4. *$NC^i = NC^{i+1} \to NC^i - Complete = AC^i - Complete = NC^{i+1} - Complete$.*

Proof. If $NC^i = NC^{i+1}$, all $NC^i - Complete, AC^i - Complete, NC^{i+1} - Complete$ can reduce each other and $NC^i - Complete, AC^i - Complete, NC^{i+1} - Complete$ in NC^i. Therefore, this theorem was shown. □

Theorem 5. *$NC^i \subsetneq NC^{i+1}$*

Proof. To prove it using reduction to absurdity. We assume that $NC^i = NC^{i+1}$. It is trivial that $NC^i = AC^i = NC^{i+1} = AC^{i+1} = \cdots$.
Because $NC^i = NC^{i+1}$ and mentioned above 4, $R\left(FAC^i - Complete\right) \subset FAC^i - Complete$. Therefore
$NC^i = NC^{i+1} \to \forall A, B \in R\left(FAC^i - Complete\right) \exists C \in FAC^0 \left(A \circ B = A \circ C\right)$
A is reversible circuits family. Therefore A have A^{-1}.
$NC^i = NC^{i+1}$
$\to \forall A, B \in R\left(FAC^i - Complete\right) \exists C \in FAC^0 \left(A^{-1} \circ A \circ B = A^{-1} \circ A \circ C\right)$
$\to \forall B \in R\left(FAC^i - Complete\right) \exists C \in FAC^0 \left(B = C\right)$
This means $FAC^0 = FAC^i$. But this contradict contradict $AC^0 \subsetneq NC^1 \subset AC^i$.
Therefore, this theorem was shown than reduction to absurdity. □

3. PH IS PROPER

Definition 6. 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. We will use the term "Δ_k", "Σ_k", "Π_k" as each Polynomial hierarchy classes. "$f \circ g$" as composite problem that output of g are input of f. "$R(A)$" as "reversible TM" that equal A. Reversible mean that $\left(R(A) \circ (R(A))^{-1}\right)(x) = x$.

Theorem 7. *$R(\Sigma_k) \subset \Sigma_k$, $R(\Pi_k) \subset \Pi_k$.*

Proof. We can reduce Σ_k and Π_k to another Σ_k and Π_k that have tree graph of computation history. (if all configuration keep input, computation history become tree graph.) These Σ_k,Π_k are $R(\Sigma_k)$, $R(\Pi_k)$ because each computation history of each output only reach one input. Therefore $\left(R(A)\circ(R(A))^{-1}\right)(x)=x$. We can compute these reduction in FP. Therefore, this theorem was shown. □

Theorem 8. $P \subsetneq NP$

Proof. To prove it using reduction to absurdity. We assume that $P=NP$.

As we all know that if $P=NP$ then all NP can reduce $P-Complete$ under FL. And all $NP\circ FP\subset NP$. Therefore

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

Mentioned above7, $R\,(NP-Complete)\subset NP-Complete$. Therefore

$P=NP\rightarrow\forall D\in R\,(NP-Complete)\,\forall B\in FP\exists C\in FL\,(D\circ B=D\circ C)$

D is reversible function. Therefore D have D^{-1}.

$P=NP$

$\rightarrow\forall D\in R\,(P-Complete)\,\forall B\in FP\exists C\in FL\left(D^{-1}\circ D\circ B=D^{-1}\circ D\circ C\right)$

$\rightarrow\forall D\in R\,(P-Complete)\,\forall B\in FP\exists C\in FL\,(B=C)$

This means $FP=FL$. But this contradict $FL\subsetneq FP$ mentioned above5. Therefore, this theorem was shown than reduction to absurdity. □

Theorem 9. $\Pi_k=\Pi_{k+1}\rightarrow\Pi_k-Complete=\Pi_{k+1}-Complete$

Proof. If $\Pi_k=\Pi_{k+1}$, all $\Pi_k-Complete,\Pi_{k+1}-Complete$ can reduce each other and $\Pi_k-Complete,\Pi_{k+1}-Complete$ in Π_k. Therefore, this theorem was shown. □

Theorem 10. $\Pi_k\subsetneq\Pi_{k+1}$

Proof. To prove it using reduction to absurdity. We assume that $\Pi_k=\Pi_{k+1}$. It is trivial that $\Pi_k=\Pi_{k+1}=\Pi_{k+2}=\cdots$.

Mentioned [2] Theorem 6.26, $\Pi_k-Complete$ under polynomial time reduction exist. Therefore all $\Pi_{k+1}-Complete$ can reduce $\Pi_k-Complete$ under FP. Because $\Pi_k=\Pi_{k+1}$ and mentioned above 9, $R\,(\Pi_k-Complete)\subset\Pi_k-Complete$. Therefore

$\Pi_k=\Pi_{k+1}\rightarrow\forall A,B\in R\,(\Pi_k-Complete)\,\exists C\in FP\,(A\circ B=A\circ C)$

A is reversible function. Therefore A have A^{-1}.

$\Pi_k=\Pi_{k+1}$

$\rightarrow\forall A,B\in R\,(\Pi_k-Complete)\,\exists C\in FP\left(A^{-1}\circ A\circ B=A^{-1}\circ A\circ C\right)$

$\rightarrow\forall B\in R\,(\Pi_k-Complete)\,\exists C\in FP\,(B=C)$

This means $\Pi_k=FP$. But this contradict contradict mentioned above8. Therefore, this theorem was shown than reduction to absurdity. □

Theorem 11. $\Delta_k\subsetneq\Sigma_k,\Sigma_k\neq\Pi_k$

Proof. Mentioned [2] Theorem 6.12,

$\Sigma_k=\Pi_k\rightarrow\Sigma_k=\Pi_k=PH$

$\Delta_k=\Sigma_k\rightarrow\Delta_k=\Sigma_k=\Pi_k=PH$

This contraposition is,

$(\Sigma_k\subsetneq PH)\vee(\Pi_k\subsetneq PH)\rightarrow\Sigma_k\neq\Pi_k$

$(\Delta_k\subsetneq PH)\vee(\Sigma_k\subsetneq PH)\vee(\Pi_k\subsetneq PH)\rightarrow\Delta_k\neq\Sigma_k$

From mentioned above 10,

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

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

Mentioned [2] Theorem 6.10,

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

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

Theorem 12. $\Pi_k \not\subset \Sigma_k, \Sigma_k \not\subset \Pi_k$

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.

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

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,

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

Therefore,

$\Pi_k \subset \Sigma_k$

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

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

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

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 11. 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. □

Theorem 13. $\Delta_k \subsetneq \Pi_k$

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 12.

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

Theorem 14. $\Sigma_k \subsetneq \Delta_{k+1}, \Pi_k \subsetneq \Delta_{k+1}$

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 12. 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. □

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