# Strong Exponential Time Hypothesis

脊戸 和寿
北海道大学
seto@ist.hokudai.ac.jp

## 1 ETH と SETH

まずは，下記の表を見てもらいたい．これは，Ryan Williams が自身の記事 [7] の中で，未だ証明も反証もされていない各予想がどの程度信憑性があるか数値で示したものである．各計算量クラスの定義については，Complexity Zoo 等を参照してほしい．

表 1: Ryan Williams による estimate likelihood

| 命題 | 信頼度 |
|---|---|
| TRUE | 100% |
| $\mathrm{EXP}^{\mathrm{NP}} \neq \mathrm{BPP}$ | 99% |
| NEXP $\not\subset$ P/poly | 97% |
| L $\neq$ NP | 95% |
| NP $\not\subset$ SIZE($n^k$) | 93% |
| BPP $\subseteq$ SUBEXP | 90% |
| P $\neq$ PSPACE | 90% |
| P $\neq$ NP | 80% |

| 命題 | 信頼度 |
|---|---|
| ETH | 70% |
| $\mathrm{NC}^1 \neq \mathrm{TC}^0$ | 50% |
| NEXP $\neq$ EXP | 45% |
| SETH | 25% |
| NEXP $\neq$ coNEXP | 20% |
| NSETH | 15% |
| L $\neq$ RL | 5% |
| FALSE | 0% |

Ryan Williams の主観なので，みなさんの評価とは異なるであろう．これらのどれか 1 つでも証明もしくは反証できれば，非常に価値のあることであり，P≠NP が証明できれば，100 万ドルも得ることができる．

本稿では R. Impagliazzo と R. Paturi [5] によって提唱された ETH（Exponential Time Hypothesis：指数時間仮説）と SETH（Strong Exponential Time Hypothesis：強指数時間仮説）について紹介したい．まずは SAT（Satisfiability Problem：充足可能性問題）について説明する．ETH と SETH に関係するのは CNF-SAT と呼ばれる SAT である．CNF-SAT とは，与えられた CNF（Conjunctive Normal Form：和積標準形，乗法標準形）を真とする変数への割当が存在するかどうかを判定する問題である．例えば，次の式 1 は CNF を真にする割当があるので充足可能であるといい，式 2 は真にする割当が存在しないので，充足不能という．

1. $(x_1 \vee x_2) \wedge (\bar{x_1} \vee x_2) \wedge (x_1 \vee \bar{x_2})$
2. $(x_1 \vee x_2) \wedge (\bar{x_1} \vee x_2) \wedge (x_1 \vee \bar{x_2}) \wedge (\bar{x_1} \vee \bar{x_2})$

CNF の各節に制限がない SAT を CNF-SAT といい，各節に現れる変数が高々$k$ 個である CNF-SAT を $k$-SAT という．特に $k = 3$ の場合，表 2，3 の通りアルゴリズムの計算時間の改良が盛んに行われている．表は，$m$ を節の数として，$O(\mathrm{poly}(m) \cdot f(n)) = f(n) \cdot 2^{O(\log m)}$ の $f(n)$ 部分を記載している．紙面のスペースの関係上，文献情報は省略させてもらう．

表 2: 3-SAT における決定性アルゴリズムの改良の歴史

| 計算時間 | 年 | 著者 |
|---|---|---|
| $2^n$ | | 全探索 |
| $1.839^n$ | 1979 | B. Monien and E. Speckenmeyer |
| $1.769^n$ | 1983 | E. Dantsin |
| $1.618^n$ | 1985 | B. Monien and E. Speckenmeyer |
| $1.579^n$ | 1992 | I. Schiermeyer |
| $1.497^n$ | 1996 | I. Schiermeyer |
| $1.476^n$ | 1996 | R. Rodosek |
| $1.473^n$ | 2004 | T. Brueggemann and W. Kern |
| $1.465^n$ | 2008 | D. Scheder |
| $1.334^n$ | 2010 | T. R. Moser and D. Scheder |
| $1.3303^n$ | 2011 | K. Makino, S. Tamaki, and M. Yamamoto |
| $1.32793^n$ | 2018 | S. Liu |

表 3: 3-SAT における乱択アルゴリズムの改良の歴史

| 計算時間 | 年 | 著者 |
|---|---|---|
| $1.588^n$ | 1997 | R. Paturi, P. Pudlák, and F. Zane |
| $1.362^n$ | 1998 | R. Paturi, P. Pudlák, M. E. Saks, and F. Zane |
| $1.334^n$ | 1999 | T. Hofmeister, U. Schóning, R. Schuler, and O. Watanabe |
| $1.3302^n$ | 2002 | D. Rolf |
| $1.3290^n$ | 2003 | S. Baumer and R. Schuler |
| $1.32793^n$ | 2003 | D. Rolf |
| $1.3238^n$ | 2004 | K. Iwama and S. Tamaki |
| $1.32216^n$ | 2005 | D. Rolf |
| $1.32113^n$ | 2010 | K. Iwama, T. Takai, K. Seto, and S. Tamaki |
| $1.32065^n$ | 2010 | T. Hertli, R. Moser, and D. Scheder |
| $1.30704^n$ | 2011 | T. Hertli |

さて，表を見てお気づきの通り，指数の肩の部分がすべて $n$ となっている．表の計算時間は，ある定数 $\delta$ $(0 < \delta \leq 1)$ を用いて，$2^{\delta n + O(\log m)}$ と表すことができる．つまり，現状，$k$-SAT を $2^{n^{0.99} + O(\log m)}$ で解くアルゴリズムは存在しない．簡単に言うと，このようなアルゴリズムは存在しないという予想が ETH である．

厳密には，$s_k$ を次の通り定義する．

$$s_k = \inf\{\delta_k \mid k\text{-SAT を解く } O^*(2^{\delta_k n}) \text{ 時間アルゴリズムが存在する}\}$$

このとき，

$$\delta_3 \leq \delta_4 \leq \cdots \leq \delta_k$$

が成り立つ．ETH の主張は次の通りである．

**Exponential Time Hypothesis**

$$s_3 > 0$$

$k$-SAT は，E. Dantsin, A. Goerdt, E. A. Hirsch, R. Kannan, J. Kleinberg, C. Papadimitriou, P. Raghavan, U. Schöning [3] による下記の計算時間のアルゴリズムが知られている．このことから，$k$ が定数である限り，$\delta_k < 1$ を満たすことがわかる．

$$\left(2 - \frac{2}{k+1}\right)^n$$

また，SETH は次式が成り立つという仮定である．

**Strong Exponential Time Hypothesis**

$$\delta_\infty = 1$$

つまり，SETH は CNF-SAT を全探索よりも指数的に高速に解くアルゴリズムは存在しないことを意味している．ここで注意すべきは，$2^n/\omega(n^c) = 2^{n-\omega(\log n)}$ のような計算時間のアルゴリズム（いわゆる，全探索よりも超多項式時間高速なアルゴリズム）の存在は否定していない．

さて，ここで表 1 に戻ってみると，Ryan Williams は ETH が正しい可能性は 70%，SETH が正しい可能性は 25%と考えている．どちらも，P ≠ NP の 80%よりも低く見積もられている．実際，P ≠ NP を証明するためには，3-SAT を解くためには $n^{\omega(1)}$ 時間が必要であることを示せばよい．ETH や SETH を肯定的に解決することは，P ≠ NP を示すことにもつながる非常に強い仮説であり，当然といえば当然の見積もりである．みなさんはどう思われるであろうか？

私は SETH は正しくないと信じている．その根拠があるかと言われると，実はない．なんとなく信じられないのである．ただ単に信じられないと言っているだけでは何も進展しないし，幸いにも博士号を取得した 2010 年から SAT の研究を続けていることもあり，2 年前に実際に反証に向けて真剣に研究しようと思い始めた．そこで，科研費に「強指数時間仮説の反証に向けた研究」という題目で申請し，幸いにも採択され，SETH に関する研究を行なっている．

## 2　SETH を仮定したアルゴリズムの改良困難性

この数年で SETH を仮定すると他の問題に対するアルゴリズムの改良困難性が多数示されているが，これらは他の問題のアルゴリズムを改良することで SETH を反証することができることを示している．その中から代表的な結果を 2 つ紹介したい．最初に紹介するのは，2010 年の SODA で発表された M. Pătraşcu と R. Williams の結果である．

M. Pătraşcu and R. Williams ([6])

次の命題は同値である．

- ある定数 $\delta < 1$ が存在し，CNF-SAT は $2^{\delta n + O(\log m)}$ 時間で解くことができる．
- ある定数 $\epsilon > 0$ が存在し，$k$-Dominating Set ($k \geq 3$) は $O(n^{k-\epsilon})$ 時間で解くことができる．
- ある定数 $\epsilon > 0$ が存在し，2-SAT+2Clauses ($m = n^{1+o(1)}$) は $O(n^{2-\epsilon})$ 時間で解くことができる．
- ある定数 $\epsilon > 0$ が存在し，HornSAT+$k$Clauses ($k \geq 2$) は $O((n+m)^{k-\epsilon})$ 時間で解くことができる．

各問題について説明する．Dominating Set とは与えられたグラフの頂点部分集合 $S$ のうち，すべての頂点が $S$ に属するか $S$ 内の頂点に隣接するような部分集合のことである．$k$-Dominating Set とは，与えられたグラフに $k$ 個の頂点からなる Dominating Set があるかどうかを判定する問題である．これには，$k \geq 7$ のとき，$n$ 頂点グラフの $k$-Dominating Set を $n^{k+o(1)}$ 時間で求めるアルゴリズムが示されている．2-SAT+2Clauses とは，$n$ 変数，$m$ 節の 2-CNF に 2 つの任意の長さの節を加えた充足可能性問題であり，$O(mn + n^2)$ で解くことができる．HornSAT+$k$Clauses とは，$n$ 変数，$m$ 節の Horn CNF（全ての節が高々1 つの正リテラルしか含まない CNF）に $k$ 個の任意の長さの節を加えた充足可能性問題であり，$O(n^k(m+n))$ 時間で解くことができる．

この結果は，SETH を仮定すれば，$k$-Dominating Set，2-SAT+2Clauses，HornSAT+$k$Clauses 問題は現在知られている最良のアリゴリズムを本質的に改良することはできないことを示している．この結果の面白いところは，指数時間に関する仮説である SETH を仮定することで，多項式時間アルゴリズムの改良困難性を示していることである．

次に紹介するのは，2012 年の CCC で発表された M. Cygan, H. Dell, D. Lokshtanov, D. Marx, J. Nederlof, Y. Okamoto, R. Paturi, S Saurabh, M. Wahlström の結果である．

M. Cygan et al. ([1])

次の命題は同値である．

- 任意の $\delta < 1$ に対し，ある $k$ が存在し，$k$-SAT は $2^{\delta n + O(\log m)}$ 時間で解くことができない．
- 任意の $\delta < 1$ に対し，ある $k$ が存在し，$k$-Hitting Set は $O(2^{\delta n})$ 時間で解くことができない．
- 任意の $\delta < 1$ に対し，ある $k$ が存在し，$k$-Set Splitting は $O(2^{\delta n})$ 時間で解くことができない．
- 任意の $\delta < 1$ に対し，ある $k$ が存在し，$k$-NAE SAT は $O(2^{\delta n})$ 時間で解くことができない．

各問題について説明する．Hitting Set とは，与えられたサイズ $n$ の集合 $U$ の部分集合からなる集合族 $S = \{S_1, S_2, \ldots, S_m\}$ が与えられる．任意の $i$ ($1 \leq i \leq m$) に対して，$C \cap S_i \neq \emptyset$ となる高々サイズ $k$ の集合 $C$ を求める問題である．$k$-Set Splitting とは，与えられたサイズ $n$ の集合 $U$

のサイズ $k$ の部分集合からなる集合族 $S = \{S_1, S_2, \ldots, S_m\}$ が与えられる．任意の $i$ $(1 \leq i \leq m)$ に対して，$V \cap S_i \neq \emptyset$ かつ $W \cap S_i \neq \emptyset$ となる $U$ の分割 $V, W$ が存在するかを求める問題である．$k$-NAE SAT とは $k$-SAT とほぼ同じであるが，充足解のうち各節の 1 つ以上のリテラルを false にする充足解が存在するかどうかを判定する問題である．上記の結果が示していることは，SETH が正しければ，Hitting Set,，Set Splitting，NAE SAT にも全探索よりも指数的に高速なアルゴリズムが存在しないことを示している．

この結果は 1 つ目に紹介した結果とは違い NP 完全問題間での関係性を表している．この証明には Reduction が使われるのであるが，NP 完全性を示す時に使われる Reduction よりも非常に制約が厳しい．NP 完全問題の証明では，元の問題のサイズが $n$ であるときに，Reduction 先の問題のサイズが $n$ の多項式に膨れあがっても特に問題はない．しかし，この Reduction においては，Redution 先の問題のサイズが $n + o(n)$ を満たす必要があり，サイズの増加を最小限に抑える Reduction を考えないといけないところが非常にテクニカルで面白い．

# 3　SETH を反証することは可能か？

SETH を反証する 1 つの方法は直接，CNF-SAT における $O(2^{0.99\ldots n})$ 時間アルゴリズムを構築することである．現在，最良の CNF-SAT アルゴリズムの計算時間は C. Calabro, R. Paturi, R. Impagliazzo [2] や E. Dantsin と E. A. Hirsh [4] が示した下記の数値である．$n$ は変数の数，$m$ は節の数である．

$$2^{\left(1-\frac{1}{\log(m/n)}\right)n+O(\log m)}$$

節の数が $O(n)$ であるとき，$O(2^{0.99\ldots n})$ 時間アルゴリズムとなっているが，節の数が $\omega(n)$ のときには，SETH を破ることはできない．

もう 1 つの方法は，2 節で紹介したような SETH に関連する他の問題のアルゴリズムの計算時間を改良することである．例えば，$k$-Dominating Set を解く $O(n^{0.99k})$ 時間アルゴリズムや Hitting Set を解く $O(2^{0.99n})$ 時間アルゴリズムの構築ができれば，SETH を反証することができる．

どちらが有力なアプローチか明確な答えはない．しかし，多項式時間アルゴリズム，指数時間アルゴリズムを問わずに SETH を反証するための候補となる問題は非常に多い．それでは，実際に反証できるのだろうか？

これまで，私の研究では，Max-SAT の $O(2^{n-n^\epsilon})$ 時間アルゴリズム，$cn$ サイズの $B_2$-Formula-SAT の $O(2^{(1-1/2^{O(c^3)})n})$ 時間アルゴリズム，Read-$k$-times BP-SAT の $O(2^{(1-1/4^k)n})$ 時間アルゴリズムなど様々な SAT アルゴリズムを構築してきた．SAT の研究をすればするほど，SETH の壁を強く感じるようになった．しかし，壁を感じれば感じるほど，その壁を壊したくなるものである．何より，SETO と SETH が 1 文字違いなので愛着を感じずにはいられないのである．私は SETH は反証できると信じている．

SETH を反証すると，これまでの SETH を仮定していた論文の結果がゼロになるのでは？と思う人がいるかもしれないが，それは事実ではない．その証明で使われた様々な技法は残り続け，発展し続けるであろう．SETH を仮定して，困難性を証明している人たちは SETH を信じておらず，反証をするための様々なアプローチ方法を提供しているとすら私は考えている．

# 4 おわりに

この記事により，SETH を少しでも多くの人に知ってもらい，少しでも多くの人に興味を持ってもらえれば幸いです．みなさんには，あたかも研究に集中しているような印象を与えてしまっているかもしれませんが，実はこの 2 年，重要な学内業務に携わっているため，集中した時間をとれていないのが現状です．2020 年 3 月にはその業務も終わり，4 月からは集中した時間を取りやすくなるため，SETH の反証にますます真剣に取り組んでいきたいです．

今年度の夏の LA シンポジウムに数年ぶりに参加して，LA シンポジウムの楽しさを再認識しました．そのような楽しい会議を運営してくださった事務局のみなさんに感謝しております．これからも出来る限り参加して，発表もしていきたいと思いました．最後になりましたが，会誌編集担当の大舘先生にこのような機会を与えていただいた御礼を申し上げます．

# 参考文献


[1] Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, Jesper Nederlof, Yoshio Okamoto, Ramamohan Paturi，Saket Saurabh，Magnus Wahlström. On Problems as Hard as CNF-SAT. ACM Transactions on Algorithms, 12(3):41:1–24, 2016.

[2] Chris Calabro, Russell Impagliazzo, and Ramamohan Paturi. A Duality between Clause Width and Clause Density for SAT. Proceedings of the Twenty-First Annual IEEE Conference on Computational Complexity, pages 252–260, 2006.

[3] Evgeny Dantsin, Andreas Goerdt, Edward A. Hirsch, Ravi Kannan, Jon Kleinberg, Christos Papadimitriou, Prabhakar Raghavan, Uwe Schöning. A deterministic $(2 - 2/(k+1))^n$ algorithm for $k$-SAT based on local search. Theoretical Computer Science, 289(1):69–83, 2002.

[4] Evgeny Dantsin and Edward A. Hirsch. Worst-Case Upper Bounds. In Handbook of Satisfiability, Armin Biere, Marijn Heule, Hans van Maaren and Toby Walsch (eds.), pages 341–362, 2008.

[5] Russell Impagliazzo and Ramamohan Paturi. Complexity of $k$-SAT. Journal of Computer and System Sciences, 62(2):367–375, 2001.

[6] Mihai Pătraşcu and Ryan Williams. On the Possibility of Faster SAT Algorithms. Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, pages 1065–1075, 2010.

[7] Ryan Williams. Some Estimated Likelihoods for Computational Complexity. Computing and Software Science, LNCS 10000, pages 9–26, 2019.