International Journal of Pure and Applied Mathematics
Volume 94 No. 1 2014, 81-88
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v94i1.9

A SHORT PROOF THAT NP IS NOT P

Viktor Ivanov

831 Grove St. N., Saint Petersburg
FL 33701-2213, USA

Abstract: This proof of $NP \neq P$ is based on better estimates of lower bounds on the time complexity that hold for all solution algorithms. Almost no special knowledge other than logical and combinatorial efforts is needed to understand the proof.

AMS Subject Classification: 03D10, 03D15, 03D9
Key Words: algorithm time complexity, information-based complexity, problem NP-complete, P-problem, turing machine

1. Introduction

There are hundreds of important so-called NP-complete problems from various areas of mathematics [7]. If one of those problems might be solved for a polynomial time on Deterministic Turing Machines (DTM), then all of them would also be solved for a polynomial time, i.e., $NP = P$. So, any sub-problem of NP-complete problem and/or any other problem obtained for a polynomial time, using NP-complete problem, will also be NP-complete. The problem P VERSUS NP was formulated by Professors Cook and Levin in 1971. Most computer scientists believe that $NP \neq P$. Below is a short proof that $NP \neq P$, which is accessible for a wide and diverse audience. Discussion of certain results on effective non-algorithmic solution of hard problems can be seen in [4] and [6]. We consider the proof using a typical NP-complete problem P_{IS} of INDEPENDENT SET (IS) [2]. INSTANCE (or witness or input): graph $G = (V, E)$ with n vertices V, positive integer t. **Question:** Does G contain an independent

Received: April 3, 2014

set of size t, i.e., a subset $V' \subseteq V$ such that $|V'| = t$ and such that no two vertices in V' are joined by an edge in E? Let the representation of the graph $G = (V, E)$ be an ordered list of binary words

$$I_j = a_{1j}, \ldots, a_{j-1j}, j = 2, \ldots, n; a_{ij} = 0 \vee 1, 1 \leq i < j \leq n, \qquad (1)$$

where $a_{ij} = 1$, if and only if there is an edge from the note i to the node j, $i < j$. The respective input for this initial problem has the size order n^2. We introduce the binary words

$$A_{k_1,\ldots,k_t} = a_{k_1k_2}a_{k_1k_3} \cdots a_{k_1k_t}a_{k_2k_3}a_{k_2k_4} \cdots a_{k_2k_t} \cdots$$
$$a_{k_{t-2}k_{t-1}}a_{k_{t-2}k_t}a_{k_{t-1}k_t}, 1 < t < n, \qquad (2)$$

where $(k_1, \ldots, k_t)$ are arbitrary ordered combinations from $(1, \ldots, n)$ by t, which is the same as $1 \leq k_1 < \cdots < k_t \leq n$. Let the problem P_1 be the problem P_{IS} with the inputs (1), $1 < t \leq n$, and let P_2 with the same inputs be the problem $A_{k_1,\ldots,k_t} =? \mathbf{0}$, for any t and n, $1 < t \leq n$. It is clear that $A_{k_1,\ldots,k_t} = \mathbf{0}$, if and only if $(k_1, \ldots, k_t)$ is IS, and $A_{k_1,\ldots,k_t} \neq \mathbf{0}$, if and only if at least one of those $(t-1)t/2$ digits is not 0.

2. General-Type Approach

Using standard notions of computer science (see, e.g., [1], [2], [3]), the results of the author employ a novel general-type approach that can be described on qualitative level as consisting of three main parts. The first part consists of construction of a combinatorial lemma for a problem on the whole input domain. A sense of that lemma is a rather detailed spectrum of witnesses for the answer 1, true and 0, false. Having for both 1 and 0 exponential numbers of different witnesses from the input domain of a problem, we can conclude that at least one of those witnesses is such that to find its output without checking it, exponential time is required. Otherwise, we can use an exponential number of different witnesses for a polynomial time, which is impossible. Not counting that witness, we can conclude the existence another one, and so on, until we come to the existence of an exponential number of depersonalised witnesses. For each of those witnesses, any Polynomial Solution Algorithm (PSA) has to be realized to find the answer. The third part consists of an estimation of the time bounds for any PSA based on combinatorial, logical and information-based structure [10] of a given problem with regard to the whole input domain

3. Supporting Results

The time complexity of the solution of the problem P on a DTM is given by

$$T(P) = T(I, R, A) = T_n(I, R, A) = inf_A sup_x(x, y, A)(|x| = n, x \in I), \quad (3)$$

where $t(x, y, A)$ is the time for a solution algorithm A on input $x \in I$ with output $y \in R$ on a DTM. Let L_n be the set of all 2^n binary words. For any $I \in L_n$, let I' be the ordered set consisting of zero numbers in $I(\mathbf{1} = (1 \cdots 1)$ corresponds to the empty set), and $\underline{I}$ be the compliment of I to $\mathbf{0} = (0 \cdot 0) \in L_n$. For any words I_1 and I_2 from L_m and L_n, $1 \leq m \leq n$, let the intersection $I_1 \cap I_2$ (the union $I_1 \cup I_2$) be a word from $L_m(L_n)$ combining all common 0-digits (all 0-digits) of I_1 and I_2. This definition of intersection and union for binary words corresponds to the ordinary definition of intersection and union for the respective sets I_1' and I_2'. If, for example, $I_1 = (010), I_2 = (100)$, then $I_1 \cap I_2 = (110)$ and $I_1 \cup I_2 = (000)$, since $I_1' = (1, 3), I_2' = (2, 3)$, then $I_1' \cap I_2' = (3)$ and $I_1' \cup I_2' = (1, 2, 3)$.

Result 1. *While each of $X_k, k = 1, \ldots, t$, runs all 2^n words from L_n, $Y_t = \cap_1^t X_k$ runs each of its all $\begin{pmatrix} n \\ r \end{pmatrix}$ words with r 1-digits $(2^t - 1)^r$ times, $1 < t < n, 0 \leq r \leq n$. In particular, for arbitrary t independent of n, we have*

$$|[Y_t = \mathbf{1}]| = (2^t - 1)^n, |[Y_t \neq \mathbf{1}]| = 2^{tn} - (2^t - 1)^n \asymp 2^{tn}, \quad (4)$$

where for any relation $R(X)$, $X = (X_k, k = 1, \ldots, t)$, the notation $|[R(X)]|$ means the number of all different values of X such that $R(X)$ is valid.

Proof. In case of $Y_t, t = 2, (2^2 - 1)^r = (2 + 1)^r = \sum_0^r \begin{pmatrix} r \\ s \end{pmatrix} 2^s, r = 0, 1, \ldots, n$, since for each of $\begin{pmatrix} r \\ s \end{pmatrix}$ words X_1 with s 1-digits, X_2 can generate all possible 2^s words with fixed $r - s$ 1-digits corresponding to 0-digits of X_1. For the other t, the proof can be obtained by the mathematical induction. Using $Y_{s+1} = Y_s \cap X_{s+1}$, we have

$$(2^{s+1} - 1)^r = \sum_0^r \begin{pmatrix} r \\ k \end{pmatrix} (2^{s+1} - 2)^k \sum_0^r \begin{pmatrix} r \\ k \end{pmatrix} 2^k (2^s - 1)^k, r = 0, 1, \ldots, n, \quad (5)$$

where the right side means that for each k 1-digits of Y_s (each is running $(2^s - 1)^k$ times by premise), X_{s+1} can generate all possible 2^k words with fixed $r - k$ 1-digits corresponding to 0-digits of I_s.

Result 2. *Let* $X_{1k_1},\dots,X_{tk_t},k_1,\dots,k_t=t,\dots,n$, *be arbitrary words from* $L_n,n=m2^t,t>0$ *be even integer independent of* n, *and let*

$$Y_{k_1,\dots,k_t}=X_{1k_1}\cap\dots\cap X_{tk_t}=J_p,k_t=1,\dots,n,k_{t-1}\in X'_{tk_t},$$
$$k_{t-2}\in X'_{t-1k_{t-1}}\cap X'_{tk_t},\dots,k_1\in Y'_{k_2,\dots,k_t},p=1,\dots,P. \quad (6)$$

Then the number P *of different words* $Y_{k_1,\dots,k_t}$, *for which* $|[J_p=\mathbf{1}|J_q=\mathbf{1},q=1,\dots,P-1]|$ *is not less than* 1, *can be not less than* $n^{t/2}$.

Proof. Sets X'_{tk_t} can be arbitrary n from $\begin{pmatrix} n \\ n/2 \end{pmatrix}$ subsets of size $n/2$; $X'_{t-1k_{t-1}}\cap X'_{tk_t}$ can be n^2 different subsets of size $n/2^2$, n of them in each of sets $X'_{tk_t};\dots;Y'_{k_1,\dots,k_t}$ can be n^t different subsets of size $n/2^t=m$, n of them in each of subsets $Y'_{k_2,\dots,k_t}$, number of which of size $2m$ is n^{t-1}. There exists $2^{n/2^{t/2}}$ independent cases when $Y_{k_1,\dots,k_{t/2}}=\underline{Y_{k_{t/2+1},\dots,k_t}}$. Notion of independet here means that any one of cases is not a consequence of the others (see [1], pp. 343, 344). So, even if all cases in $|[J_p=\mathbf{1}|J_q=\mathbf{1},q=1,\dots,P-1]|$ are part of those whole independent ones, there still remain $2^{n/2^{t/2}}-(tn)^2$ independent cases, any $n^{t/2}$ of which can be the desired ones. Note that due to RESULT 1, we have for $|[Y_2=1]|=n^{t/2log3}$, so that $t/2log3$ is the upper estimate. Note more that $n^{t/2}$ sets of size $n/2^{t/2}$ means n^x sets of size n, where $n^x=n^{t/2}/2^{t/2}$.

Result 3. *Under the conditions and notations of RESULT 2, on any DTM*

$$T(Q_{n,t})\geq n^{t/2},Q_{n,t}=\cup_{p=1}^{P}J_p,n=m2^t. \quad (7)$$

Proof. It is clear that the union $Q_{n,t}$ in (7) is $\mathbf{1}$, if and only if all its terms are $\mathbf{1}$, and that union is not $\mathbf{1}$, if and only if at least one of its terms is not $\mathbf{1}$. So, we can describe the problem $Q_{n,t}=?\ \mathbf{1}$ as the problem $J_1=?\ \mathbf{1}$, and if $J_1=\mathbf{1}$, then $J_2=?\ \mathbf{1}|J_1=\mathbf{1}$, and so on until $J_p=?\ \mathbf{1}|J_q=\mathbf{1},q=1,\dots,p-1;p=2,\dots,P$. We show that all alternatives $J_1=?\ \mathbf{1},J_p=?\ \mathbf{1}|J_q=\mathbf{1},q=1,\dots,p-1;p=2,\dots,P$, can be independent of each other (see [1], p. 1103). It is possible that $P=n^{t/2}$ and

$$|[J_1=\mathbf{1}]|=(2^t-1)^n2^{tn(n-1)},|[J_1\neq\mathbf{1}]|=2^{tn\cdot n}-(2^t-1)^n2^{tn(n-1)},$$
$$|[J_p\neq\mathbf{1}|J_q=\mathbf{1},q=1,\dots,p-1]|\geq 2^{tn\cdot n}-P(2^t-1)^n2^{tn(n-1)},$$
$$1\leq|[J_p=\mathbf{1}|J_q=\mathbf{1},q=1,\dots,p-1]|\leq(2^t-1)^n2^{tn(n-1)},p=2,\dots,n^{t/2}. \quad (8)$$

The first two relations in (8) are a consequence of RESULT 1. The third relation is a consequence of the upper estimate in the fourth one. The result

$P = n^{t/2}$ and the lower estimate in the fourth relation follow from RESULT 2. Thus, for solving the problem $Q_{n,t} =? \mathbf{1}$, any solution algorithm A might be checking all $n^{t/2}$ two-sided independent alternatives in (8), and hence the time might be not less than $n^{t/2}$. Given the initial data I: arbitrary words $X_{sk_s}, k_s = 1, \ldots, n; s = 1, \ldots, t$, from $L_n, n = m2^t$, t is independent of n, let A be any algorithm, using these data, with the result $Q_{n,t} = \mathbf{1}$. Then (8) is also fulfilled, and hence A solved also the problem $Q_{n,t} = \mathbf{1}$. It means that $T(I, Q_{n,t} =? \mathbf{1}, A) \geq T(I, Q_{n,t} = \mathbf{1}, A) \geq n^{t/2}$.

Result 4. *The following relations are valid*:

$$A_{k_1 \ldots k_t} = \mathbf{0}, 1 \leq k_1 < \cdots < k_t \leq n, n \geq t > 1, \tag{9}$$

if and only if $Q_{n,t-1} \neq \mathbf{1}$.

Proof. A proof of (9) is based on a simple fact that $k_r \in I'_{k_{r+1}}$, if and only if $a_{k_r k_{r+1}} = 0$, and hence the intersections $I_{k_{r+1}} \cap \cdots \cap I_{k_t} \neq \mathbf{1}, r = 1, \ldots, t-1$, if there exist $k_r \in I'_{k_{r+1}} \cap \cdots \cap I'_{k_t}$ such that $a_{k_r k_{r+1}} + \cdots + a_{k_r k_t} = 0, r = 1, \ldots, t-1$, and hence the respective $A_{k_1 \ldots k_t}$ is also equal to $\mathbf{0}$. Thus, if $I_{k_2} \cap \cdots \cap I_{k_t} \neq \mathbf{1}$, then there exist $k_t \in (t, \ldots, n), k_{t-1} \in I'_{k_t}, k_{t-2} \in I'_{k_{t-1}} \cap I'_{k_t}, \ldots, k_2 \in I'_{k_3} \cap \cdots \cap I'_{k_t}$, and $k_1 \in I'_{k_2} \cap \cdots I'_{k_t}$, for which $A_{k_1 \ldots k_t} = \mathbf{0}$. If $A_{k_1 \ldots k_t} = \mathbf{0}$, then all $a_{k_r k_s} = 0, r = 1, \ldots, t-1; s = r+1, \ldots, t$, and hence the respective $I_{k_2} \cap \cdots \cap I_{k_t} \neq \mathbf{1}$.

Result 5. *There exist exponential number inputs, for each of which and any t independent of n, on any DTM*

$$T(A_{k_1 \ldots k_t} =? \mathbf{0}, 1 \leq k_1 < \cdots < k_t \leq n) = T(Q_{n,t-1} =? \mathbf{1}) \geq n^{(t-2)/2}. \tag{10}$$

Proof. The estimate in (10) is a consequence of the estimate in 7.

4. The Main Result

Theorem. $NP \neq P$.

Proof. Let $NP = P$. Then the problem P_2 is also in P. So, there exists a constant C independent of n, t such that the time of solution of P_2 does not exceed n^C. But, due to RESULT 5, there exist exponential number inputs, for each of which, checking all $A_{k_1 \ldots k_t}$ can require the time not less than $n^{(t-2)/2}$, which can be more than n^C.

5. Discussion

There is D. Johnsons famous statement: *A common failing in P vs. NP proofs is the step in which the author says (without proof), any algorithm for solving this problem must do it in the following way.* Our proof above does not have A common failing. Indeed, on the one hand, when the problem in question is given by special particular way and for each of exponential number inputs, in the case of certain weaker problem, there exist not less than $n^{(t-2)/2}$ multiple two-sided independent alternatives (under fixed t), and any solution algorithm must solve all those alternatives, the required time on any DTM can be not less than $n^{(t-2)/2}$ (see the proofs of RESULTS 3 and 5). On the other hand, if any algorithm solved the weaker problem and this implies that the same problem given by special way is also solved, then the time on any DTM in any case cannot be less than $n^{(t-2)/2}$. There are more traps for the author to avoid. Another well-known example of such a trap is the so-called natural proof [9]. The possible answer to this example: there is an important detail of the proof in question that is absent in any natural proof. Indeed, there exist exponential number of inputs for each of which the solution time on any DTM is not less than $n^{(t-2)/2}$, but we cannot indicate naturally any one of them (this evidence is based on Kolmogorov complexity and investigated in [8]).

Actually all above was submitted to JACM of June 17 2013. Nine month later the author received from Dr. Boaz Barak, Associate Editor of JACM, the message on his paper rejection based on the following review. Referee Comments in cursive and the author respective responses are below: Referee: 1 Recommendation: *Reject*. Why? Comments: *This paper claims to prove that P is not NP. The paper claims to construct a problem in NP that cannot be solved in polynomial time. However, the results are not clearly stated in the sense that there is no clear definition of a problem B and proof that: (1) B is in NP (2) B is not in P.* It is not true. Proof: The clear definition of a problem B as the NP-problem P_2 is given in Introduction on the page 1, and the paper is devoted to a proof that namely this B is not in P. It is also shown there that problem P_2 is a NP-complete because its solution gives the solution of the problem P_1, which is NP-complete Independent Set Problem.

The problem the author is considering seems to be implicitly defined in "Result 2", though it is not clear how it is defined and what it means. It seems that the problem is defined based on t parameters, though there is a claim in Result 3 that there are $n^{t/2}$ independent parameters. It is also not true. Proof: Result 2 is one of five supporting results (see Section 3), all of which yields a proof that the problem P_2 is not in P. The sense of the Result 2 can be seen in the

proof of the Result 3.

It is not at all clear the statements of the results make sense, let alone the proofs. In particular, it seems that the approach fails for exactly the reason discussed in Section 5 - there is an assumption that since there are exponentially many witnesses, any algorithm must check all of them. It is all not true. Proof: All results in this paper are mathematically formulated and rigorously proven. In particular, it is proven that there exists exponential number witnesses, for each of which and any t independent of n, on any DTM, the time $T(P_2) \geq n^{(t-2)/2}$, which is a super-polynomial time. The main reason is the proven fact that in the case of the problem P_2, for each of those witnesses, any solution algorithm must check $n^{(t-2)/2}$ independent two-sided alternatives. It is possible that all checking gives the negative result, that is the time can be not less than $n^{(t-2)/2}$.

By this reasoning, it should also be possible to show that one cannot solve linear equations modulo 2 in polynomial time, since in that case one can also have exponentially many solutions. It is completely wrong, since there are no super-polynomial number independent two-sided alternatives in the case of linear equations modulo 2. For any problem in P, such number of independent two-sided alternatives is not possible because otherwise a problem would not be in P.

Additional Question: Most ACM journal papers are researcher-oriented. Is this paper of potential interest to developers and engineers? *No* It is not true. Proof: There is Section 2 devoting to a novel general-type approach that allows developers and engineers to construct and ground lower bounds closed to optimal ones for many problems in discrete mathematics and computer science. Namely that approach allows the author to prove NP is not P. The readers should know that this paper claims to solve one of Millennium problem, *That is the most mind-boggling problem facing theoretical computer science and maybe all of the science at the moment* (Donald Knuth).

The author couldn't agree with the referee review and the decision of Dr. Barak. His appeal with copy of the paper [5] to Editor-in-Chief of JACM Professor Victor Vianu and Co-Chairs of Publication Board of ACM Professors Jack Davidson and Joseph Konstan were all in vain: they confirmed the decision of Dr. Barak without any proof.

So, let the mathematical community and any reader decides who and what are right.

References

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to Algorithms*, The MIT Press (2001)

[2] M. R. Garey, D. S. Johnson, *Computers and Intractability. A Guide to the Theory of NP-Completeness*, Freeman and Co. (1979)

[3] J. E.Hopcroft, J. D. Ullman, *Introduction to Automata Theory, Language, and Computations*, Addison Wesley (1979)

[4] V. V. Ivanov, *Global minimum formula and its implications*, Proceeding of International Symposium Problem of Optimization of Computations, National Academy of Ukraine, Vol. 1, (2009), 273-278.

[5] V. V. Ivanov, *Ethics in Mathematics*, Notices of the AMS, Vol. 60, Number 2,(2013), 152.

[6] V. V. Ivanov, N. V. Ivanova, *Mathematical Models of the Cell and Cell Associated Objects*, Elsevier (2006)

[7] D. S, Johnson, *The NP-completeness column: an ongoing guide*, J. Algorithms, starting with the first edition, 393-405, and continuing in ACM Transactions on Algorithms (1981-2006)

[8] M. Li, P. Vitanyi, *An Introduction to Kolmogorov Complexity and Its Applications*, Springer-Verlag (2008)

[9] A. A. Rasborov, S. Rudich, *Natural Proofs*, Proceeding of 26th Annual Symposium on the Theory of Computing, (1990), 204-213.

[10] J. F. Traub, H. Wosniakowski, *General Theory of Optimal Algorithms*, ACM monograph ser. (1980).