How Hard Are n^2-Hard Problems?

Stephen A. Bloch* Jonathan F. Buss† Judy Goldsmith‡

Abstract

Many of the "n^2-hard" problems described by Gajentaan and Overmars can be solved using limited nondeterminism or other sharply-bounded quantifiers. Thus we suggest that these problems are not among the hardest problems solvable in quadratic time.

Introduction

One approach to determining the difficulty of computational problems is completeness—*NP*-completeness, exponential-time completeness, etc. Completeness can also be defined for classes inside P, but few problems have been classified in this way. Hence we were excited to discover in the previous Computational Geometry column [5] a discussion of n^2-hard problems. We discuss here a more precise bound on the complexity of these problems.

Gajentaan and Overmars [4] define a problem to be n^2-*hard* if the following base problem can be reduced to it.

> SumOfThree: Given three sets A, B, and C of at most n integers, are there $a \in A$, $b \in B$ and $c \in C$ such that $a + b + c = 0$?

Although this problem may indeed require $\Omega(n^2)$ time, we believe it is probably not hard for quadratic time in the usual sense that all problems in quadratic time have linear-time (or "almost linear-time") reductions to it. We give a simple proof that if SumOfThree is hard for quadratic time in this general sense, then $P \neq NP$.

Algorithms and Structural Results

An instance of SumOfThree can be solved in quadratic time as follows. Sort B in increasing order and C in decreasing order. For each a in A, scan B and C to check for a pair with sum $-a$. When the sum is too small, advance in B; when the sum is too large, advance

*Department of Computer Science, University of Kentucky, Lexington, KY 40506-0027, USA. Electronic mail: `sbloch@ms.uky.edu`.

†Department of Computer Science, University of Waterloo, Waterloo, Ontario, CANADA N2L 3G1. Electronic mail: `jfbuss@math.uwaterloo.ca`.

‡Department of Computer Science, University of Kentucky, Lexington, KY 40506-0027, USA. Electronic mail: `goldsmit@ms.uky.edu`.

in C. Each a is thus tested in a linear number of arithmetic operations, and the total time is quadratic. No significantly better deterministic algorithm is known.

A nondeterministic algorithm can solve SumOfThree in $O(n \log n)$ time by guessing a correct value of a (if any), and then checking it with a linear number of arithmetic operations. Because the set A has linear size, the algorithm uses only $\log n$ bits of nondeterminism.

In an analogous way, the following languages, shown to be n^2-hard by Gajentaan and Overmars, can be accepted in $O(n \log n)$ time using $\log n$ bits of nondeterminism: GeomBase, 3-Points-on-Line, Point-on-3-Lines, Separator1, Separator2, Hole-in-Union, Point-Covering, Visibility-between-Segments, Visibility-from-Infinity, and Visible-Triangle. In each case, the required witness is either the x-coordinate of a point or the slope of a line. (In most cases, testing a witness requires a linear number of arithmetic operations; in some cases it also requires sorting intersection points along a line or sorting lines by slope.) The complements of the n^2-hard languages Strips-Cover-Box and Triangles-Cover-Triangle can also be accepted in this way.

Buss and Goldsmith [3] defined the classes N^iP_j, for $i \geq 0$ and $j \geq 1$, to be the set of languages acceptable using $i \log n$ bits of nondeterminism and time $O(n^j \log^k n)$, for any fixed k, on a multi-tape Turing machine. For $j = 1$, the classes are termed *quasilinear*. All of the above problems (or complements) are in N^1P_1. The polylogarithmic factors in the definition allows some degree of robustness in the classes; in particular, they are closed under quasilinear-time reductions.[1] The related class NP_1, quasilinear time with no additional bound on the nondeterminism, was introduced by Schnorr [6], who showed that SAT is complete for NP_1 under quasilinear-time reductions.

For any i and j, the class N^iP_j is a subset of $P_{i+j} = N^0P_{i+j} = \bigcup_k \mathrm{TIME}(n^{i+j} \log^k n)$. Whether this containment is proper is an open question, related to but not determined by the $P =? NP$ question. The classes P_{i+j} all have complete sets under quasilinear-time many-one reductions; the equality $N^iP_j = P_{i+j}$ holds if and only if N^iP_j contains a complete set for P_{i+j} [3].

If some problem in N^1P_1 is P_2-hard, then there are significant consequences.

Theorem *If there are $\delta > 0, \varepsilon > 0$ such that $N^\delta P_1$ contains a hard problem for* $\mathrm{DTIME}(n^{1+\varepsilon})$ *under quasilinear-time many-one reductions, then $P = \bigcup_i N^iP_1$ and $P \neq NP$.*

Proof: For simplicity, we prove the theorem for $\delta = \varepsilon = 1$; the generalization is straightforward. Similarly, one can generalize the type of reduction, for instance to bounded-truth-table reductions.

If N^1P_1 contains a hard problem for P_2, then $P_2 \subseteq N^1P_1$. Then $P_3 \subseteq N^1P_2 \subseteq N^2P_1$, and so on; thus any problem in P is in N^iP_1 for some i.

Now suppose, for contradiction, $P = NP$. Then for some k, we have SAT$\in N^kP_1 \subseteq P_{k+1}$. Since SAT is quasilinear-time complete for NP_1 (see [6]), it follows that $NP_1 \subseteq P_{k+1}$. But since $P \subseteq NP_1$ by the previous paragraph, this contradicts the time hierarchy theorem. Hence $P \neq NP$. ∎

[1] Gajentaan and Overmars use a restricted notion of reduction that preserves closure without allowing extra factors of $\log n$. None of the comments in this note is affected by the difference.

Conclusions

We think it unlikely that the listed n^2-hard problems are general quadratic-time hard. Not only would it rather simply and elegantly imply $P \neq NP$, but we question whether SumOfThree is even hard for N^1P_1. The problem seems too structured to capture general computations.

The motion-planning problems considered by Gajentaan and Overmars do not have obvious N^1P_1 or co-N^1P_1 algorithms. Although we have not yet precisely characterized their complexity, one possible approach is provided by the "sharply bounded hierarchy" [1, 2], which allows a constant number of alternations of $\log n$-bit quantifiers. Similar results to the above would follow from such a classification.

References

[1] S. A. Bloch, J. F. Buss, J. Goldsmith, "Sharply Bounded Quantifiers within P," technical report CS94-21, University of Waterloo, 1994.

[2] S.A. Bloch, J. Goldsmith, "Sharply Bounded Alternation within P," technical report 92–04, University of Manitoba, 1992.

[3] J. F. Buss, J. Goldsmith, "Nondeterminism within P," *SIAM J. Computing* 22 (1993) 560–572.

[4] A. Gajentaan, M. H. Overmars, "n^2-Hard Problems in Computational Geometry," technical report RUU-CS-93-15, Utrecht Univ., 1993.

[5] J. O'Rourke, "Computational Geometry Column 22," *SIGACT News* 25,1 (March 1994) 31–33.

[6] C.P. Schnorr, "Satisfiability is Quasilinear Complete in NQL," *J. Assoc. Comput. Mach.* 25 (1978) 136–145.