On the riddle of coding equality function in the garden hose model

Oded Margalit
IBM Research Lab
Haifa 3498825, Israel
Email: odedm@il.ibm.com

***Abstract*— Recently, Harry Buhrman et al introduced a novel communication complexity model, called "garden-hose". This model sprout from a research on using quantum properties to allow for position based cryptography. SAT is one of the fundamental NP-Complete problem — finding a satisfying assignment to a CNF (conjunctive Normal Form) formula, or proving that none exists. We will not get into the details of the quantum physics, neither we are going to explain the internal work of SAT solver. Instead we will start from the mathematical garden hose model; describe the way we used SAT solver as a tool; give some lower and upper bounds on implementing the equality function; and conclude with open questions.**

I. Introduction

Buhrman et al ([1]) presented a novel communication complexity model and used it for position based quantum cryptography. In this paper we show some results on the complexity of the equivalence function in this model. The work that we are describing was done at 2012, but is published here for the first time. This allows for a circular reference — we can cite past work ([2]) which already cites this work (as private communication).

The structure of the paper is as follows: In section II we define the model; in section III we explain how we solved the simple cases using SAT solver; in section IV we change from n bits setting to N possibilities; in section V we tell the story of IBM Ponder–This monthly challenge; in section VI we discuss upper bounds. in section VII we discuss open questions and further research.

II. Definitions

The **Garden Hose model** is a communication complexity model in which two parties *Alice* and *Bob* wants to compute a Boolean function $f(X_A, X_B)$ on two n bits numbers X_A and X_B; where $Alice$ knows X_A and only Bob knows X_B. Alice and Bob are in two different sides of a wall, and they can not talk to each other. The only way they can communicate is by using m pipes that pass through the wall. After Alice is given her secret n bit number X_A, she can connect a water source to one of the pipes and connect disjoint pairs of pipes together. Similarly, after Bob is given hi secret n bit number X_B, he can connect set of pairs of pipes from his side. And then, the water source is activated, and water starts flowing though the pipes. Since each side of each pipe is connected to at most one other pipe, the water must spill over on one of the sides.

The goal of Alice and Bob is to make sure that the water will fall into Alice's side if and only if $f(X_A, X_B) = 1$. In this paper we will only consider the case where f is the equivalence function; i.e. $f(x, y) = 1$ if, and only if, $x = y$.

We will denote $g(m) = n$ the maximal n such that the equivalence function on n bits can be computed using m pipes.

III. On the usage of SAT solver

The way we model the problem of answering whether $g(m) \geq n$; i.e. if there is a solution for the n bit equivalence problem using m pipes is by introducing $m^2 2^n + (m+1)(m+2)4^n$ variables as follows:

- $2^n \binom{m}{2}$ variables for Bob's behavior ($b(i_b, i_m, i_{m'})$).
- $2^n \binom{m}{2}$ variables for Alice's behavior ($a(i_a, i_m, i_{m'})$) which define the pipe–connections made by Alice.
- $2^n m$ variables for Alice's behavior ($a'(i_a, i_m)$) which define the way Alice connects the water source.
- $4^n(m+2)(m+1)$ auxiliary variables ($c(i_a, i_b, m, t)$); the c variables characterize where the water flows, given Alice's input (i_a), Bob's input (i_b), pipe number (-1 is Alice's side, 0 is Bob's side, and $1, \ldots, m$ are the pipes).

Where $0 \leq i_a < 2^n$; $0 \leq i_b < 2^n$; $1 \leq i_m < i_{m'} \leq m$; $-1 \leq m \leq m$; and $0 \leq t \leq m$.

To make notation simpler we double the number of variables in a and b by defining $a(i, j) \equiv a(j, j)$ and $b(i, j) \equiv b(j, j)$, thus making it symmetric.

To characterize the problem of $g(m) \geq n$, we use the following CNF clauses:

1) First we define the connections rules:
   a) Each pipe is connected to at most one other pipe. On Alice's side $\forall\ 0 \leq i_a < 2^n$, $\forall\ 1 \leq i_{m_1}, i_{m_2}, i_{m_3} \leq m$, such that m_1, m_2 and m_3 are different,
   $$\overline{a(i_a, i_{m_1}, i_{m_2})} \vee \overline{a(i_a, i_{m_1}, i_{m_3})}.$$
   b) And on Bob's side: $\forall\ 0 \leq i_a < 2^n$, $\forall\ 1 \leq i_{m_1}, i_{m_2}, i_{m_3} \leq m$, such that m_1, m_2 and m_3 are different,
   $$\overline{b(i_b, i_{m_1}, i_{m_2})} \vee \overline{b(i_b, i_{m_1}, i_{m_3})}.$$
   c) Alice's side has also the water source, so we should make sure that it connects to no more than one

pipe: $\forall\ 0 \le i_a < 2^n$, $\forall\ 1 \le i_{m_1}, i_{m_2} \le m$, such that $m_1 \neq m_2$,

$$\overline{a'(i_a, i_{m_1})} \vee \overline{a'(i_a, i_{m_2})}.$$

d) And Alice's water source can not be connected to a connected pipe: $\forall\ 0 \le i_a < 2^n$, $\forall\ 1 \le i_{m_1}, i_{m_2} \le m$, such that $m_1 \neq m_2$,

$$\overline{a(i_a, i_{m_1}, i_{m_2})} \vee \overline{a'(i_a, i_{m_1})}.$$

2) Now we can define the rules of the flow of the water:

a) Water starts, at time $t = 0$, by flowing in the pipe that is connected to the water source: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 1 \le i_m \le m$,

$$\overline{a'(i_a, i_m)} \vee c(i_a, i_b, i_m, 0).$$

b) Water must be, at any time, in at most one pipe: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 1 \le i_{m_1} < i_{m_2} \le m$, $\forall\ 0 \le t \le m$,

$$\overline{c(i_a, i_b, i_{m_1}, t)} \vee \overline{c(i_a, i_b, i_{m_2}, t)}.$$

c) Water flow according to the pipes connections:

i) For even times t, according to Bob's connections: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 1 \le i_{m_1} < i_{m_2} \le m$, $\forall\ 0 \le t < m$, such that t is even,

$$\overline{c(i_a, i_b, i_{m_1}, t)} \ \vee \overline{b(i_b, i_{m_1}, i_{m_2})} \\ \vee c(i_a, i_b, i_{m_2}, t+1).$$

ii) And for odd times t, according to Alice's connections: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 1 \le i_{m_1} < i_{m_2} \le m$, $\forall\ 0 \le t < m$, such that t is odd,

$$\overline{c(i_a, i_b, i_{m_1}, t)} \ \vee \overline{a(i_a, i_{m_1}, i_{m_2})} \\ \vee c(i_a, i_b, i_{m_2}, t+1).$$

d) If water are flowing on a pipe whose output is disconnected, they fall on the floor:

i) On Alice's side, for odd times t: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 0 \le t < m$, such that t is odd,

$$\overline{c(i_a, i_b, i_{m_1}, t)} \ \vee \bigvee_{i_{m_2} \neq i_{m_1}} a(i_a, i_{m_1}, i_{m_2}) \\ \vee c(i_a, i_b, -1, t+1).$$

ii) And on Bob's side, for even times t:: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 0 \le t < m$, such that t is even,

$$\overline{c(i_a, i_b, i_{m_1}, t)} \ \vee \bigvee_{i_{m_2} \neq i_{m_1}} b(i_a, i_{m_1}, i_{m_2}) \\ \vee c(i_a, i_b, 0, t+1).$$

e) Once the water hit the floor - they must stay there:

i) On Alice's side: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 0 \le t < m$,

$$\overline{c(i_a, i_b, -1, t)} \vee c(i_a, i_b, -1, t+1).$$

ii) And on Bob's side: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, $\forall\ 0 \le t < m$,

$$\overline{c(i_a, i_b, 0, t)} \vee c(i_a, i_b, 0, t+1).$$

3) And the last part, to make sure that water ends up on the right side:

a) If $i_a = i_b$, water falls on Alice's side: $\forall\ 0 \le i_a < 2^n$,

$$c(i_a, i_b, -1, m).$$

b) And if $i_a \neq i_b$, water falls on Bob's side: $\forall\ 0 \le i_a < 2^n$, $\forall\ 0 \le i_b < 2^n$, such that $i_a \neq i_b$,

$$c(i_a, i_b, 0, m).$$

After defining the problem as satisfying a CNF, we used IBM's SAT solver, but we could have used any other SAT solver to solve it. Since the nature of the problem is super-exponential, we can solve only small instances of it. So we found the trivial solution for $n = 1$ (depicted in Figure 1) which shows that $g(3) \ge 1$:

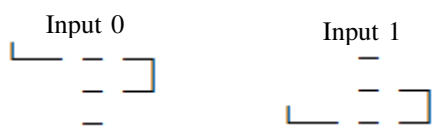

Fig. 1. Solution for $n = 1$ and $m = 3$

Since $n = 1$, the input for Alice and Bob is a single bit (zero or one). In the left part of Figure 1 we have the setting when the number is zero and in the right part of this figure we show the setting when the number is one. In each one of the two parts, the left side dictates what Alice does and the right side instructs Bob what to do. In this simple case, Alice does not use any connectors, only connects the water source. One can easily verify that if the input for Alice and Bob are equal, water flows to Alice's side, as they should; and when we try to combine the left side of one part with the right side of another — water flows into Bob's side, as required.

Trying to find a solution for $n = 1$ and $m = 2$ gives us "UNSAT" answer from the SAT solver, which means that $g(2) = 0$. UNSAT answer for $n = 2$ and $m = 3$ proves that $g(3) = 1$. Similarly, we proved, using the SAT solver, that $g(4) = 2$ and got the solution detailed in Figure 2).

Again, if we look at each one of the four cases, water flows at Alice's side, and if we marry two parts from different cases, water flows at Bob's side. While this solution is correct, it is by no mead the *only* possible solution; and here come the manual part of the story — after examining the solution one can find a more symmetrical solution: Alice connects the water source to her input i_a, and connects $i_a + 1$ to $i_a + 2$ (all the additions are modulo 4). Bob connects his input i_b to $i_b + 3$. Now it

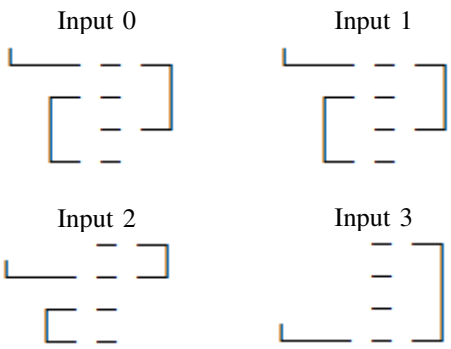

Fig. 2. Solution for $n = 2$ and $m = 4$

easy to verify that water will flow at Alice's side if, and only if, $i_a = i_b$.

The main lesson that one can learn from that is that using the abundant computing power available today one can find working solutions to hard problems; using pen-and-paper to solve these problems one can find elegant solutions. The benefit of the elegant solutions is that they lend themselves better to generalization. The benefit of the computer generated solutions is that they an trigger the manual effort by

1) Showing that a solution exists (sometimes it is easier to find a solution once one knows that it exists); and
2) Helping to jump–start the creative process of solving the problem.

IV. Changing from n to N

Even without the super–exponential nature of the problem, one can not hope to get an asymptotic solution from a SAT solver; but optimally solving small problems can be used to cascade them into arbitrary larger ones and get asymptotic results. After using the SAT solver to show that $g(5) = 3$, an important insight was discovered. What we actually doing is finding $N = 2^n$ different pairs of pipes connections (one for Alice and one for Bob) such that all the pairs fit in a way that guarantees that water will flow on Alice side; and combining two parts from different pairs will make water come out on Bob's side. The insight is that we can generalize the problem into cases where N is not necessary a power of 2. We define $G(m) = N$ to be the maximal number of input possibilities N that can be solved using m pipes. Clearly

$$G(m) \geq 2^{g(m)}. \tag{1}$$

; But, there are cases when the inequality is acute: we can not only solve for $n = 2$ ($N = 4$) and $m = 4$, but also solve for $N = 6$ and $m = 4$; and we can prove tha tactually $G(4) = 6 > 2^{g(4)}$. The solution is let Bob connect one pair of pipes according to its input while Alice uses one of these pipes as water source, connects all the rest in pairs, and leaves the other one free. It is easy to verify that it solves the problem for every even m and $N = \binom{m}{2}$. Note that m should be even, since this will allow us to "close" all the pipes except the source and target and make sure that every other value for Bob's input water will, indeed, flow to Bob's side.

Any solution for G can be transformed into a solution for g; and if m is even then we get similar asymptotic behavior: Suppose that we have a construction for N possible inputs and m pipes (thus proving that $G(m) \geq N$) then we first add one more pipe (if needed) to make m even. Then we add more connections to Alice's construction to make sure that in all cases, all her pipes, but one, are connected. We can do that since adding such connections does not hurt when the inputs are different (such change can only make water flow from Alice's side to Bob's side) and in the case when the inputs are the same — we know which pipe is used last, just before the water spill into Alice's side, and can arbitrarily connect all the rest in pairs (remember that the number of pipes, m, is even).

Leaving only one open pipe at Alice's side enables us to connect the output of one construction to serve as an input (the water source) of another construction and cascase k copies of m pipes into a bigger super–construction with km pipes and N^k possible inputes. So we get

$$g(km_0) \geq \lfloor \log_2(G(km_0)) \rfloor \geq \log_2(G(m_0)^k) - 1.$$

On the other hand, from equation (1) we get the other side and thus:

$$\lim_{m\to\infty} \frac{m}{G(m)} \leq \lim_{m\to\infty} \frac{m}{g(m)} \leq \frac{m_0}{\log_2(G(m_0))}.$$

So we proved that working with the G function is asymptotically the same as the g function; and we can more on to telling the story of Ponder–This challenge.

V. IBM Ponder This challenge

IBM Research publishes, for more than 15 years, a monthly mathematical challenge ([4]). On April 2012 the monthly challenge was to show that $g(8) \geq 5$. When the question was asked, we already knew how to cascade the solution for $N = 6$ and $m = 4$ twice and get show that $G(8) \geq 36$; but we asked for $N = 2^5$ only for two somewhat contradicting reasons:

1) To make the problems somewhat easier; and
2) To make a red–herring — throwing people into thinking of $N = 32$ as encoding a 5–bit number.

In the first 10 days of April, we were still trying to prove the optimality of $N = 36$, (i.e. prove that $G(8) = 36$) until Don Dodson sent us a solution with $N = 39$. Instead of drawing the pipes like we did in Figures 1 and 2, we show Don's answer (Figure 3) in the format we asked in Ponder–This challenge. The format is N lines, each line has two parts: the first is Alice's setting and the second is Bob's setting for the same input. Bob's connection is a set of pairs of pipes (denoted by the first 8 letters of the English alphabet); while Alice's connections contain one more letter, T, which denotes the water source (tap).

Once we saw that $N = 36$ is not an upper bound, we stopped trying to prove it as such and went, instead, to try to get better solutions. There are (up to symmetries) $2,520$ compatible configurations for Alice and Bob which consists of Alice connecting a tap and three pairs and Bob connecting

AB,CE,Tg,FH:BD,AG	TC,DE,BF,AH:AC,GH
AB,TD,CE,FG:AD,BH	TC,DE,BG,AH:AF,CH
BC,AF,DG,TH:CE,BH	BC,AE,TF,GH:CD,BF
TA,BD,CF,GH:AC,EF	AB,CE,TF,DH:AF,BG
TC,AD,EF,GH:AB,CD	BC,DE,TG,AH:AF,GH
TA,BD,CG,FH:CE,AG	BE,CF,TG,AH:AD,GH
BC,TD,AF,GH:BD,CE	CE,TF,AG,BH:BD,FH
TD,AE,BF,CH:CD,GH	AD,CF,BG,TH:CE,FH
CE,AF,BG,TH:BD,GH	AC,DE,TG,BH:BF,GH
AC,TD,FG,BH:AD,CE	TA,BC,DF,GH:AC,BE
TA,CD,BF,EG:AB,FH	AD,BF,TG,EH:CE,GH
AB,TC,DE,GH:AC,BF	BC,TD,AE,FG:CD,BH
AC,BE,TG,DH:EF,BG	TA,BF,DG,CH:BE,AF
AD,BE,TF,GH:CD,AF	TD,BE,CG,AH:BD,EF
TA,BC,DF,EG:AB,CH	BC,AE,TF,DH:EF,AG
CD,BF,AG,TH:EF,BH	AB,DE,TG,FH:AC,BG
BC,DF,TG,AH:CE,BG	AB,TD,CE,GH:BD,AF
TA,CD,EF,BG:AB,GH	TD,AE,BG,CH:CD,FH
DE,TF,AG,BH:BF,CH	

Fig. 3. $G(8) \geq 39$

TA,CE,BD,FH:AC,EG	TA,CE,BD,FG:AC,EH
TA,CF,BD,EH:AC,FG	TA,CF,BD,EG:AC,FH
TA,DE,BC,FH:AD,EG	TA,DE,BC,FG:AD,EH
TA,DF,BC,EH:AD,FG	TA,DF,BC,EG:AD,FH
TB,CE,AD,FH:BC,EG	TB,CE,AD,FG:BC,EH
TB,CF,AD,EH:BC,FG	TB,CF,AD,EG:BC,FH
TB,DE,AC,FH:BD,EG	TB,DE,AC,FG:BD,EH
TB,DF,AC,EH:BD,FG	TB,DF,AC,EG:BD,FH
TC,AB,GH,FD:CE	TC,AB,GH,ED:CF
TD,AB,GH,FC:DE	TD,AB,GH,EC:DF
TC,AB,EF,DH:CG	TC,AB,EF,DG:CH
TD,AB,EF,CH:DG	TD,AB,EF,CG:DH
TA,CD,GH,BF:AE	TA,CD,GH,BE:AF
TB,CD,GH,AE:BF	TB,CD,GH,AF:BE
TA,CD,EF,GH:AB	TC,AB,EF,GH:CD
TE,AB,CD,GH:EF	TG,AB,CD,EF:GH

Fig. 4. Solution for $N = 32$ and $m = 8$

two pairs. We denote such configuration as *regular*. Each such compatible configuration guarantees that when both Alice and Bob are using the same configuration — water will flow on Alice's side. Let's look at all the pairs of configurations. Sonme of these pairs are ok — meaning that choosing parts from two different configuration leads to water falling in Bob's side; while other are not (even though we choose Alice's and Bob's connections from two different pairs, still water will flow on the wrong side — Alice's side.

So our problem is to find a maximal clique in a graph of $2,520$ vertices. This is not a trivial task; and furthermore, it is not clear that using this subset of $2,520$ possible configurations must give the optimal solution (see note about it later, regarding Figure 5); but still solving it by simple greedy backtracking algorithm we found a solution for $N = 41$.

This solution was an important milestone, since it yields an asymptotic solution of $m/g(m) < 1.5$.

Few days later, another Ponder–This solver, Thomas Fleming, send us an interesting solution (see Figure 4). While his solution only solved our problem for $N = 32$, it had a very nice structure.

TD,AB,CG,EF:AD,BH	TB,CD,EF,GH:AC,BD
TA,BF,CD,EG:AF,BH	TE,AD,BF,CH:BG,EF
TE,AB,CH,DF:AG,BE	TA,CF,DG,EH:AG,BD
TF,AE,CG,DH:BG,CF	TF,AB,CG,EH:AF,BD
TF,AH,BC,EG:BD,CF	TG,AB,CH,DE:CF,GH
TE,BH,CF,DG:AF,CE	TF,AC,DG,EH:BH,EF
TG,AD,BH,CE:AF,DG	TA,BH,CG,DF:AG,CE
TD,AG,BC,FH:DE	TD,AC,BG,EF:BH,DG
TG,AB,DF,EH:AC,BG	TH,AD,BG,EF:CF,EH
TE,AH,BD,CF:BE,DG	TH,BG,CF,DE:AD,EH
TB,AG,CE,DH:BE,CF	TB,AF,DE,GH:BC
TF,AH,BG,CE:CD,EF	TC,AF,BG,DE:CD,EH
TH,AF,BE,DG:BH,CE	TA,BE,CF,DG:AD,GH
TA,BC,DF,GH:AC,BE	TB,CF,DE,GH:AD,BE
TH,AC,BF,EG:DG,EH	TD,AC,BH,EG:AF,CD
TH,BE,CG,DF:AC,GH	TG,AE,BH,CD:AG,EF
TC,AF,BH,DE:CE,DG	TA,BG,CE,DF:AC,EH
TF,AB,CE,DH:DG,FH	TC,AF,BE,DH:BG,CE
TC,AD,BH,EF:AG,CD	TF,AE,BG,CH:CD,FH
TA,BF,CD,GH:AD,CE	TH,BE,CD,FG:AF,GH
TD,AF,BE,CG:BD,EH	TD,AH,BE,CG:BD,EF
TC,AD,BF,EG:BH,CF	TG,AC,BD,EH:EF,GH
TA,BG,CD,EF:AF,EH	TA,BC,DF,EG:AC,BH
TG,AF,BD,EH:BG,CD	TC,AF,BD,EG:BH,CD
TG,AD,BH,CF:BE,GH	

Fig. 5. $G(8) \geq 49$
Solution for $N = 49$ and $m = 8$

When kick-starting the greedy backtracking search mentioned above with the first half of the solution in Figure 4 — we quickly get a solution with $N = 45$, and several cpu hours later we got to $N = 46$. But the best solution we got is $N = 49$, as shown in Figure 5.

Note that while $N = 49$ is the best solution we know of for $m = 8$, and it does not use only configurations from the set of $2,520$ regular configurations; still, there may be another solution which is both $N \geq 49$ and, maybe, consists of only regular configurations.

VI. Upper bounds

When the inputs of Alice and Bob are the same, they both chose compatible parts of a pipe configuration. For each input i from the set of possible N values, denote the set of pipes that the water flows through as *wet set* W_i.

First note that we can always disconnect any dry (non–wet) pipe from Bob's side. The reason for that is that it does not change the water flow in case of equality (by definition of a wet pipe — water does not flow through a dry pipe in case of equality) and te other (non-equal) cases, the fact that we disconnect any pipe on Bob's side can only help to make sure that water eventually falls on Bob's side. So, without loss of generality, we may assume that all the dry pipes are not connected on Bob's side.

We prove that there are no two different inputs i_1 and i_2 such that $W_{i_1} \subseteq W_{i_2}$: Suppose, by contradiction, that there is such case, then let's examine what happens when we give Alice i_1 as her input and give Bob i_2 as his. Since giving i_2 to both Alice and Bob make the water flow on Alice's side, Bob's configuration for input i_2 must connect all the pipes in the wet set W_{i_2}; otherwise water would have spilled to

Bob's side. Since giving i_1 to both Alice and Bob makes the water come out on Alice's side, there is no connection from any pipe in W_{i_1} to a pipe outside W_{i_1}. From our assumption, $W_{i_1} \subseteq W_{i_2}$, we get that water can not come out on Bob's side, and we get the contradiction we need.

Therefore, we can not have N be more that the size of the largest anti-chain in the power set of size m. It is easy to see that this size is $\binom{m}{\lfloor m/2 \rfloor}$.

Applying this upper bound on the $m = 8$ case we get an upper bound of $G(8) \leq 70$ which translates, asymptotically, to

$$n/g(n) \geq 1.305.$$

This upper bound also suggests (though, of course, does not prove) that using wet sets that are not of size $m/2$ is "a waste"; and therefore hints that the best solution that we know of (49, see Figure 5) may be improved.

As for the general case, the same upper bound translates to

$$\frac{n}{g(n)} > \frac{2m}{\log_2(\binom{2m}{m})} \approx \frac{2m}{\log_2(2^{2m}/\sqrt{2\pi m})} \approx \frac{2m}{2m - \log_2(m)/2} = 1 + \Omega(\frac{\log(m)}{m}).$$

Bouding it (slightly) away from 1.

VII. Open problems and future research

The natural question is, of course, the asymptotic limit of $n/g(n)$. We got to 1.425 and a very recent work ([3]) reached 1.359.

Since we asked for the exact value of $G(8)$ and made many good people ponder on this specific question — it would be nice to close this finite problem and find the optimal solution.

Acknowledgment

The author would like to thank Harry Buhrman for introducing him to this lovely problem; to Arie Matsliah and Alexander Ivrii from IBM for fruitful discussions; and to Ponder–This solvers for their ideas.

References

[1] Harry Buhrman and Serge Fehr and Christian Schaffner and Florian Speelman, "The Garden-Hose game: A new model of computation, and application to Position-Based quantum cryptography," *QCrypt(2011)*

[2] Harry Buhrman and Serge Fehr and Christian Schaffner and Florian Speelman, "The garden-hose model," *ITCS '13 Proceedings of the 4th conference on Innovations in Theoretical Computer Science* Pages 145–158.

[3] Well Y. Chiu and Mario Szegedy and Chengu Wang and Yixin Xu, "The Garden Hose Complexity for the Equality Function," *arXiv preprint arXiv:1312.7222 (2013).*

[4] Oded Margalit, "IBM Ponder This," *Monthly challenge Internet site www.research.ibm.com/ponder (1998–2014)*