Electronic Colloquium on Computational Complexity, Report No. 3 (2009)

Comments on ECCC Report TR06-133: The Resolution Width Problem is EXPTIME-Complete

Alexander Hertel *
2525 Cranmore Road
Victoria, B.C., V8R 1Z8
Canada
ahertel@cs.toronto.edu

Alasdair Urquhart †
Department of Computer Science
University of Toronto
Toronto, ON CANADA
urquhart@cs.toronto.edu

January 6, 2009

Abstract

The main argument of the report TR06-133 is in error. The paper claims to prove the result of the title by reduction from the $(\exists, k)$-pebble game, shown to be $\mathcal{EXPTIME}$-complete by Kolaitis and Panttaja. This note shows that the principal lemma is incorrect by providing a simple counter-example.

1 The counter-example

The main theorem of the paper depends on the following claim, stated as Lemma 5.3:

"If $\mathcal{A}$ and $\mathcal{B}$ are coloured graphs, and $k \geq 3$, then the Spoiler has a winning strategy for the $(\exists, k)$-pebble game on $\mathcal{A}$ and $\mathcal{B}$ if and only if the Prover has a winning strategy for the $k+2$-width game on $\Sigma(\mathcal{A}, \mathcal{B})$."

The counter-example that follows shows that this claim is incorrect. We provide two graphs $\mathcal{A}$ and $\mathcal{B}$, for which the Prover has a winning strategy for the 5-width game on $\Sigma(\mathcal{A}, \mathcal{B})$, but on the other hand, the Spoiler does not have a winning strategy for the $(\exists, 3)$-pebble game on $\mathcal{A}$ and $\mathcal{B}$.

The graphs $\mathcal{A}$ and $\mathcal{B}$ are easy to describe. The graph $\mathcal{A}$ is the complete graph K_5, and the graph $\mathcal{B}$ is the complete graph K_4. The colours of the nodes in the graphs play no role (we can think of all the nodes as being coloured the same colour), so we shall ignore them in the remainder of this note.

*Research supported by the Natural Sciences and Engineering Research Council of Canada.
†Research supported by NSERC grant -9700239.

ISSN 1433-8092

Clearly, the Spoiler wins the $(\exists, 5)$-pebble game on K_5 and K_4, since when pebbles have been placed on all the nodes of K_5, the Duplicator is forced to place two different pebbles on the same node, thus violating the partial homomorphism property. On the other hand, the Duplicator wins the $(\exists, k)$-pebble game on K_5 and K_4, for $k \leq 4$, since as long as the Duplicator is careful never to place two pebbles on the same node, the partial homomorphism property can never be violated.

We now have to demonstrate a winning strategy for the Prover in the 5-width game on $\Sigma(K_5, K_4)$. The set of clauses $\Sigma(K_5, K_4)$ contains 20 variables P^i_j, for $1 \leq i \leq 5$ and $1 \leq j \leq 4$, together with 15 variables Q^i_j, for $1 \leq i \leq 5$ and $1 \leq j \leq 3$.

The clauses constituting $\Sigma(K_5, K_4)$ are as follows:

1. Q^i_1, for $1 \leq i \leq 5$.
2. $Q^i_j \leftrightarrow (P^i_j \vee Q^i_{j+1})$, for $1 \leq j < 3$, and $Q^i_3 \leftrightarrow (P^i_3 \vee P^i_4)$.
3. $\neg P^i_j \vee \neg P^l_j$, where $i < l$.
4. $\neg P^i_j \vee \neg P^i_k$, where $1 \leq j < k \leq 4$,

They can be interpreted as asserting that there is a bijective mapping from a set of size 5 into a set of size 4, and so the fact that $\Sigma(K_5, K_4)$ is contradictory formalizes the pigeonhole principle. In fact, $\Sigma(K_5, K_4)$ is essentially the same as the formalization of the pigeonhole principle $EPHP^5_4$ given by Atserias and Dalmau [1].

In explaining the Prover's winning strategy in the 5-width game, it helps to think of the width game as a pebbling game. The game is played on a board, consisting of a set of locations, where either a black or a white pebble can be placed (we think of a white pebble as representing "true," and a black pebble as representing "false"). A round in the game consists of the Prover first (optionally) removing some pebbles, and then querying a location. The Adversary responds by placing a pebble, either black or white, on the location in question. Certain configurations of pebbles are declared forbidden. If a play of the game ends with a forbidden configuration on the board, but no more than k pebbles have been on the board at any time, then the Prover is said to win the k-width game.

The board on which the k-width game for $\Sigma(K_5, K_4)$ is played can be visualized as a seven by four matrix, with the P^i_j variables on the righthand side, and the extension variables Q^i_j on the left. The forbidden configurations are defined by the clauses. For example, the clauses of type 3 correspond to the fact that in the board game, a righthand column containing two white pebbles is forbidden.

Q^1_1	Q^1_2	Q^1_3		P^1_1	P^1_2	P^1_3	P^1_4
Q^2_1	Q^2_2	Q^2_3		P^2_1	P^2_2	P^2_3	P^2_4
Q^3_1	Q^3_2	Q^3_3		P^3_1	P^3_2	P^3_3	P^3_4
Q^4_1	Q^4_2	Q^4_3		P^4_1	P^4_2	P^4_3	P^4_4
Q^5_1	Q^5_2	Q^5_3		P^5_1	P^5_2	P^5_3	P^5_4

The Prover's strategy consists of two parts. First, the Prover, playing solely on the righthand side of the board, forces a row containing four black pebbles. Second, using the lefthand side of the board, the Prover forces a black pebble on a location containing a variable of the form Q^i_1, or a white pebble on Q^5_1 and black pebbles on P^5_1 and Q^5_2 – under the assumption that the Adversary puts up as long a resistance as possible.

For the first part of the strategy, the Prover queries all the variables P^1_j. If the Adversary responds with black pebbles to all the queries, then the Adversary can proceed to the second part. However, let us assume that the Adversary places a white pebble on one of the locations, say P^1_2. The Prover then removes all the pebbles except the white pebble from the first row, and then queries the locations P^2_1, P^2_3, P^2_4. If the Adversary responds with black pebbles to all of these queries, then the Prover queries P^2_2. Since there is a white pebble on P^1_2, the Adversary must respond with a black pebble, and again we have forced a black row. So, let us assume that the Adversary responded with a white pebble on P^2_3. The Adversary now removes all the pebbles except the white pebbles on P^1_2 and P^2_3, and queries P^3_1 and P^3_4. Again, if the Adversary answers with a black pebble to both queries, the Prover can force black pebbles on P^3_2 and P^3_3, removing the white pebbles when they are no longer needed. So, again, we can assume that the Adversary answers with a white pebble to one of the queries. Proceeding in the same manner on the fourth row, the Prover either forces a black row, or four white pebbles, none of which are in the same row or column (that is to say, they represent a partial homomorphism from K_5 to K_4). In the second case, by querying successively the P^5_j variables, and then removing the corresponding white pebbles in the appropriate column, the Prover can force either a black row, or an immediate termination of the game (if the Adversary ever responds with a white pebble).

So, let us assume that the Prover has forced black pebbles on all of the locations P^5_1, P^5_2, P^5_3, P^5_4 (by symmetry, the strategy is the same for all the rows). The Prover then queries Q^5_3. The Adversary is forced to answer with a black pebble, or the game ends immediately. The Prover then removes the black pebbles on P^5_3 and P^5_4, and queries Q^5_2. Again, the Adversary must answer with a black pebble, or the game ends with a win for the Prover. Finally, the Prover queries Q^5_1. No matter how the Adversary answers, the game ends with a win for the Prover. Since this strategy never uses more than five pebbles, the Prover wins the 5-width game on $\Sigma(K_5, K_4)$. More generally, the Prover wins the $n+1$-width game on $\Sigma(K_{n+1}, K_n)$, showing that the minimum width of a resolution refutation of $\Sigma(K_{n+1}, K_n)$ is exactly n.

2 The nature of the error

The error in the paper is slightly subtle, so it is of some interest to see just where it lies. It is possible to pinpoint the mistake by following through the Prover's strategy from the previous section, while employing the Adversary's strategy from the paper.

The error occurs in the second part of Lemma 5.3. The argument that purports to show that the Adversary's strategy succeeds uses implicitly the following principle:

> If C is an initial clause containing only variables of the form P^i_j and Q^i_j, f is a partial homomorphism, and a_i is in the domain of f, then C is true under the assignment $\beta[f]$.

The principle is easily seen to be true, since the initial clauses are all intended to describe a partial homomorphism.

The mistake arises from an incorrect assumption that can be expressed as follows:

> If f is the partial homomorphism maintained by the adversary, a_i is in $\mathrm{Dom}(f)$, and a variable of the form P^i_j or Q^i_j is assigned a value by the current assignment, this value is in agreement with the assignment $\beta[f]$.

Somewhat surprisingly, this assumption is false. Let us follow through the Adversary's strategy to see where it fails.

We take up the game at the point where the Prover has forced a partial homomorphism of size 4; let's say, for example, that the variables P^1_1, P^2_2, P^3_3 and P^4_4 are all assigned true under the current assignment. Now the Prover queries P^5_4, which the Adversary is compelled to answer with "false." Next, the Prover removes the white pebble on P^4_4 (removes this assignment from the current assignment) and queries P^5_3. At this point, the Adversary, following the given strategy, must extend the new partial homomorphism to include a_5 in the domain. However, the only consistent way to do this is to set $f(a_5) = a_4$. This setting contradicts the assumption above, since the current assignment sets P^5_4 to false, but $\beta[f]$ sets it to true. In fact, two more queries result in the initial clause $Q^5_3 \rightarrow (P^5_3 \vee P^5_4)$ being set to false (assuming that the Adversary continues following the given strategy).

The precise point where the error occurs is in the sentence: "In the first two cases, this is clearly true, by the definition of an extendible k-family." So, this confirms once again the mathematician's rule of thumb – to find errors in a paper, look for points where words like "obviously," "clearly," "trivially" and the like occur.

3 Status of the problem

So where does this leave the problem? The original claim was that a generic reduction would work to reduce instances of the existential k-pebble game to the width problem for resolution – the reduction would be "generic" in the sense that it would not depend on the details of the particular instance.

However, the fact that the Prover wins with only $n + 1$ pebbles in the $\Sigma(K_{n+1}, K_n)$ width game seems to depend on the precise details of the set of clauses. It does not seem easy to generalize the strategy. Consequently, it appears at the moment that the complexity of the width problem for resolution must be considered open once again.

References

[1] Albert Atserias and Victor Dalmau. A combinatorial characterization of resolution width. In *18th IEEE Conference on Computational Complexity (CCC)*, pages 239–247. IEEE Computer Society Press, 2003.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de/