On (an error in) the Simultaneous Hall Condition

Kevin Fockaert, Laurent Janssens, Bart Demoen

Report CW 632, January 2013

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

On (an error in) the Simultaneous Hall Condition

Kevin Fockaert, Laurent Janssens, Bart Demoen

Report CW 632, January 2013

Department of Computer Science, KU Leuven

Abstract

CoRR abs/1004.2626 states a necessary and sufficient condition, SIM-HC, for the existence of a perfect matching in an overlapping bipartite graph, a so-called simultaneous (perfect) matching, a generalization of Hall's marriage theorem. A suprisingly small counterexample shows that the condition is not sufficient. A short proof of the necessity of the condition is given here, and the culprit in the original proof is identified.

On (an error in) the Simultaneous Hall Condition*

Kevin Fockaert
KU Leuven

Laurent Janssens
KU Leuven

Bart Demoen
KU Leuven

Abstract

[1] states a necessary and sufficient condition, SIM-HC, for the existence of a perfect matching in an overlapping bipartite graph, a so-called simultaneous (perfect) matching, a generalization of Hall's marriage theorem. A suprisingly small counterexample shows that the condition is not sufficient. A short proof of the necessity of the condition is given here, and the culprit in the original proof is identified.

1 Introduction

Problems with *overlapping* all_different constraints occur all the time[1], and they have been studied extensively. A nice survey about the all_different constraint can be found in [5]. A necessary and sufficient condition (SIM-HC) for the existence of a solution for a set of overlapping all_different constraints would generalize the famous *Marriage Theorem* by P. Hall [3]. [1] proves such a SIM-HC for the existence of a perfect matching in an overlapping bipartite graph, a so-called simultaneous (perfect) matching, i.e. a solution for two overlapping all_different constraints. We refer to that paper often, in particular its **Theorem 2** and use freely its notation: the interested reader should have [1] handy - or the version at CoRR [2].

We report here on a counterexample to the main theorem in [1] **Theorem 2**. Only while writing down this report, we became aware of the PhD thesis [4] of one of the co-authors of [1] in which the author shows a different (larger) counterexample to that particular theorem: so the (negative) result was previously known, but we arrived at it independently.

We think it is important to expose the (negative) result better. It is indeed very tempting to trust [1]: it was written by respectable researchers, peer-reviewed, and impenetrable. Exposure of the inaccuracy of *Theorem 2* could encourage other researchers to search for and discover the *correct* version of

*This research was carried out in the context of a bachelor's thesis by the first two authors, in partial fulfilment of the requirements for the degree of Bachelor in Informatics, and under the supervision of the third author

[1] E.g. Sudoku, Latin Square, and most scheduling problems.

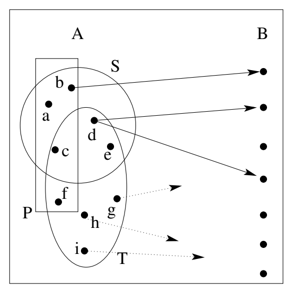
that theorem, and not try to prove new results starting from a flawed theorem. Moreover, we feel like exposing the (negative) result because two years after (at least a subset of) the authors of [1] themselves became aware of the flaw in their **Theorem 2**, they have not yet amended their CoRR version with a caveat about the theorem.

We also want to report on how we found the counterexample: exploratory programming played an important role. Exploratory programming is often defined as a technique within software engineering, but it is also a very valid and useful tool in the discovery and validation of (potential) theorems.

This report is structured as follows: Section 2 explains the theorem for which a counterexample was found. Section 3 shows the counterexample. Section 4 tries to pinpoint where the proof of **Theorem 2** goes wrong. Section 5 explains how the counterexample was found.

2 Theorem 2 in [1]

In the figure at the right, we have drawn part of the bipartite graph $(A \cup B, E)$ about which **Theorem 2** speaks: we have the arcs as an arrow in order to make the mapping N more apparent. The set of nodes A is the union of two non-empty overlapping sets S and T. [1] introduces the following notation:



- for $P \subseteq A$: $P^S = P \cap (S \setminus T)$ and $P^T = P \cap (T \setminus S)$
- for $P \subseteq A$: $N(P) = \{v | \exists u : (u, v) \in P\}$
- $|P|$ denotes the number of elements in a set P

In the figure, $P^S = \{a, b\}$ and $P^T = \{f, g, h, i\}$
In what follows, *SIM-BM* means *Simultaneous Bipartite Matching*. We now state

(Erroneous) Theorem 2 from [1]: *(Simultaneous Hall Condition (SIM-HC))*. Let $G =< A \cup B, E >$ and sets S, T be an overlapping bipartite graph. There exists a *SIM-BM*, *iff* $|N(P)| + |N(P^S) \cap N(P^T)| \geq |P| \; \forall P \subseteq A$. ■

In Section 3 we give a counterexample to the *if* part of the above non-theorem. A quick proof of the *only if* part is given below - just to be on the safe side, it is formulated explicitly:

Theorem 2.1. *If G has a* SIM-BM*, then*

$$\forall P \subseteq A, |P| \leq |N(P)| + |N(P^S) \cap N(P^T)|.$$ [2]

<u>Proof</u> Let there be a SIM-BM M for G: M is a function that maps elements of A to elements of B. Then define $Same_T$ as follows:

$Same_T = \{t \in A^T | \exists s \in A^S, M(t) = M(s)\}$

Then M is a perfect match from $(A \setminus Same_T)$ to $N(A)$. So, by Hall's theorem we have $|A \setminus Same_T| \leq |N(A)$.

Since $Same_T \subset A$, we have $|A \setminus Same_T| = |A| - |Same_T|$, and since also $|M(Same_T)| = |Same_T|$, we derive $|A| \leq |N(A)| + |M(Same_T)|$

Finally, since $M(Same_T) \subseteq (N(A^S) \cap N(A^T))$, we derive:

$$|A| \leq |N(A)| + |N(A^S) \cap N(A^T)|$$

Since A has a SIM-BM, every subset P of A has one as well, and satisfies the same inequality. In summary, we have

$$\forall P \subseteq A, |P| \leq |N(P)| + |N(P^S) \cap N(P^T)|$$ ■

3 The counterexample

The following is a CLP-like specification of the counterexample:

```
dom(U,[1,2]), dom(V,[1]), dom(X,[1,2,3]), dom(Y,[1,3]),
alldiff([U,V,X]), alldiff([V,X,Y])
```

In terms of [1], {U,V,X,Y} equals the set A, {1,2,3} equals the set B, S = {U,V,X} and T = {V,X,Y}. Figure 1 shows it at the left. One can check that it satisfies SIM-HC.

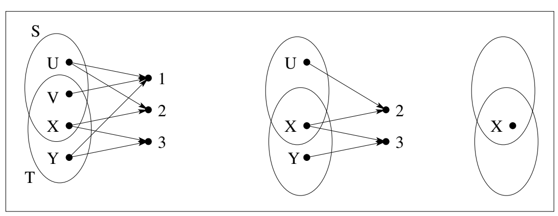


Figure 1: At the left the example; the other two show propagation at work

V has only one neighbour, so one can remove edge (V,1). That results in the graph in the middle. Then U and Y both have only one neighbour, and the result is at the right: there remain no possibilities for X.

To see that the initial graph satisfies SIM-HC, it sufficies to notice that

- for $P \subset A$ with $|P| = 1$, $N(P)$ contains at least one element

[2] We formulate it this way because we became acquainted with Hall's theorem by a formula containing $|P|$ at the left side. The formulation in [3] uses natural language expressing the same.

- for $P \subset A$ with $|P| = 2$, $N(P)$ contains at least two elements
- for $P \subset A$ with $|P| = 3$, $N(P)$ contains always three elements
- for $P = A$ we have $|P| = 4$, $|N(P)| = 3$ and $|N(P^S) \cap N(P^T)| = |\{1\}| = 1$

In all cases $|P| \leq |N(P)| + |N(P^S) \cap N(P^T)|$.

4 Where does the proof in [1] go wrong ?

The proof of the *if-part* of the theorem clearly does not work.

Take the example in 1: **Case 1** of [1] does not apply to it. Consider in **Case 2** $P = \{V\}$, then the resulting graph Q is exactly the graph in the middle of Figure 1. It clearly does not satisfy SIM-HC: it suffices to take (in this new graph) $P = \{U, X, Y\}$.

This means that the analysis in **Case 2** must be incorrect.

One problem with checking the proof in [1] is that it mixes the *if* and *only-if* part of the statement of the theorem. That results in confusion. However, consider the counterexample, and take in **Case 2** of the proof $P = \{V\}$: clearly $|P| = |N(P)| + |N(P^S) \cap N(P^T)| = 1$ as $P^S = P^T = \emptyset$. The graph Q subsequently constructed in the proof consists of the edges $\{(U, 2), (X, 2), (X, 3), (Y, 3)\}$, i.e. the graph in the middle of Figure 1. Then the proof says (we cite literally)

We claim that the SIM-HC holds also for Q

It is clear that SIM-HC does not hold for Q, so the error is somewhere in the proof of the claim.

In the proof, it says at some point:

Similarly $N(P^S \cup P^{S'}) = N(P^S) \cup N_Q(P^{S'})$

However, $P^S = \emptyset$, and $P^{S'} = \{U\}$ so the statement is equivalent to $N(\{U\}) = N_Q(\{U\})$; but $N(\{U\}) = \{1, 2\}$, while $N_Q(\{U\}) = \{1\}$, so, apparently, this line is in error.

5 How the counterexample was derivered

We tried to verify by *explorative programming* **Theorem 2** in [1]. We were not expecting to find a counterexample, even though some details of the proof escaped us: we trusted it to be true. We used the following approach:

For a given number of variables V, and domain size D, we first generate random set S and T of variables so that $S \cap T \neq \emptyset$ and $|S \cup T| = V$. We then associate each variable with domain $1..D$. That constitutes the initial constraint problem CSP_u and we continue only if this has a solution.

```
repeat
      copy CSP_u to CSP_s;
      randomly remove one value from a domain of a variable in CSP_u;
until not(has_a_solution(CSP_u))
```

At the end of this program, we know that CSP_s has a solution while CSP_u has not, and that CSP_s and CSP_u differ only by one domain element for one variable. So CSP_s and CSP_u are at the edge of having a SIM-BM or not, and they are good candidates for checking the necessary and sufficient SIM-HC of [1]. About 5% of the generated CSP_u satisfied the SIM-HC: the counterexample shown above is just one of them. The program was written in [6].

6 Future Work

We intend to work on establishing a sufficient condition for the existence of a SIM-BM, and to generalize the condition to more than two overlapping all_different constraints.

Acknowledgments

We thank M. Garcia de la Banda for verifying our counterexample, and for pointing at the induction argument as a possible culprit in the proof of **Theorem 2**.

References

[1] C. Bessiere, G. Katsirelos, N. Narodytska, C.-G. Quimper, and T. Walsh. Propagating conjunctions of alldifferent constraints. In M. Fox and D. Poole, editors, *AAAI*. AAAI Press, 2010.

[2] C. Bessiere, G. Katsirelos, N. Narodytska, C.-G. Quimper, and T. Walsh. Propagating conjunctions of alldifferent constraints. *CoRR*, abs/1004.2626, 2010.

[3] P. Hall. On representatives of subsets. *J. London Math. Soc.*, 10:26–30, 1935.

[4] N. Narodytska. *Reformulation of Global Constraints*. PhD thesis, The University of New South Wales, Sydney, Australia, 2011.

[5] W. J. van Hoeve. The alldifferent constraint: A survey, 2001. URL `http://www.citebase.org/cgi-bin/citations?id=oai:arXiv.org:cs/0105015`.

[6] N.-F. Zhou. The language features and architecture of B-Prolog. *TPLP*, 12 (1-2):189–218, 2012. ISSN 1471-0684.