Sudoku Players' Forums

Strong inferences induced by the UR

Goto page Previous 1, 2, 3, 4, 5, 6 Next

newtopic postreply Sudoku Players' Forums Forum Index -> Advanced solving techniques

View previous topic :: View next topic

Author	Message
David P Bird Joined: 16 Sep 2008 Posts: 138 Location: Middle England	Posted: Thu Jul 23, 2009 9:41 am Post subject: quote

MJ Thank you very much for your grid which has kept me occupied for quite a while! I have to admit I deliberately didn't read your approach so as not to spoil my own problem solving pleasure. This is what I got:

The cells marked with a + sign below are an Almost Unavoidable Set (BUG) which needs to be disrupted by at least one of (4)r2c5 & (6)r6c6 being true.
This assigns (8)r2c6 through the chain: (4)r2c5 =[BUG]= (6)r6c6 - (6=8)r2c6 => r2c5 <> 8

Code:

```
|----------------|----------------|----------------|
| 569  8    45   | 46   1    59   | 2    3    7    |
| 7    46   2    | 3    68+4 68+  | 5    1    9    |
| 59   1    3    | 57   79   2    | 6    8    4    |
|----------------|----------------|----------------|
| 1    2    7    | 45   34   35   | 9    6    8    |
| 48+  5    6    | 9    78+  1    | 47+  2    3    |
| 48+  3    9    | 67   2    78+6 | 47+  5    1    |
|----------------|----------------|----------------|
| 2    9    1    | 8    67+  67+  | 3    4    5    |
| 3    7    8    | 2    5    4    | 1    9    6    |
| 56   46   45   | 1    39   39   | 8    7    2    |
|----------------|----------------|----------------|
```

Using AICs I can get the same exclusion along with others depending on how the congruent links are used as weak or strong inferences, but navigating these is quite a long job.

That's a bit like producing a rabbit out of hat though, so I decided to document my approach: This grid has three cells with three candidates which represent the start points for a BUG survey. If one of the digits is mentally selected as the possible disruptor in one of these cells, it is a case of following our noses to build up a BUG from the remaining two using the surrounding bivalues.

Starting with (6)r6c6 as a disruptor, we need to include the (48) and (47) in row 6 to complete the BUG complement of two instances of each digit for the row. This then forces (48) & (47) in boxes 4 & 6 which then sets (78) in row 5 which also satisfies box 5. In column 6 we need (68) and (67) to even up the 7s and 8s. We then need (68) in r2 & b2 and (67) in r7 & b8 which successfully closes the BUG loop in c5 and sets (4)r2c5 as the second possible disruptor.

Now to try out this approach for the candidates in r1c1:

Code:

```
|----------------|----------------|----------------|
| 56+9 8    45+  | 46+  1    59   | 2    3    7    |
| 7    46+  2    | 3    468  68   | 5    1    9    |
| 59   1    3    | 57+  79   2    | 6    8    4    |
|----------------|----------------|----------------|
| 1    2    7    | 45+  34   35   | 9    6    8    |
| 48   5    6    | 9    78   1    | 47   2    3    |
| 48   3    9    | 67+  2    678  | 47   5    1    |
|----------------|----------------|----------------|
| 2    9    1    | 8    67   67   | 3    4    5    |
| 3    7    8    | 2    5    4    | 1    9    6    |
| 56+  46+  45+  | 1    39   39   | 8    7    2    |
|----------------|----------------|----------------|
```

Trial 1 using (56+9)r1c1. Box 1 is easily balanced as are columns 1,2, & 3, and hence box 7. We need (46)r1c4 to balance row 1, which forces us to use the other 3 bivalues in column 4 to balance it. Row 3 now has (5)r3c4 which must be balanced by (5)r3c1, but this introduces a third 5 in box1 so the operation fails.

Code:

```
|----------------|----------------|----------------|
| 59+6 8    45   | 46+  1    59   | 2    3    7    |
| 7    46   2    | 3    468  68   | 5    1    9    |
| 59+  1    3    | 57+  79+  2    | 6    8    4    |
|----------------|----------------|----------------|
| 1    2    7    | 45+  34   35   | 9    6    8    |
| 48   5    6    | 9    78   1    | 47   2    3    |
| 48   3    9    | 67+  2    678  | 47   5    1    |
|----------------|----------------|----------------|
| 2    9    1    | 8    67   67   | 3    4    5    |
| 3    7    8    | 2    5    4    | 1    9    6    |
| 56   46   45   | 1    39   39   | 8    7    2    |
|----------------|----------------|----------------|
```

Trial2 using (59+6)r1c1: We balance the houses in the following order box1, row3, c4, we get to the marked position and find we have no second 6 in r1 we can use to balance (6)r1c4 to give us a second failure.

Code:

```
|----------------|----------------|----------------|
| 69+5 8    45+  | 46+  1    59+  | 2    3    7    |
| 7    46+  2    | 3    46+8 68   | 5    1    9    |
| 59+  1    3    | 57+  79+  2    | 6    8    4    |
|----------------|----------------|----------------|
| 1    2    7    | 45+  34+  35+  | 9    6    8    |
| 48   5    6    | 9    78   1    | 47   2    3    |
| 48   3    9    | 67+  2    67+8 | 47   5    1    |
|----------------|----------------|----------------|
| 2    9    1    | 8    67+  67+  | 3    4    5    |
| 3    7    8    | 2    5    4    | 1    9    6    |
| 56+  46+  45+  | 1    39+  39+  | 8    7    2    |
```

|----------|----------|----------|

Trial3 using (69+5)r1c1: we can satisfy b1, b7, r1, r3, c4, c6 before we must make a decision about how to split (468)r2c5. The right choice is fairly easy to spot and we end up with this grid to find that we now have 21 cell latent BUG with a set of 3 possible disruptors - success! This produces:

(7=8)r5c5 - (8#2)r2c5,r6c6 =[BUG]= (5)r1c1 - (5=9)r2c1 - (9=7)r3c5 - Loop => r7c5 <> 7, r9c1 <>5

This exercise has shown me that when a grid is brimming with bivalues, this type of check can be made quicker than following various paths through an AIC net. However at no time have I yet made use of any imaginary candidates!

Please consider this an interim report, as I still have some mulling to do.

Back to top profile pm

eleven

Posted: Thu Jul 23, 2009 10:28 am Post subject: quote

Joined: 10 Feb 2008
Posts: 474

ronk wrote:

> ***Corollary 4*** *: Any placement of a candidate which forces a grid into a BUG+1 is a valid move. (example)*
>
> ... and set r8c5=6.

Nice.

Back to top profile pm

RW

Posted: Thu Jul 23, 2009 9:03 pm Post subject: quote

Joined: 16 Mar 2006
Posts: 981
Location: Finland

ronk wrote:

> but we could use **Jeff**'s ...
>
> ***Corollary 4*** *: Any placement of a candidate which forces a grid into a BUG+1 is a valid move. (example)*
>
> ... and set r8c5=6.

Was this corollary ever proven? All I can find is

Jeff wrote:

> I think the proposed corollary would work too if there is only one single non-BUG candidate left in the BUG grid after the placement of a digit.
>
> The reason being that: For any BUG grid that has only one non-BUG candidate, a unique solution can be obtained by placing the remaining non-BUG candidate in the poly-valued cell. It follows that any move that can create a BUG grid with one single non-BUG candidate is a valid move too.

I can't find anywhere a proof for the sentence *For any BUG grid that has only one non-BUG candidate, a unique solution can be obtained by placing the remaining*

non-BUG candidate in the poly-valued cell. Not that I suspect this would be false, but it would be nice to see a proof.

RW

Back to top profile pm

Myth Jellies

Posted: Thu Jul 23, 2009 9:21 pm Post subject: quote

Joined: 19 Sep 2005
Posts: 623

Interesting that if you take DPB's big BUG, my big BUG, and DPB's "rabbit" BUG, you get the following

Code:

```
        r1c1 r2c5 r6c6
MJ BUG   5    6    7
DPB BUG  5    8    8
Rabbit   -    4    6
```

and without even consulting the grid, you can place a 5 in r1c1 based only on this information.

David, I must have misunderstood. I thought you were looking for full BUG grids that used every unsolved cell by adding imaginary candidates.

Back to top profile pm

David P Bird

Posted: Fri Jul 24, 2009 1:02 am Post subject: quote

Joined: 16 Sep 2008
Posts: 138
Location: Middle England

RW wrote:

> I can't find anywhere a proof for the sentence *For any BUG grid that has only one non-BUG candidate, a unique solution can be obtained by placing the remaining non-BUG candidate in the poly-valued cell*. Not that I suspect this would be false, but it would be nice to see a proof.

If a puzzle has a unique solution we can be sure that every Unavoidable Set it contains will include at least one given to avoid having two ways of filling in the cells. So, assuming uniqueness we can say if this non-BUG candidate is false, we have two ways of solving the Unavoidable Set that would be left, but this would mean one of them should have been a given and as this isn't the case, the non-bug candidate must be true.

Now there is a class of puzzles that have an odd number of solutions. The ones with three solutions I've found all resolve down to a trivalue cell holding (abc), where if (a) is false we get a unavoidable set giving two solutions, and if it's true we get the third solution.

If the puzzle composer is expecting us to use uniqueness techniques, he can predict that we will set (a) true when we see the BUG threat, and so consider this type of puzzle to be acceptable (ie an Unavoidable Set without a given). Personally I don't agree unless this particular stance is clarified to the solver. This is because it obliges us to (falsely) assume uniqueness in reaching our solution.

A by-product of this is that for those of us that compose our own puzzles, we can't be sure that we have a unique solution if we have depended on a uniqueness based method when it was composed.

Back to top profile pm

eleven

Joined: 10 Feb 2008
Posts: 474

Posted: Fri Jul 24, 2009 1:42 am Post subject: quote

RW wrote:

> **ronk wrote:**
>
> > but we could use **Jeff**'s ...
> >
> > ***Corollary 4*** *: Any placement of a candidate which forces a grid into a BUG+1 is a valid move. (example)*
> >
> > ... and set r8c5=6.
>
> Was this corollary ever proven?

Hm, now I doubt, that its easy to prove.
From Jeff's definition this is a BUG+1:

Code:

```
--------------------
12 125 .  | .  .  .  |
 .  .  .  | .  .  .  |
 .  .  .  | 34 34 .  |
--------------------
12 12  .  | .  .  .  |
 .  .  .  | .  .  .  |
 .  .  .  | 34 34 .  |
--------------------
```

Back to top profile pm

RW

Joined: 16 Mar 2006
Posts: 981
Location: Finland

Posted: Fri Jul 24, 2009 3:21 am Post subject: quote

DPB, I understand very well the logic of the BUG technique. But I now realized that I quoted too little when I asked for the proof... Jeff says:

"For any BUG grid that has only one non-BUG candidate, a unique solution can be obtained by placing the remaining non-BUG candidate in the poly-valued cell. It follows that any move that can create a BUG grid with one single non-BUG candidate is a valid move too."

Does the first sentence automatically prove to the second? Or is there a possibility that if we arrive at a BUG+1 grid by making a bad elimination we have a grid with no solutions?

To prove this rule we must show that every BUG+1 grid always has a unique solution, or actually it's enough to show that any BUG+1 grid always has at least one solution. However, it is not enough to know that a BUG+0 grid always has zero

or multiple solutions.

[Edit:

Here's an example

Code:

```
*-------------------------------------------------*
| 1     28    3     | 4     5     6     | 7     29    89    |
| 4     5     7     | 3     89    29    | 12    6     18    |
| 6     28    9     | 7     18    12    | 4     5     3     |
|-----------------+-----------------+-----------------|
| 9     7     4     | 5     3     8     | 12    12    6     |
| 5     3     6     | 19    2     19    | 8     4     7     |
| 8     1     2     | 6     4     7     | 9     3     5     |
|-----------------+-----------------+-----------------|
| 27    9     1     | 28    6     3     | 5     78    4     |
| 27    6     5     | 28    19    4     | 3     78    19+8  |
| 3     4     8     | 19    7     5     | 6     19    2     |
*-------------------------------------------------*
```

BUG+1 grid with 0 solutions, which I arrived at by "mistakenly" eliminating a true candidate (the solved cells do have an unique solution). Jeff's corollary states that *any placement* that leads to a BUG+1 is valid. Now, I didn't reach this grid by placing a candidate, I reached it by eliminating a candidate. But how is it proven that a placement cannot lead to a grid like the grid above?

/Edit]

RW

Last edited by RW on Fri Jul 24, 2009 3:32 am; edited 1 time in total

Back to top profile pm

ronk

Posted: Fri Jul 24, 2009 3:28 am Post subject: quote

Joined: 02 Nov 2005
Posts: 2492
Location: Southeastern USA

I think a pursuit, if any, of a proof or disproof for Corollary 4 should be pursued on the original **BUG thread**.

eleven wrote:

From Jeff's definition this is a BUG+1:

Code:

```
--------------------
12 125 .  | .  .  .  |
 .  .  .  | .  .  .  |
 .  .  .  | 34 34 .  |
--------------------
12 12  .  | .  .  .  |
 .  .  .  | .  .  .  |
 .  .  .  | 34 34 .  |
--------------------
```

Although not explicitly stated, I think it's clear that a BUG is not meant to be a

composite of BUG-Lites.

Back to top profile pm

RW

Joined: 16 Mar 2006
Posts: 981
Location: Finland

Posted: Fri Jul 24, 2009 3:34 am Post subject: quote

> **ronk wrote:**
> Although not explicitly stated, I think it's clear that a BUG is not meant to be a composite of BUG-Lites.

And I think that a invalid BUG+1, with 0 solutions, will always be a composite of BUG-lites (it will have at least one independent BUG-lite+0 and one BUG-lite+1 pattern with 0 solutions). But the independent BUG-lite is not always easy to spot when dealing with big BUGs... And this is so far just a guess.

RW

Back to top profile pm

eleven

Joined: 10 Feb 2008
Posts: 474

Posted: Fri Jul 24, 2009 6:15 am Post subject: quote

My opinion is, that this corollary never can be proven from the theorem alone. The simple reason is, that the theorem only is true for valid grids.
But *Any Placement* will lead to an invalid grid, if it is wrong, and nothing is valid, what we know about valid grids without proving it seperately. So to prove something then, we first would need an exact definition of this kind of invalid grids.

[Added:]
Take this theorem:
In a unique puzzle any placement, which leads to a solution, is valid.

This is absolutely true and probably, what Jeff had in mind. But as i showed above (and RW demonstrated with a real life puzzle), in an invalid puzzle a BUG+1 not neccessarily has a unique solution.

So obviously Jeff made a mistake.

Back to top profile pm

David P Bird

Joined: 16 Sep 2008
Posts: 138
Location: Middle England

Posted: Fri Jul 24, 2009 8:03 am Post subject: quote

RW, I did wonder why you asked for a formal proof which is why I went on to discuss puzzles with missing givens and an odd number of solutions.

But now I have the opportunity of mounting a hobby horse of mine! I remember well how difficult it was for me to understand posts on Sudoku forums because correspondents had little or no regard for any other readers that might be trying to follow a discussion. For example, too often I find some arcane abbreviation defined somewhere in another thread is being reintroduced in a new one. This makes it

extremely difficult for a new reader to grasp what the heck anyone is talking about. It wouldn't cost a great deal of effort if we occasionally provided reminders of what the abbreviations stood for and/or shortcuts to their definitions, particularly if they are in minority use.

For the benefit of our silent readers I therefore deliberately try to compensate by using the full names for something before switching to abbreviations, and often taking things back a step further than I know any regular contributor would need. I don't do it to be condescending, but simply to help newcomers who in time could have useful contributions of their own to make, and I wish some others would follow suit now and then.

Back on topic, I too have misgivings about the corollary you meant to question, and some of the other corollaries are tantamount to "plugging-in" a candidate to see what happens which is against my principles.

Back to top profile pm

ronk

Posted: Fri Jul 24, 2009 9:18 am Post subject: quote

Joined: 02 Nov 2005
Posts: 2492
Location: Southeastern USA

> **eleven wrote:**
>
> Take this theorem:
> In a unique puzzle any placement, which leads to a solution, is valid.
>
> This is absolutely true and probably, what Jeff had in mind. But as i showed above (and RW demonstrated with a real life puzzle), in an invalid puzzle a BUG+1 not neccessarily has a unique solution.
>
> So obviously Jeff made a mistake.

Your example has multiple solutions ... and **RW**'s example has [a candidate missing] which would ultimately be true. Therefore, I consider neither to be a convincing counter-example.

[edit: By definition, the BUG technique is only applicable to puzzles with exactly one solution.]

Last edited by ronk on Fri Jul 24, 2009 12:03 pm; edited 1 time in total

Back to top profile pm

eleven

Posted: Fri Jul 24, 2009 11:27 am Post subject: quote

Joined: 10 Feb 2008
Posts: 474

> **ronk wrote:**
>
> Your example has multiple solutions ... and **RW**'s example has removed a candidate which would ultimately be true. Therefore, I consider neither to be a convincing counter-example.

Yes, there is none so far, i failed to find one manually (but note there are about 10^25 puzzles, where you could find one). But there is neither a proof nor a good argument for this corollary. So i think, my time is better invested in other things than trying to prove or disprove it.

Quote:

> By definition, the BUG technique is only applicable to puzzles with exactly one solution.

Thats the problem. Who tells you, that you dont arrive at a BUG+1 after a wrong placement (in a 0 solution grid)?

[Edit:]
This is a very subtile question.
Since your placement led to a valid solution, it (a posteriori) was perfectly valid (and i still find it elegant).
But this is not, what we expect from a technique. An application of a technique never leads to an invalid puzzle.
So, we would need a method to validate a BUG+1, that it gives a single solution.

Back to top profile pm

RW

Posted: Fri Jul 24, 2009 1:12 pm Post subject: quote

Joined: 16 Mar 2006
Posts: 981
Location: Finland

I wrote:

> I think that a invalid BUG+1, with 0 solutions, will always be a composite of BUG-lites (it will have at least one independent BUG-lite+0 and one BUG-lite+1 pattern with 0 solutions). ... And this is so far just a guess.

This was a false guess, here's a counterexample:

Code:

```
*-------------------------------------------------*
| 6     5     18    | 78    4     9     | 17    2     3     |
| 49    2     18    | 78    3     5     | 14+7  6     79    |
| 3     7     49    | 6     1     2     | 45    8     59    |
|-----------------+-----------------+-----------------|
| 5     8     6     | 9     2     1     | 3     7     4     |
| 24    1     47    | 3     5     78    | 28    9     6     |
| 29    3     79    | 4     6     78    | 25    1     58    |
|-----------------+-----------------+-----------------|
| 1     6     3     | 2     9     4     | 78    5     78    |
| 8     9     5     | 1     7     3     | 6     4     2     |
| 7     4     2     | 5     8     6     | 9     3     1     |
*-------------------------------------------------*
```

This can be split out into two separate BUG-lite+1 patterns.

ronk wrote:

> Your example has multiple solutions ... and **RW**'s example has [a candidate missing] which would ultimately be true. Therefore, I consider neither to be a convincing counter-example.

Ron, right now your reasoning is that because you haven't seen a counterexample, you assume the technique to be true. That's not how to prove a technique IMO. I also assume that it is true, but I would still like to see a proof. And I don't see how removing a candidate that ultimately would be true differs from placing a candidate that is false (which is exactly what might happen if this corollary turns out to be

false and we use it under the assumption that it is true).

RW

Back to top profile pm

David P Bird

Joined: 16 Sep 2008
Posts: 138
Location: Middle England

Posted: Fri Jul 24, 2009 3:51 pm Post subject: quote

Myth,

> **Quote:**
> and without even consulting the grid, you can place a 5 in r1c1 based only on this information.

- a very neat observation!

> **Quote:**
> I thought you were looking for full BUG grids that used every unsolved cell by adding imaginary candidates.

- that thought had never crossed my mind! The way I was looking at it was simply identifying when a set of cells had already been proved not to be an unavoidable set because one of the DP candidates had already been excluded using inferences external to the cell set.

I've had less free time today but sufficient to conclude that if I keep to my original thinking, your grid is too close to a solution for a nearly bug pattern to exist ie with one or two cells with a missing DP candidate. Switching back to the imagining candidates approach, say we have a cell (ac) where we would like to consider that (b) hasn't yet been eliminated. To build up a BUG pattern our cell needs to see other instances of (b) in each of the three houses it occupies. There just aren't any openings for doing this in your grid that I can find.

Generally for the larger BUGs there won't be enough bivalues around early in a solution, and there won't be enough combinational freedom left in the last throws. The window of opportunity should consequently be quite restricted, which is probably why no-one has tripped over any live cases bigger than URs yet.

Now your thinking is even further out of the box than mine because you're adding candidates to solved cells. Thinking about this though the number of combinations to test could get too large for the method to be quicker than plodding through the AICs (which we know will get there in the end). That's why I was looking for a method that could be employed.

After posting yesterday, I briefly tried the 'follow your nose" nose approach on other grids with reasonable success, but sometimes the disrupting digits turn out to be a conjugate pair which disappointingly gives us nothing new.

Back to top profile pm

Display posts from previous: All Posts Oldest First Go

newtopic postreply **Sudoku Players' Forums Forum Index -> Advanced solving techniques**

All times are GMT - 8 Hours
Goto page Previous 1, 2, 3, 4, **5**, 6 Next

Page 5 of 6

Jump to: Advanced solving techniques Go

You **cannot** post new topics in this forum
You **cannot** reply to topics in this forum
You **cannot** edit your posts in this forum
You **cannot** delete your posts in this forum
You **cannot** vote in polls in this forum