



# Sudoku Players' Forums

- [FAQ](#)
- [Search](#)
- [Memberlist](#)
- [Usergroups](#)
- [Register](#)
- [Profile](#)
- [Log in to check your private messages](#)
- [Log in](#)

## The BUG (Bivalue Universal Grave) principle

Goto page [Previous](#) [1](#), [2](#), [3](#) ... [10](#), [11](#), [12](#), [13](#), [14](#) [Next](#)



[Sudoku Players' Forums Forum Index](#) -> [Advanced solving techniques](#)

[View previous topic](#) :: [View next topic](#)

### Author

### Message

**ronk**

Posted: Tue Dec 19, 2006 7:50 pm Post subject: Re: BUG Type 3 - a definition



Joined: 02 Nov 2005  
 Posts: 2489  
 Location: Southeastern USA

**claudiarabia wrote:**

You have this BUG also with 3 cells with 3candidates or 4. They form a naked quad or even quint then, leaving other cells in-line with a single candidate. This is the BUG3-principle.

I've never heard of a "BUG3-principle." Do you have link to where that terminology might have originated?

TIA, Ron

[Back to top](#)



**ArkieTech**

Posted: Wed Dec 20, 2006 3:08 am Post subject:



Joined: 29 May 2006  
 Posts: 105  
 Location: USA Arkansas

claudiarabia said:

**Quote:**

**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*8	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
*-----*								

I also was confused about the BUG Type 3. This BUG appears rather seldom. It's solution lays always in one line. (If somebody shows me an example with a box I take this back). In this line (column or row), in our example column 7, you have at least two cells with three candidates, disturbing the BUG-symmetry. BUG-Symmetrie means, that in every line and in every box every candidate has to appear exactly twice and every cell has 2 candidates.

You single out now the two Candidates disturbing the BUG-symmetry. In your case these are the 7 in r2c7 and the 8 in r6c7. These two candidates form together with another cell of the respective column, here with r7c7, a naked triplet. By this method r1c7 and r5c7 stay with one candidate only. The rest is placing singles.

Thanks -- works for me -- I like it.

dan



[Back to top](#)

**emalvick**

Posted: Wed Dec 20, 2006 10:15 am Post subject:



Joined: 01 Aug 2005  
Posts: 13

**Mike Barker wrote:**

Most anything you can do with a Unique Rectangle you can do with a BUG (and more since BUGs are bigger). Links for several threads discussing BUGs are found in the [Collection of Solving Techniques](#). Check out the UR links as well (especially for the basic four types). Examples can be found in [the Local Zoo](#).

In light of this and Claudiarabia's example, I can see the analogies. I guess part of it is that while I understand the theory behind UR and even BUGs, I never find myself with a Type 3 situation with either. While I find situations that are close to being a Type 3 UR, I never have enough to actually have a Type 3 and instead usually end up with Type 4 UR's or utilizing forcing chains to make eliminations. With BUGs, I rarely look at analogies (except with BUG-lites) to UR's and instead look at forcing chains to make eliminations.

I am usually doing these puzzles by hand, so I often wait until I am down to about 3 or 4 poly-valued cells to even try utilizing the BUG. So, generally I don't really think much about the URs or BUGs in terms of a type. Heck, I don't even look at the AURs much differently.

Thanks for pointing out the links. I enjoy using the examples as I learn more from them, and it is easier than just doing puzzles hoping that examples will show up.

Erik

[Back to top](#)**emalvick**

Posted: Wed Dec 20, 2006 10:49 am    Post subject: Another BUG question, observations



Joined: 01 Aug 2005  
Posts: 13

I was reading through this thread yesterday (before I had asked about the Types of BUGs) and saw the example that had the + candidate for the BUG in a bi-valued cell rather than the tri-valued cell.

I wish I would have noticed that sooner, as I ran across a puzzle the other day that came down to an apparent BUG+2 situation (2 poly-valued cells). However, neither of the cells could be reduced to create a BUG. It turned out that the BUG candidates were actually in bi-value cells in the same block (for each case). Working with those two cells solved the puzzle.

I'm not sure that the situation is common, but I think I ran into it one other time in a BUG+1 situation. I wish I would have kept the puzzles, and will keep track of it the next time I run into that situation.

The key observation I make is that the polyvalued cell does not have a number that shows up 3 times in its r/c/b (perhaps in 2 units but not all 3 units). That usually draws me to another cell in the block that has the number with the most potential to be the + candidate (perhaps it showed up 3 times in 2 of the units).

So, while the Poly-valued cell couldn't be BUG reduced, a buddy cell could be (i.e. the Poly-valued cell helped, just indirectly).

Erik

[Back to top](#)**ronk**

Posted: Wed Dec 20, 2006 11:55 am    Post subject: Re: Another BUG question, observations



Joined: 02 Nov 2005  
Posts: 2489  
Location: Southeastern USA

**emalvick wrote:**

I'm not sure that the situation is common, but I think I ran into it one other time in a BUG+1 situation. I wish I would have kept the puzzles, and will keep track of it the next time I run into that situation.

The key observation I make is that the polyvalued cell does not have a number that shows up 3 times in its r/c/b (perhaps in 2 units but not all 3 units). That usually draws me to another cell in the block that has the number with the most potential to be the + candidate (perhaps it showed up 3 times in 2 of the units).

So, while the Poly-valued cell couldn't be BUG reduced, a buddy cell could be (i.e. the Poly-valued cell helped, just indirectly).

A pencilmark grid with a single poly-valued cell that is not a BUG+1:

**Code:**

```


```

5	4	67	23	9	26	8	37	1
3	9	2	14	17	8	67	5	46
78	1	68	35	67	45	39	2	49
-----+								
78	3	4	6	18	17	5	9	2
6	2	78	59	4	59	1	78	3
1	5	9	27	28	3	67	4	68
-----+								
29	8	5	49	26	146	39	13	7
29	7	1	8	3	29	4	6	5
4	6	3	17	5	79	2	18	89

For prior discussion see [here](#).

[Back to top](#)



**emalvick**

Posted: Wed Dec 20, 2006 12:45 pm Post subject: Re: Another BUG question, observations



Joined: 01 Aug 2005  
Posts: 13

**ronk wrote:**

A pencilmark grid with a single poly-valued cell that is not a BUG+1:

**Code:**

5	4	67	23	9	26	8	37	1
3	9	2	14	17	8	67	5	46
78	1	68	35	67	45	39	2	49
-----+								
78	3	4	6	18	17	5	9	2
6	2	78	59	4	59	1	78	3
1	5	9	27	28	3	67	4	68
-----+								
29	8	5	49	26	146	39	13	7
29	7	1	8	3	29	4	6	5
4	6	3	17	5	79	2	18	89

For prior discussion see [here](#).

Thanks Ronk,

I do understand that (and that discussion was the case I was thinking of), and the situation I had may not have been a BUG+1, which is why I stated apparent BUG. This where what I observed was interesting as one of the bi-valued cells in the 1 polyvalued cell puzzle had 3 of one of its possibilities in its row, column, and block (same block and row as the polyvalued cell). This number turned out to be the correct answer. Is this a coincidence? I wish I still had the puzzle to check it out.

The other puzzle I referred to (the 2 polyvalued case, an apparent BUG+2) had the same thing happen. a buddy cell for each one had the 3 candidates in each row, col, and box. One of those was correct. I used a forcing chain to reduce a common cell under the assumption that one of the extra candidates had to be correct.

Unfortunately, I hadn't read the thread you just pointed to beforehand, or I would

have saved both puzzles. It seems likely that in each of the cases, using the value that was not the extra candidate would have produced a BUG grid, which probably explains how I got the correct solution even though the grids were not technically BUGs to begin with (i.e. we don't want to put the puzzle into a BUG state).

[Back to top](#)



**claudiarabia**

Posted: Thu Dec 21, 2006 6:48 am Post subject: Re: Another BUG question, observations



Joined: 14 May 2006  
Posts: 287  
Location: Berlin

[quote="emalvick"]

**ronk wrote:**

It seems likely that in each of the cases, using the value that was not the extra candidate would have produced a BUG grid, which probably explains how I got the correct solution even though the grids were not technically BUGs to begin with (i.e. we don't want to put the puzzle into a BUG state).

There are many situations looking buglike but aren't. Of course you can avoid it, when you eliminate candidates in the bi-value-cells. Experience showed that the candidates in the poly-valued cells, whose reduction leads to the BUG-pattern are mostly easier to eradicate with short forcing chains than the ones in the bi-value-cells. For them you need usually longer and more complicated chains, a brute step. These are two methods of solving a puzzle.

Claudia

[Back to top](#)



**emalvick**

Posted: Thu Dec 21, 2006 8:10 am Post subject: Re: Another BUG question, observations



Joined: 01 Aug 2005  
Posts: 13

**claudiarabia wrote:**

There are many situations looking buglike but aren't. Of course you can avoid it, when you eliminate candidates in the bi-value-cells. Experience showed that the candidates in the poly-valued cells, whose reduction leads to the BUG-pattern are mostly easier to eradicate with short forcing chains than the ones in the bi-value-cells. For them you need usually longer and more complicated chains, a brute step. These are two methods of solving a puzzle.

Claudia

Since I am usually working by hand, I usually don't like using long complicated frames. Unfortunately, I didn't think much about what I was doing when I utilized the concepts from a true BUG+1 for the bi-valued cells (i.e. looked for the number in 3 r/c/b). It happened to work, and I can only guess that it worked because the other number probably would have created a BUG.

You know, part of the reason I've been trying to really understand the whole BUG

and UR (and AUR) concepts is that I am trying to avoid having to use too complicated chains since I am working things by hand. What this whole pseudo-BUG situation has taught me is that my method (although it worked) is not sound. I will, however, observe its behavior should the situation come up again. I do believe there is something there with the bi-valued cell having the extra candidate, and I'll see if it leads anywhere when I get it next time (and whether it makes sense).

Erik

[Back to top](#)



**claudiarabia**

Posted: Tue Jan 02, 2007 6:46 pm Post subject: A puzzle with two BUGs



Joined: 14 May 2006  
Posts: 287  
Location: Berlin

**Code:**

```

. . . . . . . . .
. 4 . 2 9 . . 7 .
. . 5 . . 7 6 . .
. . 4 . . 1 7 . .
. 2 . . 6 . . 9 .
. . 6 5 . . 4 . .
. . 1 3 . . 8 . .
. 9 . . 2 6 . 3 .
. . . . . . . . .

```

With non-forcing chain techniques and a Turbot fish, which eliminates the candidate 8 in r1c5 you get a BUG Type 4 which is connected with a BUG Type 1 afterwards. This is the first puzzle I saw with this constellation and it is also the first in which I spotted 4 candidates in a bug-cell.

Here the pencilmark-grid:

**Code:**

```

*-----*
| 17  17  9   | 6   4   3   | 2   58  58  |
| 6    4   8   | 2   9   5   | 3   7   1   |
| 2    3   5   | 18  18  7   | 6   4   9   |
+-----+-----+
| 58   58  4   | 9   3   1   | 7   26  26  |
| 17   2   3   | 47  6   48  | 15  9   58  |
| 9    17  6   | 5   78  2   | 4   18  3   |
+-----+-----+
| 4    6   1   | 3   57  9   | 8   25  27  |
| 58   9   7   | 18  2   6   | 15  3   4   |
| 3    58  2   | 47  1578# 48 | 9   156# 67 |
+-----+-----+
*-----*

```

BUG4: Cells r9c58 are the two BUG-cells. BUG Type 4 always has it's BUG-center-cells in one row as BUG type 3 has. We need the 1 in these two cells because row 9 needs a one. 7 and 8 in r9c5 and 5 in r9c8, who show up three times in lines and boxes, have to remain too because they avoid the deadly pattern. Only the 5 in r9c5 and the 6 in r9c8 can be eliminated.

You can place some singles now. Then the following grids shows up:

**Code:**

```

*-----*
| 17  17  9  | 6  4  3  | 2  58  58  |
| 6  4  8  | 2  9  5  | 3  7  1  |
| 2  3  5  | 18 18 7  | 6  4  9  |
+-----+-----+
| 58  58  4  | 9  3  1  | 7  6  2  |
| 17  2  3  | 47 6  48 | 15 9  58 |
| 9  17  6  | 5  78 2  | 4  18  3  |
+-----+-----+
| 4  6  1  | 3  5  9  | 8  2  7  |
| 58  9  7  | 18 2  6  | 15 3  4  |
| 3  58  2  | 47 178# 48 | 9  15  6  |
*-----*

```

A nice BUG type 1 easy to solve by placing 8 in cell r9c5 as the only number showing up thrice in each connected line and it's box. Creating that puzzle was it worth spending the German night in front of the computer.

 Claudia 

[Back to top](#)

 [profile](#)  [pm](#)  [www](#)

**ronk**

 Posted: Tue Jan 02, 2007 7:10 pm    Post subject: Re: A puzzle with two BUGs

 [quote](#)

Joined: 02 Nov 2005  
Posts: 2489  
Location: Southeastern USA

**claudiarabia wrote:**

**Code:**


```

*-----*
-----*
| 17  17  9  | 6  4  3  | 2  58
58  |
| 6  4  8  | 2  9  5  | 3  7  1
|
| 2  3  5  | 18 18 7  | 6  4  9
|
+-----+-----+
| 58  58  4  | 9  3  1  | 7  26
26  |
| 17  2  3  | 47 6  48 | 15  9
58  |
| 9  178 6  | 5  78 2  | 4  18  3
|
+-----+-----+
| 4  6  1  | 3  57  9  | 8  25
27  |
| 58  9  7  | 18 2  6  | 15  3  4
|
| 3  58  2  | 47 1578# 48 | 9  156#
67  |
*-----*
-----*

```

BUG4: Cells r9c58 are the two BUG-cells. BUG Type 4 always has it's BUG-center-cells in one row as BUG type 3 has. We need the 1 in these two cells because row 9 needs a one. 7 and 8 in r9c5 and 5 in r9c8, who

show up three times in lines and boxes, have to remain too because they avoid the deadly pattern. Only the 5 in r9c5 and the 6 in r9c8 can be eliminated.

Very nice example, but I'm not aware of a "BUG Type 4" definition. Would you please point me to a source 

For some time, it was customary on this thread to illustrate the non-BUG candidates as "xy+a...", as in the grid below. "xy" denotes the BUG candidates and "+a..." the one or more non-BUG candidates.

**Code:**

17	17	9		6	4	3		2	58	58
6	4	8		2	9	5		3	7	1
2	3	5		18	18	7		6	4	9
-----										
58	58	4		9	3	1		7	26	26
17	2	3		47	6	48		15	9	58
9	178	6		5	78	2		4	18	3
-----										
4	6	1		3	57	9		8	25	27
58	9	7		18	2	6		15	3	4
3	58	2		47	15+78	48		9	16+5	67

Then, to avoid the Bivalue Universal Grave, at least one of (r9c5=7 or r9c5=8) and r9c8=5 must be true. All three cases imply r9c5<>5.

[Back to top](#)



**claudiarabia**

Posted: Wed Jan 03, 2007 12:12 pm Post subject: Re: A puzzle with two BUGs



Joined: 14 May 2006  
Posts: 287  
Location: Berlin

**ronk wrote:**

[For some time, it was customary on this thread to illustrate the non-BUG candidates as "xy+a...", as in the grid below. "xy" denotes the BUG candidates and "+a..." the one or more non-BUG candidates.

**Code:**

17	17	9		6	4	3		2	58	58
6	4	8		2	9	5		3	7	1
2	3	5		18	18	7		6	4	9
-----										
58	58	4		9	3	1		7	26	26
17	2	3		47	6	48		15	9	58
9	178	6		5	78	2		4	18	3
-----										
4	6	1		3	57	9		8	25	27
58	9	7		18	2	6		15	3	4
3	58	2		47	15+78	48		9	16+5	67

Then, to avoid the Bivalue Universal Grave, at least one of (r9c5=7 or r9c5=8) and r9c8=5 must be true. All three cases imply r9c5<>5.



You are totally right. I will follow this notation in future. A definition of BUG 4 you find in the Explainer. That the two BUG Type4-Cells with the extra candidates are always in one line is my observation. I wonder if there are examples where the BUG-Center-cells are in one box. If somebody brings me an example I will rectify my assumption.

Claudia

[Back to top](#)



**claudiarabia**

Posted: Wed Jul 04, 2007 8:43 am Post subject: BUG Type 1+BUG Type 2



Joined: 14 May 2006  
Posts: 287  
Location: Berlin

Today I made a puzzle in which I originally wanted to produce through highly symmetrical placement of numbers some cycles. This sudoku SE 9.0 ends up according to the Explainer in a BUG Type 2 which is easily transformed into a final BUG Type 1-situation.

**Code:**

```
8 . . 6 . . . . 5
. . . . . 7 . 2 .
. . 1 3 . . . . .
6 . . . . . . 7 .
. 2 . . . . . . .
5 . 4 . . . 3 . 1
. . . 4 . . 8 . .
. 9 . . 2 . . . .
4 . . 5 . 1 . . 6
```

**Code:**

```
*-----*
| 8  37  79  | 6  49  2  | 1  34  5  |
| 39  4  5   | 19  18  7  | 6  2  38  |
| 2   6  1   | 3   45  58  | 79  48  79  |
|-----+-----+-----|
| 6   1  38  | 89  35  *45+3 | 2   7  49  |
| 39  2  79  | 17  16  34  | 5   68  48  |
| 5   78  4   | 2   67  89  | 3   69  1   |
|-----+-----+-----|
| 7   5  36  | 4   39  *69+3 | 8   1  2   |
| 1   9  *68+3 | 78  2   6-3  | 4   5  37  |
| 4   38  2   | 5   78  1   | 79  39  6   |
|-----+-----+-----|
*-----*
```

Here the BUG Type 2-pattern. If all the 3 in r4c6, r7c6 and r8c3 would be missing, then you would have the Bivalue Grave, highly symmetric, but without a solution. So at least on the the three 3s has to remain in the grid. Regardless, which of the three 3s will stay, each of them eliminate the 3 in r8c6. Placing the 6 into r8c6 and afterwards in r7c3 you will have the final BUG Type 1-Situation below:

**Code:**

```
*-----*
```

8	37	79	6	49	2	1	34	5
39	4	5	19	18	7	6	2	38
2	6	1	3	45	58	79	48	79
6	1	38	89	35	*3-45	2	7	49
39	2	79	17	16	34	5	68	48
5	78	4	2	67	89	3	69	1
7	5	6	4	39	39	8	1	2
1	9	38	78	2	6	4	5	37
4	38	2	5	78	1	79	39	6

It's clear now, which 3 remains to avoid the deadly BUG-pattern and the puzzle is solved.

Claudia

Last edited by claudiarabia on Thu Jul 05, 2007 12:08 am; edited 1 time in total

[Back to top](#)

[profile](#) [pm](#) [www](#)

re'born

Posted: Wed Jul 04, 2007 1:59 pm Post subject: Re: BUG Type 1+BUG Type 2



Joined: 31 May 2007  
 Posts: 552  
 Location: Wilmington, MA

**claudiarabia wrote:**

Today I made a puzzle in which I originally wanted to produce through highly symmetrical placement of numbers some cycles. This sudoku SE 9.0 ends up according to the Explainer in a BUG Type 2 which is easily transformed into a final BUG Type 1-situation.

**Code:**

```

8 . . 6 . . . 5
. . . . . 7 . 2 .
. . 1 3 . . . . .
6 . . . . . . 7 .
. 2 . . . . . . .
5 . 4 . . . 3 . 1
. . . 4 . . 8 . .
. 9 . . 2 . . . .
4 . . 5 . 1 . . 6
    
```

**Code:**

*-----*								
8	37	79	6	49	2	1	34	5
39	4	5	19	18	7	6	2	38
2	6	1	3	45	58	79	48	79
-----								
6	1	38	89	35	*345	2	7	49
39	2	79	17	16	34	5	68	48
5	78	4	2	67	89	3	69	1
-----								
7	5	36	4	39	*369	8	1	2
1	9	*368	78	2	6-3	4	5	37
4	38	2	5	78	1	79	39	6
*-----*								

Here the BUG Type 2-pattern. If all the 3 in r4c6, r7c6 and r8c3 would be

missing, then you would have the Bivalue Grave, highly symmetric, but without a solution. So at least on the the three 3s has to remain in the grid. Regardless, which of the three 3s will stay, each of them eliminate the 3 in r8c6. Placing the 6 into r8c6 and afterwards in r7c3 you will have the final BUG Type 1-Situation below:

Alternatively, you can note that if  $r4c3=3$ , then  $r5c6=3$  and this will again give a BUG situation. Therefore,  $r4c3=8$ , solving the puzzle.

[Back to top](#)



**RW**

Posted: Sat Jul 25, 2009 11:41 am Post subject:



Joined: 16 Mar 2006  
Posts: 981  
Location: Finland

Moving [this](#) discussion into the thread where it belongs...

I'm digging up this old thread to question the validity of this four years old corollary:

**Quote:**

Corollary 4: Any placement of a candidate which forces a grid into a BUG+1 is a valid move.

According to this, if I place a candidate and then arrive at a BUG+1 grid through use of logical techniques, then the candidate I placed was valid. I would like to emphasize that "forces a grid into a BUG+1" must be interpreted as "can be reduced to a BUG+1 through use of logical techniques". The techniques allowed cannot be limited to a specific set.

Now for the counter-evidence:

(This example is probably not the best, there's a very long way from the placement to the BUG+1. But if I can find this situation manually in 15 minutes, then I'm sure someone with computing power and coding skills can find a better example quite easily.)

**Code:**

```
*-----*
|.3|4..|.9|
|45|.8|.2|
|8.9|32|.4..|
|---+---+---|
|28.|713|...|
|3..|.2|1..|
|...|8..|...|
|---+---+---|
|54|.3|.9.|
|732|.98|5.4|
|9.8|5.4|2..|
*-----*
```

(minimal 37 by Havard)

If I place candidate 5 (which is false) in r4c8, I will soon arrive here:

**Code:**

16	2	3	4	57	1567	67	8	9
4	5	17	69	8	1679	367	137	2
8	167	9	3	2	167	4	17	5
2	8	4	7	1	3	9	5	6
3	79	57	69	56	2	1	4	8
16	169	156	8	4	59	37	2	37
5	4	16	2	3	67	8	9	17
7	3	2	1	9	8	5	6	4
9	16	8	5	67	4	2	37	137

From here I can go on using chains to eliminate a few candidates and arrive at this BUG+1 grid:

**Code:**

16	2	3	4	57	15	67	8	9
4	5	17	69	8	79	36	13	2
8	67	9	3	2	16	4	17	5
2	8	4	7	1	3	9	5	6
3	79	57	69	56	2	1	4	8
16	19	56+1	8	4	59	37	2	37
5	4	16	2	3	67	8	9	17
7	3	2	1	9	8	5	6	4
9	16	8	5	67	4	2	37	13

I do not need to explain what chains I used, because that is of no importance. What is important is that the puzzle can be reduced to this state, not how it is reduced. Since the puzzle is already in an invalid state, there will exist chains to prove that every single candidate is invalid, every candidate will lead to a contradiction, therefore I don't even need to find the specific chains myself.

So, I placed a false vandidate and arrived at a BUG+1 grid... Before I officially bury corollary 4, I would still like to ask if someone can find any flaws in my reasoning. Is there something wrong with my BUG+1 grid (apart from it having 0 solutions), or have I missed something else?

RW

[Back to top](#)



**ronk**

Posted: Sat Jul 25, 2009 12:27 pm Post subject:



Joined: 02 Nov 2005  
Posts: 2489  
Location: Southeastern USA

**RW wrote:**

So, I placed a false vandidate and arrived at a BUG+1 grid... Before I officially bury corollary 4, I would still like to ask if someone can find any flaws in my reasoning. Is there something wrong with my BUG+1 grid (apart from it having

0 solutions), or have I missed something else?

I think the *any candidate* of Corollary 4 was meant in the context of a possible BUG+n grid where, were all the extra candidates to be false, both dual candidate and single candidate cells remain.

Even if that's not what was meant, it's the only way the corollary has ever been used AFAIK.

[Back to top](#)



Display posts from previous:



**Sudoku Players'**  
**Forums Forum Index** - **Goto page** [Previous](#) [1](#), [2](#), [3](#) ... [10](#), [11](#), [12](#), [13](#), [14](#) [Next](#)  
> **Advanced solving techniques**

All times are GMT - 8 Hours

**Page 11 of 14**

Jump to:

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

Powered by phpBB © 2001, 2005 phpBB Group