Sudoku Players' Forums

Fully supersymmetric chains

Goto page Previous 1, 2, 3 ... , 19, 20, 21 Next

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

View previous topic :: View next topic

Author | **Message**

Allan Barker

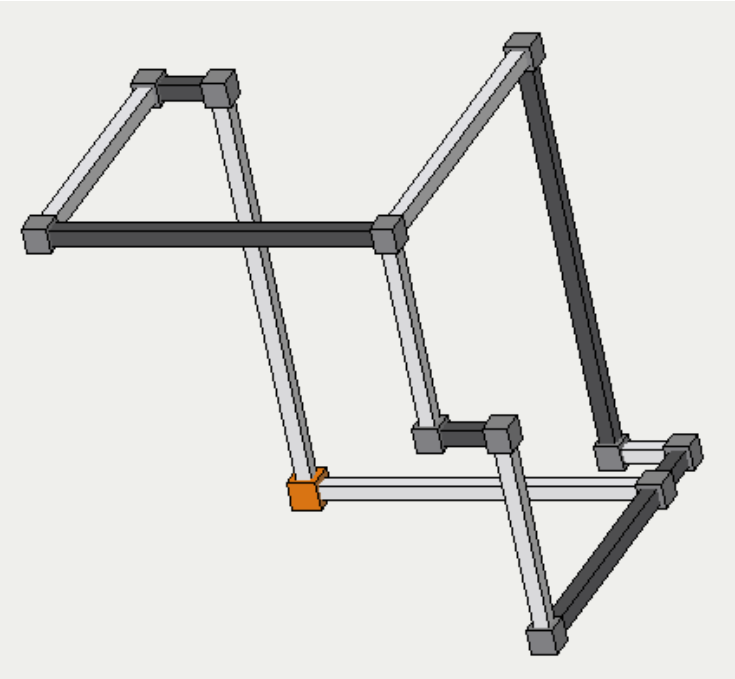
Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Posted: Tue Dec 29, 2009 6:20 am Post subject: quote

Denis Berthier wrote:

> What you have shown from the beginning and you continue to show is 2D diagrams.

".... are 3D diagrams or their 2D projections". Even Xsudo's 2D graphics is drawn by projection (which makes them look "soft"). The definition of multiple 2D spaces is yours entirely, I find it makes logical relations harder to grasp. Just to make a point, look at the 3D logic below and identify your 2D spaces, but be careful, Xsudo has 720 degree rotation option, the orientation could be anything.



I wait your answer or what I now call an "eleven", a quick, professional acknowledgement of a logical point. :)

Denis Berthier wrote:

> I don't know if it'd be easy to do, but if you want to use XSudo to display oriented chains without depriving them of their chain structure, you should:
> - provide an input format for the adequate syntax (nrc)
> - draw arrows instead of simple lines between consecutive rlc's and llc's
> - display the additional lines justifying the z- and t- candidates as light dotted lines (need not be oriented) / don't display them at all; the best being the possibility of switching from one mode to the other: highlight only the main structure (the whip) vs show also the details. .

I have thought about doing this, maybe it's a good time. As you mention, a truth/link diagram may have lost the details of the process. However, it retains all logical information relevant to Sudoku, independent of process. It would be good to combine the two formats.

My only worry might be the notation, which is much more work to read in software than to write. If the notation changes then that's all lost.

Back to top profile pm www

denis_berthier

Posted: Tue Dec 29, 2009 7:09 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

Allan Barker wrote:

> **Denis Berthier wrote:**
>
> > What you have shown from the beginning and you continue to show is 2D diagrams.
>
> "..... are 3D diagrams or their 2D projections". Even Xsudo's 2D graphics is drawn by projection (which makes them look "soft"). The definition of multiple 2D spaces is yours entirely, I find it makes logical relations harder to grasp.
> ...
> I wait your answer or what I now call an "eleven", a quick, professional acknowledgement of a logical point.

Of course. But I meant it the other way round: you have shown 2D diagrams - and 3D - not 1D. Chains are 1D, not 2D or 3D.

Allan Barker wrote:

> **Denis Berthier wrote:**
>
> > I don't know if it'd be easy to do, but if you want to use XSudo to display oriented chains without depriving them of their chain structure, you should:
> > - provide an input format for the adequate syntax (nrc)
> > - draw arrows instead of simple lines between consecutive rlc's and llc's
> > - display the additional lines justifying the z- and t- candidates as light dotted lines (need not be oriented) / don't display them at all; the best being the possibility of switching from one mode to the other: highlight only the main structure (the whip) vs show also the details. .
>
> I have thought about doing this, maybe it's a good time. As you mention, a truth/link diagram may have lost the details of the process. However, it retains all logical information relevant to Sudoku, independent of process.

It has lost more than details. It has lost the major information about the sequence, the ordering of ll and rl candidates necessary to define a unique chain.

Allan Barker wrote:

> It would be good to combine the two formats.
> My only worry might be the notation, which is much more work to read in software than to write. If the notation changes then that's all lost.

After the slight modification I've introduced recently, I have no reason to change the nrc notation. I think it now perfectly reflects the two complementary aspects of chains: chains of candidates and chains of 2D cells.

What about the absolute rating? I consider it as a very interesting idea (yours). For each n, it would give an upper bound of what can be solved using combinations of at most n 2D-cells (n truths) in all the possible ways.

Back to top profile pm www

Mauricio

Posted: Mon Jan 18, 2010 1:57 am Post subject: quote

Joined: 22 Mar 2006

denis_berthier wrote:

> As I mentioned in the previous post, #187 (SER = 9.4) is one of the two grids in the top1465

Posts: 1101

collection that can't be solved by nrczt-whips but can by nrczt-braids (as can be verified with a simple T&E procedure).

First tentative solution with whips:
***** SudoRules version 13.7w-bis *****
6.....3...5..9..8...2..6..98.....7...7..5..4......1..51..3..5...4..2..6...8..7..2
hidden-singles ==> r4c3 = 5, r7c2 = 2
interaction row r4 with block b5 for number 4 ==> r6c5 <> 4, r6c4 <> 4
nrczt-whip-cn[11] {n4 n1}r3c7 - {n1 n9}r9c7 - {n9 n8}r8c7 - n8{r8c9 r5c9} - n1{r5c9 r5c3} - n1{r4c2 r1c2} - n9{r1c2 r1c3} - n9{r8c3 r8c1} - {n9 n5}r8c6 - n1{r9c5 r8c4} - {n1r9c5 .} ==> r2c7 <> 4
;;; end common part
nrczt-whip[21] n8{r8c9 r7c9} - n4{r7c9 r9c7} - {n4 n1}r3c7 - n1{r8c7 r8c9} - n7{r8c9 r7c8} - {n7 n5}r3c8 - {n5 n2}r1c8 - n1{r1c8 r4c8} - n1{r4c2 r1c2} - n9{r1c2 r1c3} - {n9 n6}r7c3 - {n6 n4}r7c5 - {n4 n9}r7c6 - {n9 n5}r8c6 - n5{r9c4 r1c4} - n1{r1c4 r2c4} - n2{r2c4 r2c6} - n2{r4c6 r4c4} - n9{r4c4 r4c2} - {n9 n3}r9c2 - {n3r9c8 .} ==> r8c4 <> 8
GRID 187 NOT SOLVED. 55 VALUES MISSING.
Here again, we notice a very long whip (length 21), an indication that this puzzle has few chains.

Let's now solve it with braids:
...

I think your code fails to detect some whips, my first whip[20] eliminates the same candidate as your whip[21], then I detect a second whip[20].

Code:

```
600000300050090080002006009800000700070050040000001005100300500040020060008007002
r4c3=5
r7c2=2
r6c4<>4, whip[1] r4n4{c6 .}
r6c5<>4, whip[1] r4n4{c6 .}
r2c7<>4, whip[11] r3c7{n4 n1} - r9c7{n1 n9} - r8c7{n9 n8} - c9n8{r7 r5} - r5n1{c9
c3} - r2n1{c3 c4} - r1n1{c4 c2} - r1n9{c2 c3} - r7n9{c3 c6} - r8c4{n9 n5} -
r8c6{n5 .}
r8c4<>8, whip[20] r7n8{c5 c9} - b9n4{r7c9 r9c7} - r3c7{n4 n1} - r8n1{c7 c9} -
b9n7{r8c9 r7c8} - r3c8{n7 n5} - r1c8{n5 n2} - c8n1{r1 r4} - c2n1{r4 r1} - r1n9{c2
c3} - r7n9{c3 c6} - r8c6{n9 n5} - c1n5{r8 r9} - c4n5{r9 r1} - b2n1{r1c4 r2c4} -
r2n2{c4 c6} - r4n2{c6 c4} - r4n9{c4 c2} - r9n9{c2 c8} - r8c7{n9 .}
r8c7<>1, whip[20] r3c7{n1 n4} - c9n4{r1 r7} - r9c7{n4 n9} - r9c8{n9 n3} - r9c2{n3
n6} - r7n6{c3 c5} - r7n8{c5 c6} - r7n9{c6 c3} - r1n9{c3 c2} - r6c2{n9 n3} -
r4c2{n3 n1} - r5n1{c3 c9} - r5n3{c9 c6} - r5c3{n3 n6} - r6c3{n6 n4} - c1n4{r6 r2}
- r2c6{n4 n2} - r2c7{n2 n6} - r2c9{n6 n7} - r1c9{n7 .}
r8c9<>7, whip[7] b9n3{r8c9 r9c8} - b9n1{r9c8 r9c7} - r3c7{n1 n4} - r1c9{n4 n1} -
r5n1{c9 c3} - r2n1{c3 c4} - r8n1{c4 .}
r7c3<>7, whip[1] r8n7{c1 .}
r6c3<>6, whip[5] r7c3{n6 n9} - r1n9{c3 c2} - r6c2{n9 n3} - r4c2{n3 n1} - r5c3{n1
.}
r9c8<>9, whip[14] b9n3{r9c8 r8c9} - b9n1{r8c9 r9c7} - b9n4{r9c7 r7c9} - c9n8{r7
r5} - r5n1{c9 c3} - c3n6{r5 r7} - r7c5{n6 n8} - r6n8{c5 c4} - r3n8{c4 c2} -
c2n1{r3 r1} - c5n1{r1 r3} - r3n3{c5 c1} - r2n3{c1 c6} - r5n3{c6 .}
r5c7<>9, whip[14] c8n9{r4 r7} - r7n7{c8 c9} - b9n4{r7c9 r9c7} - r3c7{n4 n1} -
r1c9{n1 n4} - r2c9{n4 n6} - c7n6{r2 r6} - b6n8{r6c7 r5c9} - r5n1{c9 c3} - r5n6{c3
c4} - r4n6{c4 c2} - r9n6{c2 c5} - c5n1{r9 r1} - r2n1{c4 .}
r1c8<>7, whip[16] b3n2{r1c8 r2c7} - r2n6{c7 c9} - c9n7{r2 r7} - r7c8{n7 n9} -
c7n9{r8 r6} - c7n6{r6 r5} - b6n8{r5c7 r5c9} - r5n1{c9 c3} - r2n1{c3 c4} - r8n1{c4
c9} - r9c8{n1 n3} - r6c8{n3 n2} - c1n2{r6 r5} - b4n9{r5c1 r4c2} - r9c2{n9 n6} -
r7c3{n6 .}
r9c1<>9, whip[13] r9n5{c1 c4} - r3n5{c4 c8} - c8n7{r3 r7} - r7n9{c8 c6} - r8c6{n9
n8} - r7n8{c5 c9} - r7n4{c9 c5} - b8n6{r7c5 r9c5} - r4c5{n6 n3} - r5c6{n3 n2} -
r5c1{n2 n3} - r3n3{c1 c2} - r9c2{n3 .}
r6c3<>9, whip[11] r6n4{c3 c1} - c1n2{r6 r5} - c1n9{r5 r8} - c7n9{r8 r9} - r7n9{c8
c6} - r7c3{n9 n6} - r9c2{n6 n3} - b4n3{r4c2 r5c3} - r5c6{n3 n8} - r6n8{c4 c7} -
r8c7{n8 .}
r5c1<>3, whip[13] c1n2{r5 r6} - c1n9{r6 r8} - r8c7{n9 n8} - r8c6{n8 n5} - r8c4{n5
n1} - r8c9{n1 n3} - c3n3{r8 r2} - r3n3{c1 c5} - c5n1{r3 r1} - c5n7{r1 r6} -
r6n8{c5 c4} - r3n8{c4 c2} - b1n1{r3c2 .}
r6c2<>9, whip[10] r1n9{c2 c3} - r7c3{n9 n6} - r9c2{n6 n3} - b7n9{r9c2 r8c1} -
c7n9{r8 r9} - c8n9{r7 r4} - c8n3{r4 r6} - r6c3{n3 n4} - r6c1{n4 n2} - r5c1{n2 .}
r5c3<>9, whip[12] c1n9{r5 r8} - r8c7{n9 n8} - r8c6{n8 n5} - r8c4{n5 n1} - r8c9{n1
n3} - r5n3{c9 c6} - c5n3{r4 r3} - c5n1{r3 r1} - c3n1{r1 r2} - c2n1{r1 r4} -
c9n1{r4 r5} - c9n8{r5 .}
r1c3<>4, whip[4] r6n4{c3 c1} - c1n2{r6 r5} - c1n9{r5 r8} - c3n9{r8 .}
r2c4<>4, whip[12] b1n4{r2c1 r3c1} - r3c7{n4 n1} - r2n1{c7 c3} - r5n1{c3 c9} -
c8n1{r4 r9} - b9n3{r9c8 r8c9} - r4c9{n3 n6} - r2c9{n6 n7} - c1n7{r2 r8} - r8c3{n7
n9} - r8c7{n9 n8} - c9n8{r7 .}
r4c2<>9, whip[13] c1n9{r5 r8} - r8n7{c1 c3} - r8n3{c3 c9} - r9c8{n3 n1} - r4n1{c8
c9} - r5n1{c7 c3} - r5n3{c3 c6} - c5n3{r4 r3} - c5n1{r3 r1} - r1c2{n1 n8} -
```

```
r3n8{c2 c4} - b5n8{r5c4 r6c5} - c5n7{r6 .}
r8c1<>9, whip[1] b4n9{r6c1 .}
r6c1<>3, whip[2] c1n2{r6 r5} - c1n9{r5 .}
r6c1<>4, whip[2] c1n2{r6 r5} - c1n9{r5 .}
r6c3=4
r6c4<>2, whip[3] r6c1{n2 n9} - b6n9{r6c7 r4c8} - r4n2{c8 .}
r2c7<>1, whip[8] r3c7{n1 n4} - r9c7{n4 n9} - r8c7{n9 n8} - c9n8{r7 r5} - r5n1{c9
c3} - c3n6{r5 r7} - r9c2{n6 n3} - b4n3{r6c2 .}
r4c9<>1, whip[3] r5n1{c7 c3} - r2n1{c3 c4} - r8n1{c4 .}
r3c4<>1, whip[7] r8n1{c4 c9} - b3n1{r1c9 r1c8} - r3c7{n1 n4} - r1c9{n4 n7} -
r1c3{n7 n9} - c2n9{r1 r9} - r9c7{n9 .}
r6c7<>9, whip[7] r6c1{n9 n2} - r6c8{n2 n3} - r4c9{n3 n6} - r2n6{c9 c7} - c7n2{r2
r5} - r4c8{n2 n1} - r9c8{n1 .}
r7c8<>9, whip[1] c7n9{r9 .}
r7c8=7
r5c6<>9, whip[7] r7n9{c6 c3} - c3n6{r7 r5} - r5n3{c3 c9} - r5n1{c9 c7} -
b6n8{r5c7 r6c7} - c7n2{r6 r2} - c7n6{r2 .}
r6c5<>6, whip[7] c4n6{r4 r9} - r7n6{c5 c3} - c2n6{r9 r4} - r4c9{n6 n3} - r4c5{n3
n4} - b8n4{r7c5 r7c6} - r7n9{c6 .}
r6c7<>2, whip[7] r6c1{n2 n9} - r5n9{c1 c4} - r4n9{c4 c8} - r6c8{n9 n3} - r4c9{n3
n6} - r5n6{c7 c3} - r6c2{n6 .}
r8c3<>9, whip[5] r7c3{n9 n6} - r9c2{n6 n3} - r6c2{n3 n6} - r6c7{n6 n8} - r8c7{n8
.}
r9c2<>3, whip[2] b7n6{r9c2 r7c3} - b7n9{r7c3 .}
r1c9<>1, whip[6] r3c8{n1 n5} - r1c8{n5 n2} - c7n2{r2 r5} - r5n1{c7 c3} - r2n1{c3
c4} - r8n1{c4 .}
r6c5<>8, whip[6] r6c7{n8 n6} - r4c9{n6 n3} - c5n3{r4 r3} - c5n7{r3 r1} - c9n7{r1
r2} - r2n6{c9 .}
r2c6<>4, whip[7] c1n4{r2 r3} - b3n4{r3c7 r1c9} - c9n7{r1 r2} - b1n7{r2c1 r1c3} -
c3n9{r1 r7} - r7c6{n9 n8} - r7c9{n8 .}
r5c9<>8, whip[5] r6c7{n8 n6} - r2c7{n6 n2} - r2c6{n2 n3} - r5n3{c6 c3} - r6c2{n3
.}
r8c7<>8, whip[1] c9n8{r7 .}
r8c7=9
r5c7<>1, whip[2] r3c7{n1 n4} - r9c7{n4 .}
r1c3<>1, whip[3] r5n1{c3 c9} - r2n1{c9 c4} - r8n1{c4 .}
r1c4<>1, whip[3] r8n1{c4 c9} - r2n1{c9 c3} - r5n1{c3 .}
r9c4<>1, whip[3] r8n1{c4 c9} - r2n1{c9 c3} - r5n1{c3 .}
r7c6<>8, whip[4] r7c9{n8 n4} - r1c9{n4 n7} - r1c3{n7 n9} - r7n9{c3 .}
r1c2<>9, whip[6] r9c2{n9 n6} - r7c3{n6 n9} - r7c6{n9 n4} - r9c5{n4 n1} - r1n1{c5
c8} - c7n1{r3 .}
r1c3=9
r7c3=6
r9c2=9
r7c6=9
r3c5<>4, whip[2] r1n4{c4 c9} - r7n4{c9 .}
r4c6<>3, whip[4] r4c9{n3 n6} - c2n6{r4 r6} - c2n3{r6 r3} - r2n3{c3 .}
r6c7<>6, whip[4] r4c9{n6 n3} - r5c9{n3 n1} - r5c3{n1 n3} - r6c2{n3 .}
r6c7=8
r8c9<>3, whip[4] r4c9{n3 n6} - r5n6{c7 c4} - r5n8{c4 c6} - r8n8{c6 .}
r9c8=3
r9c1=5
r6c4<>9, whip[2] r6c1{n9 n2} - r6c8{n2 .}
r4c5<>3, whip[3] r4c9{n3 n6} - c2n6{r4 r6} - r6n3{c2 .}
r1c5<>4, whip[3] r4c5{n4 n6} - r9n6{c5 c4} - b8n4{r9c4 .}
r1c4<>7, whip[4] r1c9{n7 n4} - r7n4{c9 c5} - r4c5{n4 n6} - r6c4{n6 .}
r1c8<>1, whip[4] r1c2{n1 n8} - r1c5{n8 n7} - r1c9{n7 n4} - r3c7{n4 .}
r2c6<>2, whip[4] r1n2{c4 c8} - r6c8{n2 n9} - r4n9{c8 c4} - r4n2{c4 .}
r2c6=3
r6c5=3
r6c2=6
r6c4=7
r5c9<>6, whip[2] r5n1{c9 c3} - r5n3{c3 .}
r1c5<>7, whip[3] r1n1{c5 c2} - r2c3{n1 n7} - r3n7{c1 .}
r1c9=7
r3c5=7
r3c4<>4, whip[1] r1n4{c6 .}
r1c4<>8, whip[2] r1c2{n8 n1} - r1c5{n1 .}
r1c6<>8, whip[2] r1c2{n8 n1} - r1c5{n1 .}
r1c5<>8, whip[3] r3c4{n8 n5} - r8c4{n5 n1} - c5n1{r9 .}
Singles
Most difficult rule: NRCZT Whip[20]
```

Back to top profile pm

Allan Barker Posted: Mon Jan 18, 2010 3:41 am Post subject: quote

Mauricio wrote:

Joined: 21 Feb 2008

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

denis_berthier wrote:

> As I mentioned in the previous post, #187 (SER = 9.4) is one of the two grids in the top1465 collection that can't be solved by nrczt-whips but can by nrczt-braids (as can be verified with a simple T&E procedure).
>
> First tentative solution with whips:
> ***** SudoRules version 13.7w-bis *****
> 6.....3...5..9..8...2..6..98.....7...7..5..4.....1..51..3..5...4..2..6...8..7..2
>
> Here again, we notice a very long whip (length 21), an indication that this puzzle has few chains.
>
> Let's now solve it with braids:
> ...

I think your code fails to detect some whips, my first whip[20] eliminates the same candidate as your whip[21], then I detect a second whip[20].

Code:

```
600000300050090080002006009800000700070050040000001005100300500040020060008007002
r4c3=5
r7c2=2
r6c4<>4, whip[1] r4n4{c6 .}
r6c5<>4, whip[1] r4n4{c6 .}
r2c7<>4, whip[11] r3c7{n4 n1} - r9c7{n1 n9} - r8c7{n9 n8} - c9n8{r7 r5} - r5n1{c9
c3} - r2n1{c3 c4} - r1n1{c4 c2} - r1n9{c2 c3} - r7n9{c3 c6} - r8c4{n9 n5} -
r8c6{n5 .}
r8c4<>8, whip[20] r7n8{c5 c9} - b9n4{r7c9 r9c7} - r3c7{n4 n1} - r8n1{c7 c9} -
b9n7{r8c9 r7c8} - r3c8{n7 n5} - r1c8{n5 n2} - c8n1{r1 r4} - c2n1{r4 r1} - r1n9{c2
c3} - r7n9{c3 c6} - r8c6{n9 n5} - c1n5{r8 r9} - c4n5{r9 r1} - b2n1{r1c4 r2c4} -
r2n2{c4 c6} - r4n2{c6 c4} - r4n9{c4 c2} - r9n9{c2 c8} - r8c7{n9 .}
r8c7<>1, whip[20] r3c7{n1 n4} - c9n4{r1 r7} - r9c7{n4 n9} - r9c8{n9 n3} - r9c2{n3
n6} - r7n6{c3 c5} - r7n8{c5 c6} - r7n9{c6 c3} - r1n9{c3 c2} - r6c2{n9 n3} -
r4c2{n3 n1} - r5n1{c3 c9} - r5n3{c9 c6} - r5c3{n3 n6} - r6c3{n6 n4} - c1n4{r6 r2}
- r2c6{n4 n2} - r2c7{n2 n6} - r2c9{n6 n7} - r1c9{n7 .}
```

FYI: Denis, my solver agrees with Mauricio's

Allan

Back to top profile pm www

denis_berthier Posted: Mon Jan 18, 2010 7:42 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

Mauricio, Allan
It is not impossible that there is a bug: so long chains are very rare and the corresponding rules are not as well tested as rules for shorter chains.
I've therefore checked my code but not yet found any bug.
One possibility, a problem that I had met with the previous version of Clips when I reached the memory limits, is that some rules that should fire don't. With the new version of Clips, it seemed not to happen but I never reach the memory limits for randomly generated puzzles. With this puzzle, I do.
I'll investigate again later.

Do you get your whips with the standard no loop whips?

Can you try this puzzle:
...3..5...5..1..3...7..4..12.....4...6..9.....1..6..28..7..2...9..8..5...5..9..7
with whips?

Back to top profile pm www

Mauricio Posted: Mon Jan 18, 2010 6:02 pm Post subject: quote

Joined: 22 Mar 2006
Posts: 1101

denis_berthier wrote:

> Do you get your whips with the standard no loop whips?
>
> Can you try this puzzle:

> ...3..5...5..1..3...7..4..12.....4...6..9.....1..6..28..7..2...9..8..5...5..9..7
> with whips?

Standard, no loops.

Code:

```
000300500050010030007004001200000400060090000001006002800700200090080050005009007
r8c1=7
r6c7<>7, whip[2] r2n7{c7 c6} - r5n7{c6 .}
r1c3<>2, whip[3] c2n2{r1 r9} - c5n2{r9 r3} - r2n2{c6 .}
r1c6<>2, whip[3] r2n2{c4 c3} - r8n2{c3 c4} - r5n2{c4 .}
r2c3<>9, whip[3] c1n9{r1 r6} - c7n9{r6 r3} - r1n9{c9 .}
r3c4<>2, whip[3] r2n2{c4 c3} - r8n2{c3 c6} - r5n2{c6 .}
r4c9<>9, whip[3] r6n9{c7 c1} - c1n5{r6 r5} - c9n5{r5 .}
r9c4<>2, whip[3] r5n2{c4 c6} - r2n2{c6 c3} - r8n2{c3 .}
r4c8<>9, whip[4] r4n6{c8 c9} - c9n5{r4 r5} - c1n5{r5 r6} - r6n9{c1 .}
r4c3=9
r2c9<>9, whip[7] r7n9{c9 c8} - r1n9{c8 c1} - r1n1{c1 c2} - r7n1{c2 c6} - r7n5{c6
c5} - r3n5{c5 c4} - r3n9{c4 .}
r2c7<>9, whip[9] c7n7{r2 r5} - c8n7{r4 r1} - r1n9{c8 c1} - c1n1{r1 r9} - c7n1{r9
r8} - r7n1{c8 c6} - r7n5{c6 c5} - r3n5{c5 c4} - r3n9{c4 .}
r2c7<>8, whip[11] b3n7{r2c7 r1c8} - r1c6{n7 n8} - c3n8{r1 r5} - c9n8{r5 r4} -
c8n8{r4 r9} - c8n4{r9 r7} - r7n9{c8 c9} - r1n9{c9 c1} - r1n1{c1 c2} - r7c2{n1 n3}
- c3n3{r8 .}
r1c8<>8, whip[18] r1c6{n8 n7} - r2n7{c6 c7} - r5n7{c7 c8} - r6c8{n7 n9} - r7n9{c8
c9} - r1n9{c9 c1} - r1n1{c1 c2} - r1n2{c2 c5} - r2c6{n2 n8} - c3n8{r2 r5} -
c9n8{r5 r4} - r6c7{n8 n3} - r5c7{n3 n1} - r8c7{n1 n6} - b9n3{r8c7 r8c9} - c3n3{r8
r7} - b7n6{r7c3 r9c1} - c1n1{r9 .}

No more steps
```

While you study your bug, this example would be of interest, solved by you here

Code:

```
001002000020030000400500600006000007010000080900000500007008003030090010500700400
r8c7=7
r9c6=3
r9c5=1
r7c1=1
r1c9<>8, whip[1] c7n8{r2 .}
r2c9<>8, whip[1] c7n8{r1 .}
r3c9<>8, whip[1] c7n8{r2 .}
r4c6<>1, whip[2] r3n1{c6 c9} - r6n1{c9 .}
r5c4<>2, whip[11] c5n2{r4 r7} - r7c7{n2 n9} - r5c7{n9 n3} - r1c7{n3 n8} - r2c7{n8
n1} - r4n1{c7 c4} - r4n3{c4 c1} - r6n3{c3 c4} - c4n8{r6 r2} - c1n8{r2 r8} -
c1n2{r8 .}
r9c9<>9, whip[14] r7c7{n9 n2} - r9c8{n2 n6} - r9c2{n6 n8} - b7n9{r9c2 r7c2} -
c2n6{r7 r1} - b1n5{r1c2 r2c3} - c3n9{r2 r3} - r3n3{c3 c8} - r1n3{c7 c1} -
b1n8{r1c1 r2c1} - r3n8{c2 c5} - r4n8{c5 c4} - r4n1{c4 c7} - r4n3{c7 .}
r4c7<>9, whip[16] r7c7{n9 n2} - r5c7{n2 n3} - r1c7{n3 n8} - r2c7{n8 n1} - r4n1{c7
c4} - r4n3{c4 c1} - c3n3{r5 r3} - r6n3{c3 c4} - c4n8{r6 r2} - r3n8{c5 c2} -
r4n8{c2 c5} - r4n2{c5 c8} - r3n2{c8 c9} - r3n1{c9 c6} - r3n9{c6 c8} - c9n9{r2 .}
r2c7<>9, whip[18] r7c7{n9 n2} - r9n2{c8 c3} - c3n9{r9 r3} - r3n3{c3 c8} - r3n2{c8
c9} - r3n1{c9 c6} - b2n9{r3c6 r1c4} - c9n9{r1 r5} - r5c7{n9 n3} - r4c7{n3 n1} -
r6n1{c9 c4} - c4n3{r6 r4} - c4n8{r4 r2} - r2c3{n8 n5} - r5c3{n5 n4} - r5c4{n4 n6}
- r7c4{n6 n4} - r8n4{c6 .}
r2c1<>8, whip[11] r3n8{c2 c5} - r1n8{c4 c7} - r2c7{n8 n1} - r4n1{c7 c4} - r4n8{c4
c2} - r6n8{c2 c4} - c4n3{r6 r5} - c7n3{r5 r4} - c1n3{r4 r1} - b1n6{r1c1 r1c2} -
c2n5{r1 .}
r3c6<>7, whip[18] r3c5{n7 n8} - r3c2{n8 n9} - r3c3{n9 n3} - r3c8{n3 n2} - r3c9{n2
n1} - r2c7{n1 n8} - r2c3{n8 n5} - c3n9{r2 r9} - r9n2{c3 c9} - r9n8{c9 c2} -
r8n8{c1 c9} - r8n5{c9 c6} - r5n5{c6 c5} - r5n7{c5 c1} - r6c2{n7 n4} - r5c3{n4 n2}
- c7n2{r5 r4} - c7n1{r4 .}
r4c7<>2, whip[23] r7c7{n2 n9} - r5c7{n9 n3} - r1c7{n3 n8} - r2c7{n8 n1} - r4n1{c7
c4} - r4n3{c4 c1} - r1n3{c1 c8} - r3n3{c8 c3} - r6n3{c3 c4} - c4n8{r6 r2} -
b1n8{r2c3 r3c2} - r4n8{c2 c5} - r3c5{n8 n7} - c8n7{r3 r2} - c8n5{r2 r7} - c5n5{r7
r5} - c3n5{r5 r2} - c3n9{r2 r9} - r9c2{n9 n6} - r9c8{n6 n2} - r3c8{n2 n9} -
r2n9{c8 c6} - r4n9{c6 .}
r5c3<>2, whip[2] c7n2{r5 r7} - r9n2{c9 .}
r8c9<>2, whip[17] c7n2{r7 r5} - c1n2{r5 r4} - c3n2{r6 r9} - c8n2{r9 r3} - r3n3{c8
c3} - c3n9{r3 r2} - c3n5{r2 r5} - b4n3{r5c3 r5c1} - b4n7{r5c1 r6c2} - r3c2{n7 n8}
- r4c2{n8 n4} - c2n5{r4 r1} - c9n5{r1 r2} - r8n5{c9 c6} - r4c6{n5 n9} - r3n9{c6
c9} - r5n9{c9 .}
No more steps
```

Back to top profile pm

Mauricio

Joined: 22 Mar 2006
Posts: 1101

Posted: Tue Jan 19, 2010 6:42 am Post subject: quote

denis_berthier wrote:

> But I'm wondering: as whips are much easier to code than braids, couldn't you easily look for them before braids of same length?
> As I said once to Paul:
> - it makes nicer solutions (whips are nicer than braids)
> - in SudoRules, it makes computation times shorter (but that wouldn't be necessarily the case in your approach).

Which set me thinking, why are nrczt whips nicer than nrczt braids? Is it perhaps that the candidate Ln is nrc linked to Rn-1, and so it must be eliminated when we assert Rn-1 true? The proof for whips, that its target can be eliminated, starts assuming the target is true and we reach a contradiction, and something nice is the fact that Ln is eliminated when we assert Rn-1, and then Rn must be true.

I can see why you do not allow llc's reuse, since if we follow the proof, the candidate Ln is already eliminated (assuming it is a repeated Lm and m<n), because we needed to eliminate it to prove Rm.

Something that annoyed me was that if Ln is a t or z candidate of a previous Lm Rm pair, we needed to eliminate it to prove Rm, and when we reached the pair Rn-1 Ln, the candidate Ln was already eliminated.

So, if we restrict too that a Ln candidate is not a previous llc nor a previous t or z candidate, we can follow the proof of the whip easily, since the candidate Ln is eliminated only after Rn-1 is assumed true, we do not need to eliminate Ln before Rn-1 is reached.

Let us call those kind of whips RWhips (r for restricted). Let us see a resolution path of a puzzle solved with standard whips here

Code:

```
100050009000700030870000004008000000530090600000302400060004800005900310001000000
r4c9=3
r2c2=5
r9c8=4
r7c3<>3, RWhip[1] c1n3{r9 .}
r4c8<>5, RWhip[1] r6n5{c9 .}
r4c7<>5, RWhip[1] r6n5{c9 .}
r2c9<>6, RWhip[1] c8n6{r3 .}
r4c1<>9, RWhip[2] c2n9{r4 r9} - c7n9{r9 .}
r2c3<>4, RWhip[2] r1n4{c2 c4} - r5n4{c4 .}
r2c5<>1, RWhip[5] r3n1{c4 c7} - r3n5{c7 c8} - r6n5{c8 c9} - r7n5{c9 c4} - r7n1{c4
.}
r3c6<>6, RWhip[19] r3n9{c6 c3} - r3n3{c3 c5} - r1c6{n3 n8} - r8c6{n8 n7} -
r5c6{n7 n1} - r4c6{n1 n5} - r9c6{n5 n3} - c1n3{r9 r7} - r7n9{c1 c8} - c7n9{r9 r4}
- r4n1{c7 c2} - r6c2{n1 n9} - r9n9{c2 c1} - b7n7{r9c1 r7c3} - r6c3{n7 n6} -
r2c3{n6 n2} - b4n2{r5c3 r4c1} - r4c8{n2 n7} - r5n7{c9 .}
r2c1<>2, RWhip[19] r1c2{n2 n4} - r8n4{c2 c1} - b4n4{r4c1 r5c3} - b4n2{r5c3 r4c2}
- r8c2{n2 n8} - r9c2{n8 n9} - r7n9{c1 c8} - r4c8{n9 n7} - r4c1{n7 n6} - r6n6{c1
c5} - b5n7{r6c5 r5c6} - b5n8{r5c6 r5c4} - r5n1{c4 c9} - r2c9{n1 n8} - r2c5{n8 n4}
- r4c5{n4 n1} - r7n1{c5 c4} - r7n5{c4 c9} - r6c9{n5 .}
r4c7<>2, RWhip[16] r5n2{c8 c3} - r5n4{c3 c4} - c5n4{r4 r2} - r2n2{c5 c9} -
r2n8{c9 c6} - r1n8{c4 c8} - r5c8{n8 n7} - r4c8{n7 n9} - b9n9{r7c8 r9c7} - c2n9{r9
r6} - c2n1{r6 r4} - r4n4{c2 c1} - r8n4{c1 c2} - r1c2{n4 n2} - r9c2{n2 n8} -
c4n8{r9 .}
r7c8<>7, RWhip[8] b9n9{r7c8 r9c7} - c7n5{r9 r3} - c8n5{r3 r6} - c8n9{r6 r4} -
c2n9{r4 r6} - c2n1{r6 r4} - r4c7{n1 n7} - r1n7{c7 .}
r6c9<>7, RWhip[7] c8n7{r4 r1} - r1c7{n7 n2} - r2c7{n2 n1} - r4c7{n1 n9} - r4c8{n9
n2} - r5n2{c8 c3} - b1n2{r3c3 .}
r9c7<>7, RWhip[9] c9n7{r7 r5} - c8n7{r4 r1} - b3n8{r1c8 r2c9} - c9n1{r2 r6} -
r4c7{n1 n9} - r4c8{n9 n2} - r5n2{c8 c3} - b1n2{r1c3 r1c2} - r1c7{n2 .}
r5c9<>7, RWhip[1] b9n7{r9c9 .}
r3c8<>2, RWhip[12] c7n2{r1 r9} - c7n9{r9 r4} - r4c8{n9 n7} - r5c8{n7 n8} -
r6c8{n8 n5} - r6c9{n5 n1} - r6c2{n1 n9} - r9c2{n9 n8} - c4n8{r9 r1} - c4n2{r1 r7}
- c5n2{r7 r2} - b2n4{r2c5 .}
r2c5<>2, RWhip[14] c5n4{r2 r4} - c4n4{r4 r1} - r1c2{n4 n2} - r3n2{c3 c7} -
r3n5{c7 c8} - r6n5{c8 c9} - r7n5{c9 c4} - r7n1{c4 c5} - r6n1{c5 c2} - r4c2{n1 n9}
- r9c2{n9 n8} - c4n8{r9 r5} - r6n8{c5 c8} - r6n9{c8 .}
r7c8<>2, RWhip[16] b9n9{r7c8 r9c7} - c7n5{r9 r3} - c8n5{r3 r6} - c8n9{r6 r4} -
c2n9{r4 r6} - c2n1{r6 r4} - r4c7{n1 n7} - r5c8{n7 n8} - r6n8{c9 c5} - b5n7{r6c5
r5c6} - b5n1{r5c6 r5c4} - r7n1{c4 c5} - r3n1{c5 c6} - r3n9{c6 c3} - r7n9{c3 c1} -
```

```
r7n3{c1 .}
r5c9<>2, RWhip[16] c8n2{r4 r1} - c7n2{r1 r9} - c7n9{r9 r4} - r4c8{n9 n7} -
r5c8{n7 n8} - r6c8{n8 n5} - r6c9{n5 n1} - c2n1{r6 r4} - c2n2{r4 r8} - r8n4{c2 c1}
- b4n4{r4c1 r5c3} - r5c4{n4 n1} - r7n1{c4 c5} - r7n2{c5 c4} - r3c4{n2 n6} -
r3c8{n6 .}
r1c8<>2, RWhip[1] b6n2{r5c8 .}
r5c4<>1, RWhip[5] r5c9{n1 n8} - r6n8{c8 c5} - r2n8{c5 c6} - c6n1{r2 r3} - c6n9{r3
.}
r6c9<>1, RWhip[10] r5c9{n1 n8} - r5c4{n8 n4} - c3n4{r5 r1} - r2n4{c1 c5} -
r2n8{c5 c6} - c4n8{r1 r9} - r8n8{c5 c2} - r8n4{c2 c1} - r4n4{c1 c2} - c2n1{r4 .}
r3c5<>6, RWhip[10] r3c8{n6 n5} - r6n5{c8 c9} - r7n5{c9 c4} - r4n5{c4 c6} -
b5n6{r4c6 r4c4} - c4n1{r4 r3} - r3c7{n1 n2} - r2c7{n2 n1} - c9n1{r2 r5} - c6n1{r5
.}
r2c9<>1, RWhip[10] r5c9{n1 n8} - r5c4{n8 n4} - c3n4{r5 r1} - r1c2{n4 n2} -
r2n2{c3 c7} - r3c7{n2 n5} - r9c7{n5 n9} - r9c2{n9 n8} - c4n8{r9 r1} - r2n8{c6 .}
r5c9=1
r9c4<>8, RWhip[7] r5c4{n8 n4} - c3n4{r5 r1} - r1n3{c3 c6} - r1n8{c6 c8} - r5n8{c8
c6} - r2n8{c6 c5} - r2n4{c5 .}
r6c8<>9, RWhip[7] r4c7{n9 n7} - r1c7{n7 n2} - r1c2{n2 n4} - r8n4{c2 c1} -
b4n4{r4c1 r5c3} - r5n2{c3 c8} - r4c8{n2 .}
r4c2<>9, RWhip[1] r6n9{c3 .}
r4c6<>7, RWhip[7] r5c6{n7 n8} - r8c6{n8 n6} - r1c6{n6 n3} - r9c6{n3 n5} - r9c4{n5
n2} - r9c7{n2 n9} - r4c7{n9 .}
r7c4<>2, RWhip[8] r7n1{c4 c5} - r6n1{c5 c2} - c2n9{r6 r9} - r7c3{n9 n7} - r7c1{n7
n3} - r9c1{n3 n2} - r9c7{n2 n5} - r7c9{n5 .}
r3c4<>1, RWhip[3] r7c4{n1 n5} - r4n5{c4 c6} - c6n1{r4 .}
r3c3<>2, RWhip[5] r3c4{n2 n6} - r3c8{n6 n5} - c7n5{r3 r9} - r9c4{n5 n2} - c5n2{r8
.}
r1c4<>2, RWhip[5] r3c4{n2 n6} - r3c8{n6 n5} - r6n5{c8 c9} - r7n5{c9 c4} - r9c4{n5
.}
r3c7<>2, RWhip[1] b2n2{r3c5 .}
r7c3<>9, RWhip[7] r3n9{c3 c6} - r2n9{c6 c1} - r6n9{c1 c2} - r6n1{c2 c5} - r3n1{c5
c7} - r3n5{c7 c8} - r7c8{n5 .}
r4c6<>6, RWhip[8] r4n5{c6 c4} - b8n5{r7c4 r9c6} - c7n5{r9 r3} - c7n1{r3 r2} -
c6n1{r2 r3} - r3n9{c6 c3} - r3n3{c3 c5} - c6n3{r1 .}
r7c8<>5, RWhip[8] r7n9{c8 c1} - c2n9{r9 r6} - r6n1{c2 c5} - r4c6{n1 n5} - r9n5{c6
c4} - c4n2{r9 r3} - r3c5{n2 n3} - r7n3{c5 .}
r7c8=9
r4c7=9
r1c7=7
r2c3<>2, RWhip[1] r1n2{c2 .}
r9c2<>2, RWhip[6] r1c2{n2 n4} - r4c2{n4 n1} - r6n1{c2 c5} - r7n1{c5 c4} - r7n5{c4
c9} - r9c7{n5 .}
r2c6<>8, RWhip[6] r5c6{n8 n7} - r8c6{n7 n6} - r1c6{n6 n3} - r9c6{n3 n5} - r9c4{n5
n2} - r9c7{n2 .}
r1c4<>6, RWhip[3] r1c8{n6 n8} - r2n8{c9 c5} - b2n4{r2c5 .}
r4c4<>4, RWhip[2] r1c4{n4 n8} - r5c4{n8 .}
r8c2<>2, RWhip[3] r1c2{n2 n4} - c3n4{r1 r5} - c4n4{r5 .}
r4c2<>2, RWhip[3] r1c2{n2 n4} - c3n4{r1 r5} - c4n4{r5 .}
r1c2=2
r4c6<>1, RWhip[5] r4c2{n1 n4} - c5n4{r4 r2} - r2n8{c5 c9} - r2n2{c9 c7} - r2n1{c7
.}
r4c6=5
r3c5<>1, RWhip[1] c6n1{r2 .}
r3c6<>3, RWhip[3] r3n1{c6 c7} - r2n1{c7 c6} - c6n9{r2 .}
r2c6<>6, RWhip[3] r2c3{n6 n9} - r3n9{c3 c6} - c6n1{r3 .}
r1c3<>6, RWhip[3] r1c8{n6 n8} - r2n8{c9 c5} - r2n6{c5 .}
r7c5<>3, RWhip[4] r3c5{n3 n2} - r3c4{n2 n6} - r4c4{n6 n1} - r7n1{c4 .}
r7c1=3
r9c9<>2, RWhip[5] r9c7{n2 n5} - r9c4{n5 n6} - c6n6{r8 r1} - r1c8{n6 n8} - r2c9{n8
.}
r9c5<>8, RWhip[5] r8n8{c5 c2} - r8n4{c2 c1} - b1n4{r2c1 r1c3} - r1n3{c3 c6} -
r9n3{c6 .}
r9c5<>2, RWhip[5] r9c7{n2 n5} - r9c4{n5 n6} - c6n6{r8 r1} - c8n6{r1 r3} - r3n5{c8
.}
r9c4<>6, RWhip[5] r4c4{n6 n1} - r4c2{n1 n4} - c3n4{r5 r1} - r1n3{c3 c6} - c6n6{r1
.}
r9c9<>5, RWhip[2] r9c4{n5 n2} - r9c7{n2 .}
r9c1<>2, RWhip[2] r9c4{n2 n5} - r9c7{n5 .}
r7c5<>2, RWhip[4] r3c5{n2 n3} - r1n3{c6 c3} - c3n4{r1 r5} - c3n2{r5 .}
r8c5<>6, RWhip[3] c5n2{r8 r3} - r3c4{n2 n6} - c6n6{r1 .}
r9c6<>7, RWhip[4] r7c5{n7 n1} - r6n1{c5 c2} - c2n9{r6 r9} - r9c1{n9 .}
r9c5<>7, RWhip[4] r7c5{n7 n1} - r6n1{c5 c2} - c2n9{r6 r9} - r9c1{n9 .}
r3c3<>6, RWhip[3] r2n6{c1 c5} - r9c5{n6 n3} - r3n3{c5 .}
r2c5<>6, RWhip[1] b1n6{r2c3 .}
r1c6<>8, RWhip[2] r1c4{n8 n4} - r2c5{n4 .}
r7c3<>7, RWhip[4] r7c5{n7 n1} - r6n1{c5 c2} - c2n9{r6 r9} - r9c1{n9 .}
r7c3=2
r5c8=2
r4c8=7
```

```
r4c1=2
r6c5<>8, RWhip[1] r5n8{c6 .}
r6c5<>6, RWhip[1] r4n6{c4 .}
r6c1<>7, RWhip[1] c3n7{r5 .}
r8c5<>7, RWhip[2] r6c5{n7 n1} - r7c5{n1 .}
r4c5<>1, RWhip[2] r6c5{n1 n7} - r7c5{n7 .}
r9c6<>8, RWhip[3] r8c5{n8 n2} - r3c5{n2 n3} - r9n3{c5 .}
Singles
Most difficult rule: NRCZT RWhip[19]
```

I'd say RWhips are nicer than standar whips.

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

Posted: Tue Jan 19, 2010 7:33 am Post subject: quote edit

> **Mauricio wrote:**
>
> why are nrczt whips nicer than nrczt braids? Is it perhaps that the candidate Ln is nrc linked to Rn-1

Not "perhaps". It is exactly this. This is nrc-continuity, which makes a whip a chain, whereas a braid is a DAG, i.e. a net.
It may make little difference for a programmer, but it is an essential difference in structure and acceptability for a player.

> **Mauricio wrote:**
>
> and so it must be eliminated when we assert Rn-1 true

This is a completely false interpretation of whips or braids, based on a T&E-ish programmer's vision.
Nothing is "eliminated" (except, at the end, the target).
The whole whip is built on the current grid with absolutely no modification of its values and candidates content until it is completed.

> **Mauricio wrote:**
>
> I can see why you do not allow llc's reuse, since if we follow the proof, the candidate Ln is already eliminated

False. It may be marked as false in the context of the current whip, but it isn't eliminated in the current grid.

The only reasons there are no loops in standard whips are:
- in many chains, they can be proven to be useless (all the chains without the t-extension);
- nicer,
- efficiency.
But this is not an essential condition of whips.
What limits the effect of this no-loop condition is that, most of the time, if a previous llc could have been taken as Ln, Ln can be replaced by another candidate to give an equivalent whip.

I never disallowed using a previous t- or z- candidate as an llc in a later step.
Doing this can in no way be justified (your arguments are logically invalid) and it would considerably restrict the power of whips, as your example shows: you need a whip(19) instead of a whip(16).

BTW, did you find any discrepancy between my solution and yours with standard whips?

Last edited by denis_berthier on Tue Jan 19, 2010 7:37 am; edited 1 time in total

Back to top profile pm www

Allan Barker

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Posted: Tue Jan 19, 2010 7:36 am Post subject: quote

> **Mauricio wrote:**
>
> Something that annoyed me was that if Ln is a t or z candidate of a previous Lm Rm pair, we needed to eliminate it to prove Rm, and when we reached the pair Rn-1 Ln, the candidate Ln was already eliminated.

> So, if we restrict too that a Ln candidate is not a previous llc nor a previous t or z candidate, we can follow the proof of the whip easily, since the candidate Ln is eliminated only after Rn-1 is assumed true, we do not need to eliminate Ln before Rn-1 is reached.
>
> Let us call those kind of whips RWhips (r for restricted).

Mauricio,

I fully agree that this restriction eventually leads to "nicer" logic. If you look at the **Ribbons Thread** you will see my rule number 4 for ribbons, which in effect implements the same restriction. My rule is a little stricter, in that no candidates in the new truth (containing Ln) may appear as prior candidates. Have you tried that?

Ribbon Rule 4. Do not allow overlap truths, i.e., no reuse of candidates in prior truths.

.

Back to top profile pm www

denis_berthier

Posted: Tue Jan 19, 2010 7:43 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

> **Allan Barker wrote:**
>
> my rule number 4 for ribbons, which in effect implements the same restriction. My rule is a little stricter, in that no candidates in the new truth (containing Ln) may appear as prior candidates. Do not allow overlap truths, i.e., no reuse of candidates in prior truths.

You do not allow "overlapping truths" because they raise a problem in the context of cover sets but such a restriction is totally arbitrary in the context of whips.

Back to top profile pm www

Mauricio

Posted: Tue Jan 19, 2010 7:49 am Post subject: quote

Joined: 22 Mar 2006
Posts: 1101

> **denis_berthier wrote:**
>
> > **Mauricio wrote:**
> >
> > and so it must be eliminated when we assert Rn-1 true
>
> This is a completely false interpretation of whips or braids, based on a T&E-ish programmer's vision.
> Nothing is "eliminated" (except, at the end, the target).
> The whole whip is built on the current grid with absolutely no modification of its values and candidates content until it is completed.

I was obvoiusly referring to the proof that given a whip, the target can be eliminated, ow how do you prove that if the target was correct, then all Rn candidates are true, without deleting the llc's?

Back to top profile pm

Allan Barker

Posted: Tue Jan 19, 2010 7:54 am Post subject: quote

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

> **denis_berthier wrote:**
>
> The only reasons there are no loops in standard whips are:
> - in many chains, they can be proven to be useless (all the chains without the t-extension);
> - nicer

False!

Eliminating 10 candidates in one shot with loops is nice and obviously not useless. Repeating the same logic 10 times seems pretty useless to me (IMO).

Further, these multiple eliminations can be found in a natural way in look-back chains and ribbons. (TBA)

.

Last edited by Allan Barker on Tue Jan 19, 2010 8:00 am; edited 1 time in total

Back to top profile pm www

denis_berthier

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

Posted: Tue Jan 19, 2010 7:58 am Post subject: quote edit

Mauricio wrote:

> **denis_berthier wrote:**
>
> > **Mauricio wrote:**
> >
> > and so it must be eliminated when we assert Rn-1 true
>
> This is a completely false interpretation of whips or braids, based on a T&E-ish programmer's vision.
> Nothing is "eliminated" (except, at the end, the target).
> The whole whip is built on the current grid with absolutely no modification of its values and candidates content until it is completed.
>
> I was obvoiusly referring to the proof that given a whip, the target can be eliminated, ow how do you prove that if the target was correct, then all Rn candidates are true, without deleting the llc's?

A logical proof doesn't delete anything on the grid. It just records the fact that, if Z was true, then something would true or false.

Back to top profile pm www

Allan Barker

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Posted: Tue Jan 19, 2010 7:59 am Post subject: quote

denis_berthier wrote:

> **Allan Barker wrote:**
>
> my rule number 4 for ribbons, which in effect implements the same restriction. My rule is a little stricter, in that no candidates in the new truth (containing Ln) may appear as prior candidates.
> Do not allow overlap truths, i.e., no reuse of candidates in prior truths.
>
> You do not allow "overlapping truths" because they raise a problem in the context of cover sets but such a restriction is totally arbitrary in the context of whips.

False!

My ribbon solver, just like my nrczt solver has a switch to allow overlapping truths, in fact it is the same code.

My constraint solver (not availible in Xsudo) works the same, it can find any broken wing type logic, as I have posted 2 years ago.

Please check your facts before posting, or ask me in a PM.
.

Back to top profile pm www

denis_berthier

Joined: 19 Jun 2007
Posts: 1187
Location: Paris, France

Posted: Tue Jan 19, 2010 8:01 am Post subject: quote edit

Allan Barker wrote:

> **denis_berthier wrote:**
>
> The only reasons there are no loops in standard whips are:
> - in many chains, they can be proven to be useless (all the chains without the t-extension);
> - nicer
>
> Eliminating 10 candidates in one shot with loops is nice and obviously not useless. Repeating

> the same logic 10 times seems pretty useless to me (IMO).

All these eliminations are obtained by circulating along the chain. The rest is only a matter of presentation.
But allowing loops in these chains is useless from a logical POV, in that it doesn't lead to more eliminations.

Back to top profile pm www

Display posts from previous: All Posts Oldest First Go

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

All times are GMT + 1 Hour
Goto page Previous 1, 2, 3 ... , 19, 20, 21 Next

Page 20 of 21

Stop watching this topic

Jump to: Advanced solving techniques Go