Sudoku Players' Forums

Abominable TRIAL-and-ERROR and lovely BRAIDS

Goto page Previous 1, 2, 3, 4, 5, 6, 7, **8**, 9 Next

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

View previous topic :: View next topic

Author | **Message**

David P Bird

Joined: 17 Sep 2008
Posts: 206
Location: Middle England

Posted: Sun Nov 23, 2008 10:52 am Post subject: quote

Coloin, I'm not sure of what you mean about locating this puzzle's weak spot! Although its bilocal & bivalue counts are low, it has a useful set of ALS/AHT pairs which provide sufficient routes to carry inferences around the grid. This is what makes it susceptible to my restricted method set. However, I don't think a continued discussion along such lines would be very welcome on this thread.

Back to top profile pm

coloin

Joined: 06 May 2005
Posts: 1111
Location: Devon UK

Posted: Sun Nov 23, 2008 12:04 pm Post subject: quote

I agree, but we have both shown the same insertion by different means.

Your right, a puzzle's weak spot is an intriqing concept - a bit like the most "elegant" move ! I am sure **denis** agrees with this concept, however indefinable !

My slant is the relative contribution to a puzzle of all the clues. My methods are certainly "Abominable" :(

In a minimal puzzle of course all the clues are necessary but in an easy puzzle you get eliminations with just a few clues.

In a very hard puzzle mabe one clue is not quite as necesasary as the rest. This is the case here. The lack of the 4@r7c9 is still only compatible with the inserted clue value of the 8@r8c2. There will be many ways to show this [as you did] because it is true ! Whatever way you do it it does not require "information" from the 4@r7c9.

C

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

Posted: Sun Nov 23, 2008 6:37 pm Post subject: quote edit

> **coloin wrote:**
> Your right, a puzzle's weak spot is an intriqing concept - a bit like the most "elegant" move ! I am sure **denis** agrees with this concept, however indefinable !

I agree with both the idea of "elegance" and the "undefinable".

> **coloin wrote:**
> My slant is the relative contribution to a puzzle of all the clues. My methods are certainly "Abominable" :(

Abominable from one POV (e.g. T&E) maybe more acceptable from another (braids).

If we add the clue n8r8c2, SER collapses to 9.2 (instead of 9.5) and the puzzle can be completely solved by

whips of length <= 12.

Back to top profile pm www

denis_berthier

Posted: Thu Dec 04, 2008 12:45 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

About the T&E Theorems

Question from a French friend (my translation) wrote:

> Isn't there a contradiction between the two theorems you proved in relation with T&E:
>
> Taken from the "concept of a resolution rule" thread, second post
> http://www.sudoku.com/boards/viewtopic.php?t=5600:
>
> > **denis_berthier wrote:**
> >
> > Definition: T&E is the following procedure: recursively make ad hoc hypotheses on something that you do not know to be true at the time you make them, explore their consequences, with the possibility of retracting each of these hypotheses and their consequences if they lead to a dead end.
> >
> > Theorem: Trial and Error is a resolution technique that cannot be the implementation of any resolution rule.
>
> Taken from the "abominable T&E and lovely braids" thread, second post
> http://www.sudoku.com/boards/viewtopic.php?t=6390:
>
> > **denis_berthier wrote:**
> >
> > T&E theorem: ANY ELIMINATION DONE BY T&E CAN BE DONE BY AN NRCZT-BRAID.
> > ...
> > Corollary: ANY PUZZLE SOLVABLE BY T&E IS SOLVABLE BY NRCZT-BRAIDS.

Thanks for asking. I wans't aware that this wasn't obvious. The two procedures are different.
There may seem to be a contradiction, due to the very bad name I chose in the first case for the procedure I was defining.
But, the two theorems are true and non contradictory. We must just be careful about names.

The two definitions and associated theorems are recalled here (only the name of the first procedure is changed, the definitions and proofs given in the two aforementioned threads remain unchanged):

1) "Concept of a resolution rule" thread:
What I've called T&E and then rT&E in the first thread, and which should be called something like RTEG "recursive T&E with guessing" is any structured search procedure (DFS or BFS) with no a priori depth limit, and accepting positive results (if it finds a solution, it accepts it - which is a form of guessing).
RTEG theorem: RTEG is a resolution technique that cannot be the implementation of any resolution rule.
RTEG theorem (strong form): RTEG is a resolution technique that cannot be the implementation of any set of resolution rules.

This can be extended to any RTEG(T), for any resolution theory T, where RTEG(T) means the search is pruned by the rules in T.

2) "Abominable T&E and lovely braids" thread:

Definition: Given a resolution theory T (i.e. a set of resolution rules) and a candidate z, T&E(T, z) is the following resolution technique:
- start an auxiliary grid by copying the current PM; in this auxiliary grid, delete z as a candidate and add it as a decided value; apply to this auxiliary PM all the rules in T until quiescence;

- if a contradiction is thus obtained in this auxiliary PM, then eliminate z from the original one; if no contradiction is obtained, then merely go back to the original PM.

I think this is what everyone means by "using T&E(T) to eliminate candidate z".

Definition: Given a resolution theory T, T&E(T) is the following resolution technique (this is only conceptual, any implementation should optimise this):
a) define phase = 1;
b) apply all the rules in T;
c) set changed-PM = false; mark all the remaining candidates as "not tried";
d) if there is at least one "not-tried" candidate, then choose any of them, say z; apply T&E(T, z); if it eliminates z, then set changed-PM = true and apply all the rules in T, otherwise un-mark z; in both cases, goto d;
e) if there is no not-tried candidate:
---------- if changed-PM = true, then set phase = phase +1 and goto c);
---------- if changed-PM = false, then stop.
Said otherwise, repeat T&E(T, z) for all the candidates z, as long as any one remains, until quiescence.

T&E theorem: Any elimination done by T&E can be done by an nrczt-braid.
T&E theorem: Any elimination done by T&E(T) can be done by an nrczt-braid(T).

3) Differences between the two procedures:
In the second case:
- there's no recursion (although we can also define a depth 2 T&E, but it won't be unlimited recursion).
- no "guessing" is allowed: if a solution is found in the auxiliary PM, it is merely forgotten.

Back to top profile pm www

denis_berthier

Posted: Sat Nov 07, 2009 6:12 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

Looking backwards at this thread, I realise something is missing: I've never given here resolution paths using braids. (I've given a few ones in the fully supersymmetric thread, most of them before their final implementation in SudoRules, but not here).

In the following examples, as usual in SudoRules, whips or braids of the shortest length are always preferred. Also, whips are preferred to braids of the same length.
Remember what I said in the "rating" thread: in such conditions, braids are very rare.
As braids are used, the level indicated at the end of the solution paths is the B-NRCZT level. (Remember that the B-NRCZT rating is equal to the NRCZT rating - except in ~ 45 cases in 1,000,000)

These examples are taken from the suexg-cb collection of 568,705 controlled-bias minimal puzzles.
The type of the braid (-rc, -rn, -cn, -bn) follows the same convention as for whips: it indicates in which type of cell the final contradiction is reached.

First example, in which the most complex rule is not a braid:

Code:

```
(solve
"..5..7......5621.........9.24....9..5.6...7.4.8.45..1..3..4..8...962.........1...")
*****  SudoRules version 13.7wbis-B2  *****
..5..7......5621.........9.24....9..5.6...7.4.8.45..1..3..4..8...962.........1...
singles ==> r3c6 = 4, r5c4 = 2, r5c8 = 3, r4c9 = 8, r4c8 = 5, r4c6 = 6
interaction row r6 with block b4 for number 7 ==> r4c3 <> 7
interaction row r2 with block b1 for number 8 ==> r3c3 <> 8, r3c1 <> 8, r1c1 <> 8
interaction row r2 with block b1 for number 9 ==> r1c2 <> 9, r1c1 <> 9
naked-pairs-in-a-column {n4 n7}{r2 r8}c8 ==> r9c8 <> 7, r9c8 <> 4, r1c8 <> 4
hidden-pairs-in-a-column {n4 n8}{r2 r9}c3 ==> r9c3 <> 7, r9c3 <> 2, r2c3 <> 7, r2c3
<> 3
nrc-chain[2]  n4{r1c1 r1c7} - n4{r2c8 r8c8} ==> r8c1 <> 4
interaction row r8 with block b9 for number 4 ==> r9c7 <> 4
nrct-chain[2]  n6{r1c8 r9c8} - n6{r9c2 r7c1} ==> r1c1 <> 6
nrc-chain[3]  n7{r6c3 r6c1} - n9{r6c1 r2c1} - {n9 n7}r2c2 ==> r3c3 <> 7
nrczt-whip-rc[4]  n9{r9c9 r7c9} - {n9 n5}r7c6 - {n5 n2}r7c7 - {n2r9c8 .} ==> r9c9 <>
6
nrczt-whip-rn[4]  n3{r9c5 r8c6} - {n3 n9}r6c6 - n9{r6c1 r2c1} - {n3r2c1 .} ==> r9c9
```

```
<> 3
nrczt-whip-rc[4]  n9{r9c9 r7c9} - {n9 n5}r7c6 - {n5 n6}r7c7 - {n6r9c8 .} ==> r9c9 <>
2
nrczt-whip-rn[4]  n2{r7c3 r9c2} - {n2 n6}r9c8 - {n6 n5}r7c7 - {n5r9c9 .} ==> r7c9 <>
2
nrct-chain[3]  {n6 n2}r1c8 - n2{r1c9 r6c9} - n6{r6c9 r6c7} ==> r3c7 <> 6, r1c7 <> 6
nrczt-whip-cn[5]  {n2 n6}r6c7 - {n6 n5}r7c7 - {n5 n3}r9c7 - n3{r9c5 r8c6} - {n5r8c6
.} ==> r1c7 <> 2, r3c7 <> 2
nrct-chain[3]  {n6 n2}r9c8 - n2{r9c7 r6c7} - {n2 n6}r6c9 ==> r7c9 <> 6
nrczt-braid-rc[5]  n9{r7c9 r9c9} - n5{r9c9 r9c2} - n1{r7c9 r8c9} - n7{r7c9 r8c8} -
{n7r8c2 .} ==> r7c9 <> 5     <-------
nrczt-braid-rn[5]  n6{r7c7 r7c1} - n2{r7c7 r7c3} - n5{r9c9 r3c9} - n6{r3c1 r3c2} -
{n2r3c9 .} ==> r7c7 <> 5     <-------
hidden-single-in-a-row ==> r7c6 = 5
interaction column c6 with block b5 for number 9 ==> r5c5 <> 9
naked-pairs-in-a-block {n2 n6}{r7c7 r9c8} ==> r9c7 <> 6, r9c7 <> 2
nrct-chain[6]  {n8 n1}r5c5 - {n1 n9}r5c2 - {n9 n7}r2c2 - n7{r3c2 r3c9} - n5{r3c9
r3c7} - n8{r3c7 r1c7} ==> r1c5 <> 8
nrct-chain[6]  n5{r8c2 r9c2} - {n5 n3}r9c7 - n3{r9c5 r8c6} - n8{r8c6 r5c6} - n9{r5c6
r5c2} - {n9 n7}r2c2 ==> r8c2 <> 7
nrc-chain[5]  {n5 n1}r8c2 - {n1 n9}r5c2 - {n9 n7}r2c2 - {n7 n4}r2c8 - n4{r8c8 r8c7}
==> r8c7 <> 5
nrct-chain[6]  n5{r3c9 r3c7} - {n5 n3}r9c7 - n3{r9c5 r8c6} - {n3 n9}r6c6 - n9{r6c1
r2c1} - n3{r2c1 r2c9} ==> r3c9 <> 3
nrczt-whip-bn[6]  n7{r2c9 r3c9} - n7{r3c2 r9c2} - n2{r9c2 r7c3} - n2{r3c3 r3c2} -
n6{r3c2 r3c1} - {n6r9c1 .} ==> r2c1 <> 7
nrczt-whip-rc[6]  n4{r1c1 r1c7} - n8{r1c7 r1c4} - n9{r1c4 r1c5} - n3{r1c5 r1c9} -
{n3 n7}r2c9 - {n7r2c8 .} ==> r1c1 <> 1
nrczt-whip-bn[2]  n1{r5c5 r5c2} - {n1r1c2 .} ==> r3c5 <> 1
nrc-chain[3]  {n3 n8}r3c5 - n8{r5c5 r5c6} - {n8 n3}r8c6 ==> r9c5 <> 3
nrct-chain[3]  {n3 n8}r3c5 - n8{r3c4 r9c4} - n3{r9c4 r9c7} ==> r3c7 <> 3
nrct-chain[6]  n8{r1c7 r1c4} - n9{r1c4 r1c5} - n1{r1c5 r1c2} - {n1 n9}r5c2 - {n9
n7}r2c2 - {n7 n3}r2c9 ==> r1c7 <> 3
interaction column c7 with block b9 for number 3 ==> r8c9 <> 3
nrc-chain[5]  {n3 n4}r1c1 - n4{r1c7 r8c7} - n3{r8c7 r8c6} - {n3 n9}r6c6 - n9{r6c1
r2c1} ==> r2c1 <> 3
hidden-single-in-a-row ==> r2c9 = 3
naked-pairs-in-a-block {n2 n6}{r1c8 r1c9} ==> r3c9 <> 6
interaction row r3 with block b1 for number 6 ==> r1c2 <> 6
naked-pairs-in-a-block {n2 n6}{r1c8 r1c9} ==> r3c9 <> 2
interaction row r3 with block b1 for number 2 ==> r1c2 <> 2
naked-singles
GRID 0 SOLVED. B-NRCZT LEVEL = B-NRCZT6, MOST COMPLEX RULE = NRCZT6
415937862
978562143
623184597
241376958
596218734
387459216
132745689
859623471
764891325
```

Second example, in which the most complex rule is not a braid:

Code:

```
(solve
".9.5..8.7....2......3.49.2.1....67.......4..38..1526....5....7...8.9.2..3.....96.")
*****  SudoRules version 13.7wbis-B2  *****
.9.5..8.7....2......3.49.2.1....67.......4..38..1526....5....7...8.9.2..3.....96.
singles ==> r6c2 = 3, r6c3 = 7, r7c1 = 9, r4c9 = 2
interaction row r6 with block b6 for number 9 ==> r5c8 <> 9, r4c8 <> 9
interaction row r6 with block b6 for number 4 ==> r4c8 <> 4
naked-pairs-in-a-column {n1 n5}{r3 r5}c7 ==> r7c7 <> 1, r2c7 <> 5, r2c7 <> 1
nrc-chain[2]  n5{r3c7 r5c7} - n5{r4c8 r4c2} ==> r3c2 <> 5
nrczt-whip-rc[4]  n5{r9c6 r9c9} - n8{r9c9 r7c9} - {n8 n3}r7c6 - {n3r1c6 .} ==> r9c6
<> 1
nrczt-whip-rn[5]  n6{r7c5 r1c5} - n1{r1c5 r9c5} - {n1 n3}r7c6 - n3{r8c6 r8c8} -
{n3r1c8 .} ==> r7c5 <> 8
nrczt-whip-rn[6]  n4{r7c7 r2c7} - n4{r2c1 r1c1} - n2{r1c1 r1c3} - n6{r1c3 r1c5} -
n6{r7c5 r7c4} - {n2r7c4 .} ==> r7c2 <> 4
nrczt-braid-cn[6]  {n1 n3}r1c6 - n1{r1c5 r2c6} - n1{r1c3 r9c3} - {n1 n8}r7c6 - {n1
n7}r9c5 - {n7r9c6 .} ==> r1c8 <> 1     <-------
naked-pairs-in-a-block {n3 n4}{r1c8 r2c7} ==> r2c9 <> 4, r2c8 <> 4, r2c8 <> 3
nrct-chain[5]  {n4 n3}r7c7 - n3{r8c8 r1c8} - {n3 n1}r1c6 - {n1 n8}r7c6 - n8{r9c6
r9c9} ==> r9c9 <> 4
nrct-chain[6]  n6{r3c9 r2c9} - n9{r2c9 r2c8} - {n9 n4}r6c8 - {n4 n3}r1c8 - {n3
```

```
n1}r1c6 - {n1 n6}r1c5 ==> r3c4 <> 6
nrczt-whip-rn[6]  {n4 n3}r7c7 - n3{r8c8 r1c8} - {n3 n1}r1c6 - {n1 n6}r1c5 - n6{r7c5
r7c2} - {n2r7c2 .} ==> r7c4 <> 4
interaction row r7 with block b9 for number 4 ==> r8c9 <> 4
interaction row r7 with block b9 for number 4 ==> r8c8 <> 4
nrczt-whip-cn[6]  n3{r8c8 r1c8} - {n3 n1}r1c6 - {n1 n6}r1c5 - n6{r7c5 r7c4} -
n2{r7c4 r9c4} - {n4r9c4 .} ==> r8c4 <> 3
nrc-chain[2]  n3{r2c7 r7c7} - n3{r8c8 r8c6} ==> r2c6 <> 3
nrczt-braid-cn[6]  n4{r4c2 r4c3} - n9{r4c3 r4c4} - {n4 n3}r2c7 - n3{r4c4 r7c4} -
n4{r9c3 r9c4} - {n2r9c4 .} ==> r2c2 <> 4     <-------
nrczt-braid-rc[6]  n6{r3c9 r2c9} - n6{r2c4 r1c5} - n5{r3c7 r5c7} - n6{r2c3 r5c3} -
{n6 n2}r5c2 - {n6r5c1 .} ==> r3c9 <> 5     <-------
nrczt-whip-rn[7]  {n7 n8}r3c4 - {n8 n1}r2c6 - {n1 n3}r1c6 - n3{r8c6 r8c8} - n1{r8c8
r5c8} - n8{r5c8 r5c5} - {n7r5c5 .} ==> r2c4 <> 7
nrczt-braid-rc[6]  n6{r3c9 r2c9} - n6{r2c3 r5c3} - n9{r5c3 r5c4} - n8{r3c2 r3c4} -
{n8 n3}r4c4 - {n3r2c4 .} ==> r3c2 <> 6     <-------
nrczt-braid-cn[6]  {n7 n8}r3c4 - n7{r9c6 r9c2} - {n7 n1}r3c2 - n4{r8c4 r9c4} -
n2{r9c4 r9c3} - {n1r9c3 .} ==> r8c4 <> 7     <-------
nrczt-whip-rn[7]  n7{r2c6 r3c4} - n8{r3c4 r2c4} - n3{r2c4 r2c7} - n3{r7c7 r8c8} -
n1{r8c8 r5c8} - n8{r5c8 r5c5} - {n7r5c5 .} ==> r2c6 <> 1
interaction block b2 with row r1 for number 1 ==> r1c3 <> 1
naked-pairs-in-a-block {n7 n8}{r2c6 r3c4} ==> r2c4 <> 8
nrc-chain[3]  n6{r7c5 r1c5} - {n6 n3}r2c4 - n3{r4c4 r4c5} ==> r7c5 <> 3
nrczt-whip-cn[4]  {n4 n3}r2c7 - {n3 n6}r2c4 - {n6 n4}r8c4 - {n4r8c1 .} ==> r2c3 <> 4
nrc-chain[5]  {n1 n6}r2c3 - {n6 n3}r2c4 - n3{r2c7 r1c8} - n4{r1c8 r6c8} - n9{r6c8
r2c8} ==> r2c8 <> 1
hxy-cn-chain[4]  {r4 r5}c8n8 - {r5 r8}c8n1 - {r8 r1}c8n3 - {r1 r4}c5n3 ==> r4c5 <> 8
naked-single ==> r4c5 = 3
naked-pairs-in-a-column {n1 n6}{r1 r7}c5 ==> r9c5 <> 1
x-wing-in-rows n3{r1 r8}{c6 c8} ==> r7c6 <> 3
naked-triplets-in-a-column {n7 n8 n9}{r3 r4 r5}c4 ==> r9c4 <> 8, r9c4 <> 7, r7c4 <>
8
nrczt-whip-rc[3]  n1{r9c2 r9c9} - n8{r9c9 r7c9} - {n8r7c6 .} ==> r7c2 <> 1
nrc-chain[4]  n6{r2c4 r1c5} - {n6 n1}r7c5 - {n1 n8}r7c6 - n8{r2c6 r2c2} ==> r2c2 <>
6
nrc-chain[4]  n4{r2c1 r2c7} - n4{r7c7 r7c9} - n8{r7c9 r7c6} - {n8 n7}r2c6 ==> r2c1
<> 7
hidden-pairs-in-a-row {n7 n8}r2{c2 c6} ==> r2c2 <> 5
interaction column c2 with block b4 for number 5 ==> r5c1 <> 5
hidden-pairs-in-a-row {n7 n8}r2{c2 c6} ==> r2c2 <> 1
nrc-chain[2]  n7{r2c6 r2c2} - n7{r3c1 r8c1} ==> r8c6 <> 7
interaction row r8 with block b7 for number 7 ==> r9c2 <> 7
naked-triplets-in-a-row {n2 n1 n4}r9{c2 c3 c4} ==> r9c9 <> 1
interaction row r9 with block b7 for number 1 ==> r8c2 <> 1
nrc-chain[4]  n1{r5c8 r8c8} - {n1 n5}r8c9 - {n5 n8}r9c9 - n8{r9c5 r5c5} ==> r5c8 <>
8
singles ==> r4c8 = 8, r4c4 = 9, r4c3 = 4, r4c2 = 5, r5c3 = 9
interaction column c3 with block b1 for number 6 ==> r3c1 <> 6
hidden-single-in-a-row ==> r3c9 = 6
interaction column c3 with block b1 for number 6 ==> r2c1 <> 6, r1c1 <> 6
interaction column c2 with block b7 for number 4 ==> r8c1 <> 4
naked-pairs-in-a-column {n2 n6}{r5 r7}c2 ==> r9c2 <> 2, r8c2 <> 6
nrc-chain[3]  n6{r1c5 r1c3} - n2{r1c3 r9c3} - {n2 n6}r7c2 ==> r7c5 <> 6
naked-singles
GRID 0 SOLVED. B-NRCZT LEVEL = B-NRCZT7, MOST COMPLEX RULE = NRCZT7
492561837
586327491
713849526
154936782
269784153
837152649
925618374
678493215
341275968
```

Third example, in which the most complex rule is a braid:

Code:

```
(solve
"..53.81.2........982..9.3..9..1.5.2..429..5...1..7...82..7...1..5........3.4....6")
*****  SudoRules version 13.7wbis-B2  *****
..53.81.2........982..9.3..9..1.5.2..429..5...1..7...82..7...1..5........3.4....6
singles ==> r1c2 = 9, r5c5 = 8, r8c4 = 8, r5c9 = 1, r6c1 = 5
interaction column c4 with block b2 for number 5 ==> r2c5 <> 5
x-wing-in-rows n7{r1 r5}{c1 c8} ==> r9c8 <> 7, r9c1 <> 7
naked-single ==> r9c1 = 1
```

```
x-wing-in-rows n7{r1 r5}{c1 c8} ==> r8c8 <> 7, r8c1 <> 7
interaction block b7 with column c3 for number 7 ==> r4c3 <> 7, r3c3 <> 7, r2c3 <> 7
x-wing-in-rows n7{r1 r5}{c1 c8} ==> r3c8 <> 7, r2c8 <> 7, r2c1 <> 7
xyz-chain[3]  {n6 n7}r2c2 - {n7 n4}r1c1 - {n4 n6}r8c1 ==> r2c1 <> 6
nrc-chain[3]  n1{r3c3 r3c6} - n7{r3c6 r2c6} - {n7 n6}r2c2 ==> r3c3 <> 6
nrct-chain[3]  n4{r6c6 r4c5} - {n4 n6}r1c5 - n6{r3c4 r6c4} ==> r6c6 <> 6
nrczt-whip-rn[3]  n6{r3c4 r3c8} - n6{r5c8 r5c1} - {n6r1c1 .} ==> r2c6 <> 6
nrct-chain[4]  {n4 n6}r1c5 - n6{r3c4 r6c4} - {n6 n3}r5c6 - {n3 n4}r4c5 ==> r2c5 <> 4
nrct-chain[4]  n4{r6c6 r4c5} - {n4 n6}r1c5 - n6{r3c4 r6c4} - n2{r6c4 r6c6} ==> r6c6
<> 3
nrczt-whip-cn[4]  n7{r3c6 r2c6} - n4{r2c6 r6c6} - n2{r6c6 r6c4} - {n6r6c4 .} ==>
r3c6 <> 6
nrczt-whip-rc[4]  n6{r3c4 r6c4} - {n6 n3}r5c6 - {n3 n4}r4c5 - {n4r1c5 .} ==> r2c5 <>
6
nrczt-whip-rn[5]  n1{r8c6 r8c5} - {n1 n2}r2c5 - {n2 n5}r9c5 - n5{r9c8 r7c9} -
{n3r7c9 .} ==> r8c6 <> 3
nrczt-braid-rn[5]  {n4 n6}r1c5 - n6{r3c4 r6c4} - n7{r1c1 r5c1} - n6{r5c1 r5c8} -
{n6r3c8 .} ==> r1c1 <> 4     <-------
naked-pairs-in-a-block {n6 n7}{r1c1 r2c2} ==> r2c3 <> 6
nrc-chain[2]  n4{r1c8 r1c5} - n4{r4c5 r6c6} ==> r6c8 <> 4
nrczt-whip-cn[2]  n4{r1c8 r8c8} - {n4r8c1 .} ==> r2c7 <> 4
nrczt-whip-bn[3]  n7{r2c6 r3c6} - n1{r3c6 r3c3} - {n4r3c3 .} ==> r2c6 <> 4
nrct-chain[5]  n4{r8c1 r2c1} - n3{r2c1 r2c3} - n3{r6c3 r6c8} - n3{r4c9 r4c5} -
n3{r8c5 r8c9} ==> r8c9 <> 4
nrct-chain[4]  n7{r3c6 r3c9} - n5{r3c9 r7c9} - n4{r7c9 r4c9} - n4{r6c7 r6c6} ==>
r3c6 <> 4
singles ==> r1c5 = 4, r6c6 = 4, r6c4 = 2
nrct-chain[5]  {n5 n6}r2c4 - n6{r3c4 r3c8} - n6{r1c8 r1c1} - {n6 n4}r8c1 - n4{r8c8
r2c8} ==> r2c8 <> 5
singles ==> r2c4 = 5; r3c4 = 6
nrczt-whip-cn[5]  n4{r8c1 r2c1} - n4{r2c8 r3c8} - n5{r3c8 r9c8} - {n5 n2}r9c5 -
{n2r9c7 .} ==> r8c7 <> 4
nrczt-whip-cn[5]  n3{r8c9 r4c9} - n4{r4c9 r4c7} - n4{r7c7 r7c9} - n5{r7c9 r7c5} -
{n3r7c5 .} ==> r8c8 <> 3
interaction column c8 with block b6 for number 3 ==> r4c9 <> 3
nrc-chain[3]  n7{r5c1 r5c8} - n3{r5c8 r6c8} - {n3 n6}r6c3 ==> r5c1 <> 6
xyt-chain[4]  {n9 n4}r8c8 - {n4 n6}r8c1 - {n6 n8}r7c2 - {n8 n9}r7c7 ==> r9c8 <> 9
hxy-cn-chain[4]  {r8 r6}c8n9 - {r6 r5}c8n3 - {r5 r2}c1n3 - {r2 r8}c1n4 ==> r8c8 <> 4
singles ==> r8c8 = 9, r6c7 = 9
interaction column c8 with block b3 for number 4 ==> r3c9 <> 4
interaction row r8 with block b7 for number 4 ==> r7c3 <> 4
naked-triplets-in-a-column {n6 n7 n3}{r1 r5 r6}c8 ==> r2c8 <> 6
nrct-chain[3]  n6{r5c6 r4c5} - n6{r4c7 r2c7} - n6{r2c2 r7c2} ==> r7c6 <> 6
nrc-chain[3]  {n3 n6}r4c5 - n6{r5c6 r8c6} - n1{r8c6 r8c5} ==> r8c5 <> 3
hidden-single-in-a-row ==> r8c9 = 3
interaction block b9 with column c7 for number 7 ==> r4c7 <> 7, r2c7 <> 7
hidden-pairs-in-a-column {n2 n7}{r8 r9}c7 ==> r9c7 <> 8
xy-chain[3]  {n6 n4}r4c7 - {n4 n8}r7c7 - {n8 n6}r7c2 ==> r4c2 <> 6
interaction block b4 with column c3 for number 6 ==> r8c3 <> 6, r7c3 <> 6
nrc-chain[3]  n8{r7c7 r2c7} - n6{r2c7 r2c2} - {n6 n8}r7c2 ==> r7c3 <> 8
singles ==> r7c3 = 9, r7c6 = 3, r5c6 = 6, r4c5 = 3
hidden-single-in-a-row ==> r9c6 = 9
xy-chain[4]  {n1 n2}r8c6 - {n2 n7}r8c7 - {n7 n4}r8c3 - {n4 n1}r3c3 ==> r3c6 <> 1
naked-singles
GRID 0 SOLVED. B-NRCZT LEVEL = B-NRCZT5, MOST COMPLEX RULE = B-NRCZT5
695348172
374512689
821697345
986135427
742986531
513274968
269753814
457861293
138429756
```

Back to top profile pm www

PIsaacson

Joined: 02 Jul 2008
Posts: 343
Location: Campbell, CA

Posted: Sat Nov 07, 2009 8:03 am Post subject: quote

Denis,

Thanks for posting some braids examples. Upon reviewing them, I can see that I still have many things to correct in my current braids approach. One of the biggest being that when I'm in braids construction mode, I never consider looking for standard nrczt whips/chains.

So I have to ask:

At a given depth/length level, do you first match nrczt chains, then nrczt whips, then braid chains, and finally

briad whips? Or something like that? If so, then I have a lot of work ahead to emulate that!

Cheers,
Paul

P.S. Is there a link to the suexg-cb large collection of 568,705 controlled bias puzzles?

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

Posted: Sat Nov 07, 2009 8:36 am Post subject: quote edit

PIsaacson wrote:

> At a given depth/length level, do you first match nrczt chains, then nrczt whips, then braid chains, and finally briad whips? Or something like that? If so, then I have a lot of work ahead to emulate that!

In the last version of SudoRules, there are no longer nrczt-chains or lassos: they are completely replaced by whips.
For any fixed length, the matching order is:
2D-chains (in ascending order: xy-chain, hxy-chain, xyt-chain, hxyt-chain, xyzt-chain, hxyzt-chain)
3D chains/whips (in ascending order: nrc-chains, nrct-chains, nrczt-whips)
nrczt-braids (I haven't implemented separate nrct-braids).

BUT:

nrczt-whips subsume all the previous rules, so you need not implement them separately if you don't want the simplest of simplest rules. I find it more appealing to have specific chains instead of always whips, but this is irrelevant to rating matters.

Generally, I don't activate braids at all, for several reasons:
- they are much slower than whips; this is not a matter of implementation, because they allow more possibilities, most of which are already taken care of by equivalent whips;
- they have a very limited impact on rating, as stated in my previous post;
- whips can solve all the random puzzles I've ever seen (~ 10,000,000), which provides a much stronger assertion than if I had said "braids" instead.

My implementation of braids in SudoRules relies heavily on whips. Partial braids that are equivalent to a partial whip (i.e. that have the same target and the same set of right linking candidates) are never generated; checking this is time consuming but avoids memory explosion. On a purely logical level, this means that the definition of a braid is restricted to not being equivalent to a whip. In no case does this mean that the passage by left-linking candidates could be avoided (as suggested elsewhere by Red Ed).

PIsaacson wrote:

> Is there a link to the suexg-cb large collection of 568,705 controlled bias puzzles?

Not yet. I'm currently updating my web page on classification. It will be there (hopefully) in a few days.
But it won't change anything: for this purpose, the Sudogen0_1M collection is as good as any other.

Back to top profile pm www

PIsaacson

Joined: 02 Jul 2008
Posts: 343
Location: Campbell, CA

Posted: Sat Nov 07, 2009 8:51 am Post subject: quote

Denis,

Thanks for the reply. Since implementing whips, I don't think I've ever seen a full standard nrczt chain produced. But I'm wondering if I have an edge case somewhere in my whips logic that is causing the slight differences in some of our nrczt ratings. I'm going to review my whips checking and zt promotion code yet again to see if anything pops. I'm also reading about the CLIPS engine to see if there are some fundamental concepts that might help me to exactly emulate SudoRules. So close, but no cigar...

Cheers,
Paul

Back to top profile pm

Red Ed

Joined: 06 Jun 2005
Posts: 967

Posted: Sat Nov 07, 2009 9:05 am Post subject: quote

> **denis_berthier wrote:**
>
> On a purely logical level, this means that the definition of a braid is restricted to not being equivalent to a whip.

Your *definition* apparently has no such restriction:

> **Quote:**
>
> Informally, given a target candidate z, an nrczt-braid built on z is, as an nrczt-whip, a linear sequence of candidates, alternatively called left-linking and right-linking. Its precise definition is obtained from that of an nrczt-whip by merely changing the condition "any left-linking candidate is nrc-linked to the previous right-linking one" by "any left-linking candidate is nrc-linked to a previous right-linking one or to the target".

Just so we all understand: when you mentioned the restriction of not being equivalent to a whip, were you referring just to a program that finds non-whip braids, or are you updating the definition of braids to specifically exclude whips?

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

Posted: Sat Nov 07, 2009 9:16 am Post subject: quote edit

> **Red Ed wrote:**
>
> > **denis_berthier wrote:**
> >
> > On a purely logical level, this means that the definition of a braid is restricted to not being equivalent to a whip.
>
> Your *definition* apparently has no such restriction:
>
> > **Quote:**
> >
> > Informally, given a target candidate z, an nrczt-braid built on z is, as an nrczt-whip, a linear sequence of candidates, alternatively called left-linking and right-linking. Its precise definition is obtained from that of an nrczt-whip by merely changing the condition "any left-linking candidate is nrc-linked to the previous right-linking one" by "any left-linking candidate is nrc-linked to a previous right-linking one or to the target".
>
> Just so we all understand: when you mentioned the restriction of not being equivalent to a whip, were you referring just to a program that finds non-whip braids, or are you updating the definition of braids to specifically exclude whips?

I thought it was clear I was speaking of the SudoRules implementation.
I'm not changing my definition of a braid.
Whips have always been a very special kind of braid.
In the SudoRules implementation, whips are given a preference over braids of the same length - which is very natural, as whips are simpler. As a result, only braids that are not equivalent to whips can be explicitly found as braids when braids are looked for. But of course, all the whips found are braids.

Back to top profile pm www

David P Bird

Joined: 17 Sep 2008
Posts: 206
Location: Middle England

Posted: Thu Nov 12, 2009 12:37 am Post subject: quote

> **denis_berthier wrote:**
>
> Unfortunately, for C, Pascal, Java or whatever procedural language you chose for your implementation, I'm almost totally incompetent. But if you have any questions that you can express in natural language, I'll answer

<Source>

Denis, this quote from you reveals that you too have difficulty in understanding someone else's points unless they are prepared to present them in natural language, Perhaps you then have a glimmer of how obscure much of your writing is to non-mathematicians!

In the opening posts of this thread you cite a reference to an earlier post, but that simply is the start of a recursive series of other references which I (like you with C++ & Pascal) am too incompetent to follow. I would therefore be very grateful if you could write a plain man's guide to nrctz whips and braids! No need for proofs in the main, just the do's and don'ts of how to follow the various steps required to reach a deduction.

Let me give you feedback on the impressions that I've reached in trying to follow the basics of your various points - note I'm not asserting that you have categorically claimed any of them, just listing what I've inferred. This will then give you the chance to correct any misconceptions I (and possibly others) have.

The nrctz methods:

> Use only conjugate and weak links plus locked sets in various data representations.
> Are straightforward to master.
> Don't involve guessing, trial and error, assuming uniqueness, or net-based deductions.
> Solve nearly every possible Sudoku puzzle.
> Outperform all other solving methods.
> Require a computer to follow all the possible paths that need to be explored.

Within this list there are also two very big sticking points between us:

Firstly your definition of a net based deduction is far too lax for me because you allow the contents of previously visited cells to be remembered and used as secondary links. If we were to draw the lines representing these secondary links in addition to the main path followed, we would see what most people would call a net. It boils down to the question of when is a net not a net? I don't think you should get away with a unilateral redefinition and that you should coin and define a new term for the (admittedly) severely restricted nets you allow.

Mind you, I take just as critical a view of so-called AICs that allow a limited number of branches to be followed for an indefinite length, so you are far from being alone!
I believe that there is a difficulty ceiling above which we need to allow logical ANDs into our solutions. Above that ceiling we should then rank our solutions according the ANDs they contain.

The second sticking point is an even bigger issue. It boils down to the nature of the basic premise that ncrtz methods allow. Some seem to depend on assuming a particular truth state for a candidate (or target) which I can't distinguish from starting with a guess, then repeating that process for all the (x)s in a house to see if any of them can be eliminated, which I can't distinguish from brute force trial and error. As I've posted before, if a chain is unidirectional it indicates that it is based on a single case (or guess) being tested, but if it is bi-directional it shows that an either/or scenario is being explored, and so there is a vast difference between the two. I accept that any deductions reached by either method will be logical, but that isn't the issue.

It's doubtful that you and I will ever be fully in accord, but I feel I should try to reach a better understanding of your approach, particularly now our community seems to have shrunk so much

Back to top profile pm

denis_berthier

Posted: Thu Nov 12, 2009 5:39 am Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

David P Bird wrote:

> In the opening posts of this thread you cite a reference to an earlier post, but that simply is the start of a recursive series of other references which I (like you with C++ & Pascal) am too incompetent to follow. I would therefore be very grateful if you could write a plain man's guide to nrctz whips and braids! No need for proofs in the main, just the do's and don'ts of how to follow the various steps required to reach a deduction.

As for many other topics in this forum (and others), my threads are plagued with irrelevant or confrontational material and difficult to follow, although I always try to have very explicit posts, especially those that start with capital bold titles. As of now, the best free place to start with is my website. Although some of the topics developed after the second edition of my book are not (not yet?) written as they should be for a book, all the definitions are precise.
I've already spent much more time than I wanted for Sudoku. I'm not very likely to do much more now.

David P Bird wrote:

> The nrctz methods: Use only conjugate and weak links

NO. "Weak links" is "inference level" vocabulary. I never use this. I use direct nrc-links (candidates "seeing" each other) and conjugacy modulo something, both of which are defined at the purely factual level. This makes my definitions non ambiguous.
As long as you don't understand that nothing in my approach is described in terms of weak or strong inferences, you will think at the wrong level and it's hopeless.

> **David P Bird wrote:**
>
> Are straightforward to master.

No. As anything new, some effort is required.

> **David P Bird wrote:**
>
> Don't involve guessing, trial and error, assuming uniqueness, or net-based deductions.

Right.
For uniqueness, I've never found a puzzle that nrczt-whips couldn't solve but that they could solve if I added uniqueness. But the uniqueness assumption is not incompatible with them.

> **David P Bird wrote:**
>
> Solve nearly every possible Sudoku puzzle.

Yes. Proven for 10,000,000 randomly generated puzzles, for various kinds of generators.
Moreover, it can be extended to any finite Constraints Satisfaction Problem.

> **David P Bird wrote:**
>
> Outperform all other solving methods.

I've proven that they subsume (i.e. are more general than) many other well-known rules.
I don't know what you mean by "outperform". I've never tried to deter anyone from using other rules.

> **David P Bird wrote:**
>
> Require a computer to follow all the possible paths that need to be explored.

No.
I've given many times indications on how to make it easier to find nrczt-chains:
- learn according to the sequence: xy, hxy, xyz, hxyz, xyt, hxyt, xyzt, hxyzt, nrc, nrcz, nrct, nrczt
- use the property of composability,
- use th extended Sudoku board.

> **David P Bird wrote:**
>
> Firstly your definition of a net based deduction is far too lax for me because you allow the contents of previously visited cells to be remembered and used as secondary links. If we were to draw the lines representing these secondary links in addition to the main path followed, we would see what most people would call a net. It boils down to the question of when is a net not a net? I don't think you should get away with a unilateral redefinition and that you should coin and define a new term for the (admittedly) severely restricted nets you allow.

See the definition: a whip is first of all a SEQUENCE of nrc-linked candidates.
In your view, triplets or quads must be nets!!!

> **David P Bird wrote:**
>
> if a chain is unidirectional it indicates that it is based on a single case (or guess) being tested, but if it is bi-directional it shows that an either/or scenario is being explored, and so there is a vast difference between the two.

Wrong. A good example of "thinking at the wrong level".
An xyt- or nrct- chain is not based on any predefined target. So, the question you're raising shouldn't be about the t-extension (the cause of uni-directionality) - it should be about the z-extension.
Reversibility has some advantages, but what's most important is non-anticipativeness and composability.

Back to top profile pm www

David P Bird

Joined: 17 Sep 2008
Posts: 206
Location: Middle England

Posted: Thu Nov 12, 2009 12:33 pm Post subject: quote

Denis, thanks for taking the time to respond.

> **denis_berthier wrote:**
>
> NO. "Weak links" is "inference level" vocabulary. I never use this. I use direct nrc-links (candidates "seeing" each other) and conjugacy modulo something, both of which are defined at the purely factual level. This makes my definitions non ambiguous. As long as you don't understand that nothing in my approach is described in terms of weak or strong inferences, you will think at the wrong level and it's hopeless.

OK then, but would it be fair to say that nrc links use NAND logic and conjugacies use XOR logic which is what these familiar link terms convey? If so, it's not that hopeless, just another question of terminology.

> **denis_berthier wrote:**
>
> > **David P Bird wrote:**
> >
> > Don't involve guessing, trial and error, assuming uniqueness, or net-based deductions.
>
> Right

But here are further questions of terminology. To me all these claims except the uniqueness one are questionable as they depend on our respective definitions of these terms. At the moment I would put "particular types of" in front of the other three until I'm happy that you've used them in their generally accepted senses!

> **denis_berthier wrote:**
>
> > **David P Bird wrote:**
> >
> > Require a computer to follow all the possible paths that need to be explored.
>
> No.
> I've given many times indications on how to make it easier to find nrczt-chains:
> - learn according to the sequence: xy, hxy, xyz, hxyz, xyt, hxyt, xyzt, hxyzt, nrc, nrcz, nrct, nrczt
> - use the property of composability,
> - use the extended Sudoku board.

Well, another impression I've gained is that although it may be possible to follow your logic manually, for the hardest puzzles it would require reams of paper, and as yet no one has ever achieved it.

Now some more feedback! In each of your pieces you assume that because you've previously defined your terms elsewhere, your readers will be familiar with them. Not so! It's virtually impossible to navigate though the various pieces you bring together in chronological order. Furthermore, some of your earlier work has been superseded. Consequently it's extremely difficult for someone who hasn't taken a direct interest from the start to follow all your points. That's why I'd appreciate that plain man's guide.

I searched for a definition of 'composability' but only found a vague concept of building logical constructs out of previously identified 'chunks'. It seems to relate to a breadth first search approach, but whether it is practical for a manual solver on a difficult puzzle with a multitude of chunks to scan for connections seems questionable to me. BTW that's not a problem that's confined to your nrctz approach.

> **denis_berthier wrote:**
>
> In your view, triplets or quads must be nets!!!

No, I'm happy to consider them as easily recognisable Locked Sets. However you duck the issue that your interpretation of what constitutes a net is different from the generally accepted one.

> **denis_berthier wrote:**
>
> > **David P Bird wrote:**
> >
> > if a chain is unidirectional it indicates that it is based on a single case (or guess) being tested, but if it is bi-directional it shows that an either/or scenario is being explored, and so there is a vast difference between the two.
>
> Wrong. A good example of "thinking at the wrong level".
> An xyt- or nrct- chain is not based on any predefined target. So, the question you're raising shouldn't be about the t-extension (the cause of uni-directionality) - it should be about the z-extension.
> Reversibility has some advantages, but what's most important is non-anticipativeness and

> composability.

I'm prepared to accept that there's a difference between blind guessing and informed guessing, but we're back to definitions again I believe. However I'm ready to be convinced that a method for identifying key candidates that are worthy of being tested could be made more acceptable, and may be preferable to opening the doors wide to AND constructs.

I'm sorry that you intend to drop out now as I'd hoped that once you'd satisfied your appetite for gathering statistics you'd turn to identifying puzzle attributes that cause them to be unsolvable using the more basic methods (which is surely worth your attention).

Although I'm sure you're happy with the number of puzzles nrctz logic will solve, it would be very interesting for the rest of us to have a collection of puzzles that it won't!

Back to top profile pm

denis_berthier

Posted: Thu Nov 12, 2009 4:55 pm Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

> **David P Bird wrote:**
>
> > **denis_berthier wrote:**
> >
> > > NO. "Weak links" is "inference level" vocabulary. I never use this. I use direct nrc-links (candidates "seeing" each other) and conjugacy modulo something, both of which are defined at the purely factual level. This makes my definitions non ambiguous. As long as you don't understand that nothing in my approach is described in terms of weak or strong inferences, you will think at the wrong level and it's hopeless.
> >
> > OK then, but would it be fair to say that nrc links use NAND logic and conjugacies use XOR logic which is what these familiar link terms convey?

No. At best, it is the other way round: inferences rely on nrc-links.
nrc-links rely only on the grid structure.

> **David P Bird wrote:**
>
> > some of your earlier work has been superseded.

2D classification results have been superseded by 3D ones.
No 2D definition or theorem has been superseded by 3D definitions or theorems. They remain unchanged.

I'm not entering a nth debate about terminology. A good definition is not a definition that one likes a priori. It is one that leads to productive results.

Back to top profile pm www

denis_berthier

Posted: Thu Nov 12, 2009 5:26 pm Post subject: quote edit

Joined: 19 Jun 2007
Posts: 1079
Location: Paris, France

> **David P Bird wrote:**
>
> > I'd hoped that ... you'd turn to identifying puzzle attributes that cause them to be unsolvable using the more basic methods (which is surely worth your attention).
> > Although I'm sure you're happy with the number of puzzles nrctz logic will solve, it would be very interesting for the rest of us to have a collection of puzzles that it won't!

The answer is already there and very simple:
- there are only 3 levels of T&E at which a puzzle can be solved: 0, 1 or 2;
- all the puzzles at level 0 can be solved by singles;
- all the puzzles at level 1 can be solved by nrczt-braids and probably by nrczt-whips;
- puzzles at level 2 are very rare (none found in 10,000,000 randomly generated); they need more complex braids (such as braids(LS) or braids(FP) for some family FP of elementary patterns).

Back to top profile pm www

Display posts from previous: All Posts Oldest First Go

Stop watching this topic

Jump to: Advanced solving techniques Go

You **can** post new topics in this forum
You **can** reply to topics in this forum
You **can** edit your posts in this forum
You **can** delete your posts in this forum
You **can** vote in polls in this forum