Sudoku Players' Forums

Abominable TRIAL-and-ERROR and lovely BRAIDS

Author | **Message**

Red Ed

Posted: Sun Jan 17, 2010 8:28 pm Post subject: quote

Joined: 06 Jun 2005
Posts: 1054

Mauricio wrote:

> I have implemented whips, with and without llc reuse. The one with llc reuse is only slightly slower than my braids implementation and both are much faster than the one without llc reuse.

Personally I've never liked the rule that 'L' candidates cannot be reused; but I found a good workaround:

- Write the main search routine (in my case BFS, in yours DFS) in such a way that it does **not** keep track of 'L' candidates.
- At the start, have an array level_ok[] with all entries False. When the main search routine enters level L, if level_ok[L]=False then:
  - do a restricted DFS search (call this "R-DFS");
  - if Success, set level_ok[L]=True and continue the main search;
  - else (if Failure), return back to level L-1 in the main search.
- When the main search routine thinks it's found a jackpot (which will be a valid braid / contradiction net but perhaps not a valid whip), check with R-DFS before declaring success.
- The R-DFS routine performs a DFS search whilst keeping track of 'L' candidates, but with the restriction that it can only use the 'R' candidates that the main search has used up to this point. That makes R-DFS very fast. R-DFS declares Success if it is able to use up all of the 'R' candidates with no 'L' reuse (note it doesn't need to bother checking for a contradiction: that's the job of the main search routine); otherwise it declares Failure.

Back to top profile pm

Allan Barker

Posted: Mon Jan 18, 2010 3:22 am Post subject: quote

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Mauricio wrote:

> I have implemented whips, with and without llc reuse. [....] To compare, here are some samples of the same puzzle solved by Denis here

Code:

```
1000500090007000308700000040080000005300906000003024000600048000059003100010000000
```

nrczt whip/chain solutions, various options

Just for reference, here are some results from my nrczt chain/whip solver. The main point of interest is a comparison of various combinations of other options. None of these results have L candidate reuse since I don't have that option. I also do not have braids. The options include:

POT Allow overlapping truths in path. (default in nrczt CWs)
ILoops Allow internal-loops. (default in nrczt CWs)
GL Allow group-links (a.k.a hinges, whips[BI]).

Both the length of the longest chain and the computational times show the expected trends. The one possibly surprising result is the faster computational times obtained from allowing group links. A group link takes roughly twice the computational effort however, the reduction of the path lengths is responsible for the better times.

Code:

```
method    max length     time              options
          (truths)       (sec)

nrczt         16           11.4          POT, ILoops (standard)
nrczt         18           13.1          POT
nrczt         22           10.7          ILoops
nrczt         22           12.1           --

With group links.

nrczt         12            9.9          POT, ILoops, GL
nrczt         12            3.0          POT, GL        (best time)
nrczt         14           13.4          ILoops, GL
nrczt         16            6.5          GL
```

A ribbon solution, with options +GL, -POT, -ILoop, takes 6.9 seconds with a maximum length of 12 truths. However, the ribbon solution uses fewer total truths (221) to solve the puzzle vs. whips+GL (267) (whip[BI]}, and whips alone (325), (whip[ZIP]).

Back to top profile pm www

denis_berthier

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

Posted: Mon Jan 18, 2010 9:32 am Post subject: quote edit

braids vs contradiction nets

Red Ed seems to be assimilating braids with contradiction nets. Setting aside his personal motivations for this, this raises a few questions.

1) Forcing nets/ contradiction nets are defined in Sudopedia as follows:

sudopedia wrote:

> The player selects a candidate to be tested. When this candidate belongs to a bivalue cell, finding a contradiction allows the player to place the remaining candidate in that cell. The tested candidate is assumed to be true and the implications are analyzed until a contradiction is found or no further progress can be made.

This definition is that of a procedure - a trial and error procedure - not of a pattern. It is so vague that almost anything can be a contradiction net. In this sense, a braid is a contradiction net. But it would be more correct to say that my T&E procedure is a contradiction net in this sense.

Now, what does "the implications are analyzed until a contradiction is found" mean? Nothing precise. Does this correspond to T&E(NS+HS) or to T&E(NS+HS+BI) or to T&E(FP) (and for which FP families) ? The only example in sudopedia suggests that only candidates can be used (and therefore even the T&E(NS+HS+BI) are not included)

2) Did this vague definition lead to any theory about contradiction nets - e.g. equivalence with a precisely defined T&E procedure or to confluence property?
Obviously no. If you search this forum for forcing nets / contradiction nets, you'll find almost nothing.

It is one thing to give vague definitions. It is another thing to do something with them.

Contrary to these vague definitions, nrczt-braids and braids(FP) are precisely defined as patterns and precise theorems and classification results can be stated with them.

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 1054

Posted: Mon Jan 18, 2010 9:45 am Post subject: quote

To me, while coding, braids(*) look like contradiction nets using singles. I agree that when I wrote "contradiction net" without qualification, I was not precise. The reason for writing c-net at all was to help other programmers who might have been struggling with all the reams of braid posts to realise that, as an algorithm challenge at least, there's nothing new in braids over & above c-nets with singles. In particular, programmers can ignore 'L' candidates in their search (obvious if you're thinking in c-net terms; rather less so if working with the original braids posts), and instead make an arbitrary choice of 'L's for cosmetic/display purposes at the end.

* - without additional "FP" patterns of course

Back to top profile pm

denis_berthier

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

Posted: Mon Jan 18, 2010 11:23 am Post subject: quote edit

Left-linking candidates are not there for 'cosmetic purposes', they are an essential part of the braids logic. Would anyone write an AIC with only half its candidates?

Whether you can reconstruct them later in some particular implementation is a matter of programming. It doesn't justify assimilating a braid (a sequence L1 R1 L2 R2 ...) with a vaguely defined procedure.
This is another point not present in the pseudo-definition of a contradiction net but present in braids: sequentiality. You can always say that one can reconstruct it later, but, from a patterns POV, this is a noticeable difference - a difference a procedural programmer should understand.

No one neeeds to read all the irrelevant posts laid on this thread to know the definition of a braid: it's all in a single place.

Therefore I can't see any reason, but introducing confusion, in assimilating the perfectly defined braid pattern with the vaguely defined T&E-ish contradiction net procedure.

Back to top profile pm www

David P Bird

Joined: 17 Sep 2008
Posts: 234
Location: Middle England

Posted: Mon Jan 18, 2010 1:14 pm Post subject: quote

> **Red Ed wrote:**
> there's nothing new in braids over & above c-nets with singles.

A relatively simple basis that can be used to analyse the structure of any deduction is to count the number of items of information that must be carried forward at each step in a sequence of inferences and the number of steps each item must be retained before it can be discarded as being of no further use. Of course how the results are interpreted is another issue! (I tried to float this idea in Eureka in 2007 but it was a lead balloon.)

On this basis, forcing nets and nrczt's and would appear to be much the same. The differences are in the method used to locate the deduction, that is, how many items of information were actually retained just in case they were needed later?

Of course both are still unacceptable to me.

Back to top profile pm

David P Bird

Joined: 17 Sep 2008
Posts: 234
Location: Middle England

Posted: Mon Jan 18, 2010 1:41 pm Post subject: quote

As an afterthought to my previous post:

Considering that DB's program will unintelligently explore every possible candidate in turn until a deduction is found, it depends how quick that was as to whether it was any better than using a forcing net from the start.

Which takes us back to writing programs that can exercise judgment.

Back to top profile pm

denis_berthier

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

Posted: Mon Jan 18, 2010 2:26 pm Post subject: quote edit

> **David P Bird wrote:**
>
> > **Red Ed wrote:**
> >
> > there's nothing new in braids over & above c-nets with singles.

I've already answered this in my previous post. As this is obviously FALSE, I don't see the point of repeating it.

> **David P Bird wrote:**
>
> On this basis, forcing nets and nrczt's and would appear to be much the same.

nrczt what? whips or braids? It makes a large difference: chain vs net.

> **David P Bird wrote:**
>
> The differences are in the method used to locate the deduction, that is, how many items of information were actually retained just in case they were needed later

False. The main differences are:
- between a *procedure* and a *pattern* (*program vs logic*)
- between a *vaguely defined* procedure and a *precise mathematical definition* of a pattern as a sequence of candidates.

> **David P Bird wrote:**
>
> Of course both are still unacceptable to me.

OK, but then why do you waste your time discussing them?

> **David P Bird wrote:**
>
> Considering that DB's program will unintelligently explore every possible candidate in turn until a deduction is found, it depends how quick that was as to whether it was any better than using a forcing net from the start.

As braids are not a program but patterns, all this is meaningless.
The meaning and interest of braids (as any other pattern) doesn't depend on the way they are programmed. It doesn't even depend on their being programmed at all.
If different implementations of braids can be compared (e.g. SudoRules, Mauricio's, Allan's ribbon procedure, RedEd's), it is only because they are defined in a purely logical way, not depending on any implementation.

> **David P Bird wrote:**
>
> Which takes us back to writing programs that can exercise judgment.

Thanks for this very deep insight. Let us know when you have an intelligent program.

Back to top profile pm www

Allan Barker

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Posted: Mon Jan 18, 2010 2:58 pm Post subject: quote

> **denis_berthier wrote:**
>
> If different implementations of braids can be compared (e.g. SudoRules, Mauricio's, Allan's ribbon procedure, RedEd's), it is only because they are defined in a purely logical way, not depending on any implementation.

Correction, I do not yet have any braids implimentation, either in my nrczt solver or in my ribbon solver. Would like too, but .....

Back to top profile pm www

David P Bird

Joined: 17 Sep 2008
Posts: 234
Location: Middle England

Posted: Mon Jan 18, 2010 5:07 pm Post subject: quote

Denis, I don't see much point in responding in detail to every one of your counters.

Actually what I was trying to point out was that although the inferences that have been employed by your methods and by forcing nets may be identical, the way they would be found wasn't. However, without some form of goal prioritisation there would be very little difference between running your pattern recognising algorithm repeatedly and recognising the same patterns using forcing nets.

My wording there reflects that I find word play about what constitutes a pattern recognition system being different from tracking logical sequences to be hogswash. Unless the "pattern" is "instantly recognisable" one process mimics the other. I use the quotation marks for two terms that are freely used but defy precise definition.

The term "fitness for purpose" springs to mind when I consider solving aids, and solving programs. Your program hardly replicates what a human solver could be expected to work out using a restricted supply of paper and we already have the dancing links program if we only need solutions, so where does it fit in? Using your basic methods on easy problems trivialises them, while employing your higher level methods for the harder ones would be sheer drudgery done manually. The same goes for forcing nets, and which is why I made my personally unacceptable comment.

I had a brief exchange with **Paul** somewhere about means of assessing the most promising targets to try first, but can't locate it quickly. Perhaps you will recall it or he can fill you in.

[edit] Found it! It's < here >

Back to top profile pm

denis_berthier

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

Posted: Mon Jan 18, 2010 6:06 pm Post subject: quote edit

David,
What I discuss here is resolution rules. I don't care about implementation issues. And if you're still unable to make a difference between a pattern and a procedure, this is hopeless.
If you have so bright ideas, why don't you apply them yourself, instead of harping on about them?

Last edited by denis_berthier on Mon Jan 18, 2010 6:07 pm; edited 1 time in total

Back to top profile pm www

Allan Barker

Joined: 21 Feb 2008
Posts: 511
Location: Bangkok

Posted: Mon Jan 18, 2010 6:06 pm Post subject: quote

David P Bird wrote:

> **Red Ed wrote:**
>
> there's nothing new in braids over & above c-nets with singles.
>
> A relatively simple basis that can be used to analyse the structure of any deduction is to count the number of items of information that must be carried forward at each step in a sequence of inferences and the number of steps each item must be retained before it can be discarded as being of no further use. Of course how the results are interpreted is another issue! (I tried to float this idea in Eureka in 2007 but it was a lead balloon.)
>
> On this basis, forcing nets and nrczt's and would appear to be much the same. The differences are in the method used to locate the deduction, that is, how many items of information were actually retained just in case they were needed later?.

David, this was the point I recently made about the t- extensions in nrczt whips and chains. These t-extensions do require extra links to look back, thus making crosslinks. **This cross linking is real and**

must be addressed by <u>any</u> manual solver attempting to make such a chain. Efficient or not, every computer program must also account for them, <u>or not work</u>. Denial of this fact is not an option.

The fact that the chain is found as a linear sequence does not make the links go away.
The fact that more than one t-extension can use the same link is irrelevant.
The precise mathematical definition of nrczt chains/whips as patterns includes these links.

To be balanced, there are very easy computational methods to apply these links, which make nrczt chains fast and relatively powerful solvers. Without these links their power would be much less.

Back to top profile pm www

denis_berthier

Posted: Mon Jan 18, 2010 6:20 pm Post subject: quote edit

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

> **Allan Barker wrote:**
> this was the point I recently made about the t- extensions in nrczt whips and chains. These t-extensions do require extra links to look back, thus making crosslinks. This cross linking is real and must be addressed by any manual solver attempting to make such a chain. Efficient or not, every computer program must also account for them, or not work.

Of course.

> **Allan Barker wrote:**
> The fact that the chain is found as a linear sequence does not make the links go away.

Still OK, except that the nrczt-chain is not "found as a linear sequence", it **is** a linear sequence.

> **Allan Barker wrote:**
> The precise mathematical definition of nrczt chains/whips as patterns includes these links.

Totally false. Did you ever hear the word "abstraction"? Of course, the t-candidates must be justified by previous rlc's, but they are not part of the chain. Read my definitions.

Back to top profile pm www

Red Ed

Posted: Mon Jan 18, 2010 6:27 pm Post subject: quote

Joined: 06 Jun 2005
Posts: 1054

> **denis_berthier wrote:**
> Therefore I can't see any reason, but introducing confusion, in assimilating the perfectly defined braid pattern with the vaguely defined T&E-ish contradiction net procedure.

I don't think that the attitude that braids are the only form of contradiction net worth talking about is helpful. Most people, surely, understand what contradiction nets are, even if they can't pin down the details. Compare that with braids which, whilst defined fully, (a) came later and (b) are probably far less intuitive for most players.

Thinking of a braid as a linearised presentation of c-nets-with-singles seems to be an appropriate thing, recognising prior art, and not something to be slapped down with claims of "confusion".

Back to top profile pm

ronk

Posted: Mon Jan 18, 2010 6:35 pm Post subject: quote

Joined: 02 Nov 2005
Posts: 2760
Location: Southeastern USA

> **denis_berthier wrote:**
> > **Allan Barker wrote:**
> > The precise mathematical definition of nrczt chains/whips as patterns includes these links.
>
> Totally false. Did you ever hear the word "abstraction"? Of course, the t-candidates must be justified by previous rlc's, but they are not part of the chain. Read my definitions.

denis, using the *modulo* term in your definitions doesn't make the z-links and t-links disappear. Sweeping these links under the rug is creating some rather large moguls, which you and everyone else will just keep tripping over (mixing metaphors).

Back to top profile pm

Display posts from previous: All Posts Oldest First Go

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

All times are GMT + 1 Hour

Goto page Previous 1, 2, 3 ... , 13, **14**, 15 Next

Page 14 of 15

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