Sudoku Players' Forums

THE REAL DISTRIBUTION OF MINIMAL PUZZLES

Goto page Previous 1, 2, 3 ... , 34, 35, 36 Next

newtopic postreply **Sudoku Players' Forums Forum Index -> General/puzzle**

View previous topic :: View next topic

Author	Message
Red Ed Joined: 06 Jun 2005 Posts: 784	Posted: Sat Oct 10, 2009 12:47 pm Post subject: quote **eleven wrote:** > Using a singles solver only i could check about 33% very quickly: In almost 10% of random 24's one of the clues is forced by the others through singles. And in 23% more than 6 singles are forced by the givens. > Also the about 29% puzzles, where 4-6 singles are forced, could be checked brute force rather quickly. That's pretty much the trick that I use. I've not reported anything new today because I'm busy checking the discrepancy between my and Denis' estimates of the number of proper minimal 27-clue puzzles. Tomorrow, time permitting, I will update my code to do some checks to prune off puzzles solved with only 29 clues (it won't be obvious that I need to do this but, trust me, the current implementation does not enjoy dealing with supersets that are too close to having a single solution); then I'll run a bunch of tests to understand the distribution of number-of-31s arising from each 24. Finally, I'll post results with quantified variance and, provided I'm content that I've not left any glaringly slow stuff in my program, the code too. Back to top profile pm
denis_berthier Joined: 19 Jun 2007 Posts: 916 Location: Paris, France	Posted: Sat Oct 10, 2009 11:04 pm Post subject: quote **gsf**, I've just tried your sudz decompressor. I launched it in parallel with sudoku-darwin.i386, in the context of suexg-cb, i.e. I ran: ./sudoku-darwin.i386 -q- -f%v *.sudz | ./suexg-cb-optim48-U4 and ./sudz -f%g *.sudz | ./suexg-cb-optim48-U4 After a tea (31'), the two processes (launched with the same seed) were still running in perfect sync.

Half awaken, I first thought that the 64-bit mode wasn't faster than 32-bit for this program and I felt disappointed. But things are much better.

On my quadcore, 2 cores are running full time the 2 suexg-cb processes (in 64-bit mode), which gives the sync.
The other 2 processes, sudoku-darwin-i386 and sudz, use the other 2 cores and the process stats show that **sudz is almost 6 times faster than sudoku-darwin-i386**.

Details:
./suexg-cb-optim48-U4: 31' (each)
./sudoku-darwin.i386 : 11.41'
./sudz : 2'

Thanks to your complete collection and your decompressor, in the ./sudz -f%g *.sudz | ./suexg-cb-optim48-U4 combination, **on a single processor, complete grid "generation" now takes only 5.5% of the time** and any optimisation of the deletion part will have almost full impact on the whole process.

Paul, do you have any news of the other, faster solver?

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 784

Posted: Sun Oct 11, 2009 12:29 am Post subject: quote

> **denis_berthier wrote:**
> any optimisation of the deletion part will have almost full impact on the whole process.

Well, coupled with eleven's modification to introduce a sample rate parameter, that's good news for anyone that might want to use gsf's files for unbiased tests of some other, more expensive, type than your number-of-clues work; or who might want to do some quick tests of any hypothesis without necessarily expecting to extend the tests to a long experiment.

Back to top profile pm

denis_berthier

Joined: 19 Jun 2007
Posts: 916
Location: Paris, France

Posted: Sun Oct 11, 2009 2:19 am Post subject: quote

> **Red Ed wrote:**
> coupled with ... eleven's modification to introduce a sample rate parameter[/url], that's good news for anyone that might want to use gsf's files for unbiased tests of some other, more expensive, type than your number-of-clues work; or who might want to do some quick tests of any hypothesis without necessarily expecting to extend the tests to a long experiment.

In most cases, I wouldn't recommend this approach. In any case, it should be excluded to test any hypothesis that is strongly dependent on the number-of-clues distribution (and, as we generally don't know in advance if it is dependent or not, ...).

A single scan of gsf's collection produces 21,250 puzzles in the mean, in which there will be ~54 29s, ~6 30s, no 31s.
Taking only 10% or 20% of this (rate parameter), there will be so few 29s, 30s and 31s that the computed "unbiased" mean will be biased.

Consider the SER. It isn't strongly dependent on the number of clues (0.1 correlation) but there is nevertheless a clear trend: more clues => larger SER. If you want a reasonable estimate, you need samples with sufficiently many instances for each number of clues to compute the n-clue SER. So, how to do if you don't want to compute the SER for millions of puzzles?
My approach is as follows.
From a very large sample of controlled-bias minimals, extract much smaller samples for each number of clues (of course, do this in such a way that they will remain unbiased for this number of clues), use them separately to do the SER computations and assemble the results using the real distribution of clues.
I think anyone "who might want to do some quick tests of any hypothesis" will accept the necessity of having such n-clue unbiased samples.

Back to top profile pm www

Red Ed

Joined: 06 Jun 2005
Posts: 784

Posted: Sun Oct 11, 2009 2:45 am Post subject: quote

Denis, your comments address only experiments with minimal proper puzzles. My post was much more general. If, for example, someone wanted a quick unbiased estimate of average number of completions of a random 30-clue subgrid then they wouldn't want to do billions of tests, just a few million perhaps. In that case, they'd want to sample substantially less than 100% of gsf's grids. (Or in my case, I'd just use my generator.)

Back to top profile pm

Red Ed

Joined: 06 Jun 2005
Posts: 784

Posted: Sun Oct 11, 2009 3:52 am Post subject: quote

Red Ed wrote:

> Now 27s:
>
> - 112136 minimals => std err ~= 1/335
> - Scale up by choose(81,27) / (31 x 5472730538) to give number of proper minimal 27s per grid
> - => you estimate 1.5244e15 w. std err 0.0046e15
> - I had estimated 1.4364e15 w. std err 0.0433e15
> - So **our agreement or not on the number of 27s is inconclusive**
>
> ...
> We're 2 standard deviations apart, which is pretty serious but perhaps not conclusive.

I'll let my re-run experiment go to 1000000 solution grids before stopping it. However, mid-term results from that experiment suggest that we were just unlucky to see the low estimate (1.4364e15) reported previously. It happens.

On reflection, the p-value is not even as low as 2.2% as I'd said previously, since that is a p-value for the extent to which (est_you - est_me) > 0, when

really the right thing to measure is the extent to which |est_you - est_me| > 0. So the true p-value is about double, i.e. ~1/23, which whilst still unfortunate is not nearly so scary as before.

We should see this as good news: two methods that should give the same results, do.

Back to top profile pm

denis_berthier

Posted: Sun Oct 11, 2009 3:55 am Post subject: quote

Joined: 19 Jun 2007
Posts: 916
Location: Paris, France

Red Ed wrote:

> Denis, your comments address only experiments with minimal proper puzzles.

Yes, that's what I had in mind.

Back to top profile pm www

denis_berthier

Posted: Sun Oct 11, 2009 3:59 am Post subject: quote

Joined: 19 Jun 2007
Posts: 916
Location: Paris, France

Red Ed wrote:

> **Red Ed wrote:**
>
> > Now 27s:
> >
> > - 112136 minimals => std err ~= 1/335
> > - Scale up by choose(81,27) / (31 x 5472730538) to give number of proper minimal 27s per grid
> > - => you estimate 1.5244e15 w. std err 0.0046e15
> > - I had estimated 1.4364e15 w. std err 0.0433e15
> > - So **our agreement or not on the number of 27s is inconclusive**
> >
> > ...
> > We're 2 standard deviations apart, which is pretty serious but perhaps not conclusive.
>
> I'll let my re-run experiment go to 1000000 solution grids before stopping it. However, mid-term results from that experiment suggest that we were just unlucky to see the low estimate (1.4364e15) reported previously. It happens.

Good illustration of my previous post: you need large samples.

Back to top profile pm www

Red Ed

Posted: Sun Oct 11, 2009 4:05 am Post subject: quote

Joined: 06 Jun 2005
Posts: 784

denis_berthier wrote:

> Good illustration of my previous post: you need large samples.

Actually the "unlucky" bit was nothing to do with sample size, since the estimate of the standard error takes that into account. Whatever size sample you have,

you'll come up with results with low p-values some of the time. 100p% of the time, to be precise. This appears to have been just one of those times.

We've always known that if you want better precision then you need larger samples (example: for number-of-minimals estimates, the relative error of the estimator scales like 1/sqrt(N) where N is the number of trials; you do four times the work, you squeeze your error margins by a factor of two). But let's not forget that quick trials with low precision can often be of value.

Back to top profile pm

denis_berthier

Posted: Sun Oct 11, 2009 5:45 am Post subject: quote

Joined: 19 Jun 2007
Posts: 916
Location: Paris, France

Red Ed wrote:

> the "unlucky" bit was nothing to do with sample size, since the estimate of the standard error takes that into account. Whatever size sample you have, you'll come up with results with low p-values some of the time. 100p% of the time, to be precise. This appears to have been just one of those times.

OK, you've been unlucky with the 26s.
But that raises another question. You had fewer 27s than 26s.

denis_berthier wrote:

> **Red Ed wrote:**
>
> > Curious result re 26 vs. 27.
>
> Why curious?
> As the distribution is very close to normal with mean 26.575, it is on the contrary very consistent that 27, which is a little closer to the mean than 26, has a probability a little larger.

Now that you've increased your 26s estimate, you must have still fewer 27s in proportion. Have you been that unlucky also for the 27s?

Red Ed wrote:

> We've always known that if you want better precision then you need larger samples (example: for number-of-minimals estimates, the relative error of the estimator scales like 1/sqrt(N) where N is the number of trials; you do four times the work, you squeeze your error margins by a factor of two).

Sure, but that's not what I meant. The controlled-bias generator still has a very strong bias. We know it precisely and can correct it but it is strong. We need large samples in order to have sufficiently many 29s, 30s, 31s to be able to apply the correction factors. The 1/sqrt(N) is very general and indicates a low return on investment in these matters.

Back to top profile pm www

Red Ed

Posted: Sun Oct 11, 2009 6:35 am Post subject: quote

denis_berthier wrote:

> **denis_berthier wrote:**
>
> > **Red Ed wrote:**
> >
> > > Curious result re 26 vs. 27.
> >
> > Why curious?
> > As the distribution is very close to normal with mean 26.575, it is on the contrary very consistent that 27, which is a little closer to the mean than 26, has a probability a little larger.
>
> Now that you've increased your 26s estimate, you must have still fewer 27s in proportion. Have you been that unlucky also for the 27s?

Joined: 06 Jun 2005
Posts: 784

I think we have crossed wires; my 26s estimate is fine (or we're both wrong!) and it's the 27s that I'm retrying. Those are turning out fine, too, with now more 27s than 26s -- as expected given your long trial on the unbiased source.

Back to top profile pm

PIsaacson

Posted: Sun Oct 11, 2009 10:32 am Post subject: quote

denis_berthier wrote:

> **Paul**, do you have any news of the other, faster solver?

Joined: 02 Jul 2008
Posts: 214
Location: Campbell, CA

Yes! I received an e-mail from Brian in which he stated that he had decided on a simple "free to use for personal use..." statement and that he would release/repost his latest code this weeked. I'll check Monday morning and if the code is available, I'll perform my library mods and post it along with my mods on how to use it in place of the suexg solve function. There have been many patches posted for the suexg-cb code, so I think Eleven should be the coordinator for the final version.

Cheers,
Paul

Back to top profile pm

Red Ed

Posted: Sun Oct 11, 2009 11:11 am Post subject: quote

Joined: 06 Jun 2005
Posts: 784

I've done as much experimentation with 27s as I want now, and I've got an estimate very close to Denis' one:

Code:

```
Number of solution grids: 1008902
Number of 31-clue subgrids containing a 27-clue
minimal puzzle: 5644

Total number of 27-clue minimal puzzles in those 5644
31-clue subgrids:
  +----+---------+------------+------------+
  | Cl |   Count | E(nr/grid) | E(std dev) |
  +----+---------+------------+------------+
```

```
| 27 |    21061 |   1.53e+015 |   2.97e+013 |
+----+----------+-------------+-------------+
```

So nothing to worry about after all: the methods would seem to be in agreement.

Back to top profile pm

ericdeko411

Posted: Mon Oct 12, 2009 5:25 am Post subject: quote

Joined: 12 Oct 2009
Posts: 1

> **eleven wrote:**
>
> > **Red Ed wrote:**
> >
> > > Initial indications are that the supersets method might be able to do one 24, and (mostly implicitly) all its 31-clue supersets, every five seconds or so. If debugging and testing bear that out, it would be good news.
>
> Using a singles solver only i could check about 33% very quickly: In almost 10% of random 24's one of the clues is forced by the others through singles. And in 23% more than 6 singles are forced by the givens.
> Also the about 29% puzzles, where 4-6 singles are forced, could be checked brute force rather quickly.
> But the rest i would have to leave to Red Ed

I concur, also checked through brute force and arrived at almost the same result.

Eric Deko

Back to top profile pm www

Red Ed

Posted: Mon Oct 12, 2009 10:47 am Post subject: quote

Joined: 06 Jun 2005
Posts: 784

> **Red Ed wrote:**
>
> > Initial indications are that the supersets method might be able to do one 24, and (mostly implicitly) all its 31-clue supersets, every five seconds or so. If debugging and testing bear that out, it would be good news.

Well I think it's debugged and working properly now, but the code is pretty horrible so it's difficult to be sure.

I can do about 2000 24s (and - mostly implicitly - all of their 31-clue supersets) an hour on a single 1.4GHz CPU, which is vastly quicker than has been achieved by any other method. I've only just set off a long run now; when I check again in the morning I'll be able to analyse the variance of the estimator and produce a number-of-gsf-collections-per-hour equivalent figure to make it clear to everyone just how quick this is.

Back to top profile pm

Display posts from previous: All Posts Oldest First Go

newtopic postreply **Sudoku Players' Forums Forum Index -> General/puzzle**

All times are GMT - 8 Hours

Goto page Previous 1, 2, 3 ... , 34, 35, 36 Next

Page 35 of 36

Jump to: General/puzzle Go

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