数独 on Excel_J を楽しむ
－J Sudoku でどうやって数独の問題を解くか－

西川　利男

３．数独 on Excel_J　で楽しむ

数独パズルが、まだ根強く人気を保っている。３大新聞には、毎日あきもせず連載されている。数独が出だした頃、マイ・ワイフが相当凝っていたが、やめてしまった。ところが、絵の方がうまくいかないのであろうか、最近また毎日のようにやっている。

そのワイフがいわく、初級の問題は最近はすぐ解けるが、中級となるとなかなか解けない。けれども、コンピュータで解いてしまっては、面白くないのでいやだという。

J の仲間のあいだでは、有名な Hui のプログラムにより難なく解ける。そしてさらに、数独 on Excel_J では、紙とエンピツではなく楽しみつつ解くこともできる。[4]

数独 on Excel/J では問題を入力した後、つぎのようにして解が求められる。

CTRL-ℓ …… 数独データをロードする(Excel 画面と J_Sudoku システムへ)
CTRL-s …… J_Sudoku で解を求める
CTRL-r …… 結果を Excel で表示する

もちろん、ふつうのとおりに入力して楽しむこともできる。(CTRL-m でお手伝い)

次は面白くないという数独 on Excel_J による回答である。

	A	B	C	D	E	F	G	H	I
11	3	5	8	1	6	7	2	9	4
12	7	9	1	4	5	2	3	6	8
13	4	2	6	8	9	3	5	7	1
14	9	1	5	6	2	8	4	3	7
15	8	3	7	9	1	4	6	5	2
16	2	6	4	3	7	5	1	8	9
17	5	4	2	7	3	9	8	1	6
18	1	8	9	5	4	6	7	2	3
19	6	7	3	2	8	1	9	4	5

数独には数年前、私自身もかなり凝って、Hui の J のプログラム sudoku の解析を、Lab システムを用いて行ったが[1]、必ずしも満足いくものではなかった。
Hui の J プログラムを理解するには、これをそのまま、走らせる前に、紙とエンピツで、そのアルゴリズムを追うことが、極めて有益である。

今回、sudoku プログラムの処理をあらためて見直し、筆者なりにいくつかの便利な関数を追加した。また sudoku Lab システムの説明をわかり易くしたり、一部修正、改定をおこなった。
なお、最初の部分の説明は筆者のワイフ、靖子流の紙とエンピツによる数独攻略法である。

[1] 西川利男「数独（SUDOKU）パズルを J で解く－Labs システムによる Hui のプログラムのトレース－」JAPLA 研究会資料　2006/1/28
[2] 西川利男「J のオブジェクト指向プログラミング－その４－J-Grid による数独パズルをもっと使いやすく－」J 研究会資料　2006/3/25
[3] 西川利男「J の OOP と Grid プログラミング－その５－J による数独パズル－棋譜の自動記録と棋譜データによる実行」JAPLA 究会資料　2006/6/24

[4] 西川利男「Excel から J を使う－数独 on Excel_J への応用」JAPLA 研究会資料 2010/9/25
毎日新聞 10/26 掲載の問題

```
   A0 =: main_1026
   see A0
+---+---+---+
|.5.|...|.94|
|..1|..2|..8|
|...|8..|5..|
+---+---+---+
|..5|.2.|.3.|
|8..|9.4|..2|
|.6.|.7.|1..|
+---+---+---+
|..2|..9|...|
|1..|5..|7..|
|67.|...|.4.|
+---+---+---+
```

靖子流攻略法ではまず、問題の中で何回も表れる数字から手をつける。

```
   AA =: (0&~: # ]) A0  NB. extract nonzero numbers
   AA
5 9 4 1 2 8 8 5 5 2 3 8 9 4 2 6 7 1 2 9 1 5 7 6 7 4
   NU =: >: i.9
   FR =: +/"1 NU = / AA  NB. frequency of numbers
   NU,:FR
1 2 3 4 5 6 7 8 9
3 4 1 3 4 2 3 3 3
```

つまり、数字1は 3 回, 数字2 は4回, 数字3 は 1回, .. 現れる。

以上を参考にまず、左上ブロックで数字8を入れることをやってみる。
２行目、３行目には数字８がある。
１列目には、数字８がある。
それならば、数字８が１行３列に入ることが、すぐわかる。

同様にして、左中ブロックで数字2を入れることをやってみる。
４行目、５行目には数字２がある。
３列目には、数字２がある。
それならば、数字２が６行１列に入ることが、すぐわかる。

同様にして、左下ブロックで数字5を入れることをやってみる。
８行目には数字５がある。
２列目と３列目には、数字５がある。
それならば、数字５が７行１列に入ることが、すぐわかる。

この攻略法だけで、新聞掲載の初級の問題はほとんど解けるということである。

コンピュータのプログラムでは、この処理は関数 ar により、すべてのブロックをまとめて行っている。

```
   see ar@free A0
+---+---+---+
|..8|...|...|
|...|.5.|...|
|.2.|...|...|
+---+---+---+
|...|...|4..|
|...|...|...|
|2..|..5|...|
+---+---+---+
|5..|7..|...|
|...|...|.2.|
|...|2..|..5|
+---+---+---+
```

これとは別の攻略法がある。行、列、ブロックを仔細に見ると、そこだけで、候補が１つにしぼられ、数字が決まる場合がある。

ところで、問題のセルを通じて、入れられる数字の候補の可能性の頻度を見てみる。

```
NB. check candidate frequency
   see +/"1 free A0
+---+---+---+
|3.4|434|3..|
|43.|45.|22.|
|545|.54|.44|
+---+---+---+
|33.|2.3|4.3|
|.22|.4.|13.|
|4.3|1.3|.22|
+---+---+---+
|33.|55.|344|
|.44|.43|.33|
|..3|333|4.4|
+---+---+---+
NB. 1 shows determined cell, (6, 4), (5, 7)
   A0 cand 6 4
3
   A0 cand 5 7
6
```

つまり、セル(6, 4)では数字は３に確定し、セル(5, 7)では数字は６に確定する。この処理は関数 ac で行う。なおここでは、セル(6, 4)などの表記は１オリジンである。

```
   see ac@free A0
+---+---+---+
|...|...|...|
|...|...|...|
|...|...|...|
+---+---+---+
|...|...|...|
|...|...|6..|
|...|3..|...|
+---+---+---+
|...|...|...|
|...|...|...|
|...|...|...|
+---+---+---+
```

さて今回、セルに入れられる候補の数字のすべてを出力する便利なツール candlist を開発した。ここで紹介しよう。Jのコーディングは後述。

```
     candlist A0
+-------------+-------------+-------------+
|+---+---+---+|+---+---+---+|+---+---+---+|
||2 3|   |3 6|||1 3|1 3|1 3|||2 3|   |   ||
||7  |   |7 8|||6 7|6  |6 7|||6  |   |   ||
|+---+---+---+|+---+---+---+|+---+---+---+|
||3 4|3 4|   |||3 4|3 4|   |||3 6|6 7|   ||
||7 9|9  |   |||6 7|5 6|   |||   |   |   ||
|+---+---+---+||   |9  |   ||+---+---+---+|
||2 3|2 3|3 4||+---+---+---+||   |1 2|1 3||
||4 7|4 9|6 7|||   |1 3|1 3|||   |6 7|6 7||
||9  |   |9  |||   |4 6|6 7||+---+---+---+|
|+---+---+---+||   |9  |   ||              |
|             |+---+---+---+|              |
+-------------+-------------+-------------+
|+---+---+---+|+---+---+---+|+---+---+---+|
||4 7|1 4|   |||1 6|   |1 6|||4 6|   |6 7||
||9  |9  |   |||   |   |8  |||8 9|   |9  ||
|+---+---+---+|+---+---+---+|+---+---+---+|
||   |1 3|3 7|||   |1 3|   |||6  |5 6|   ||
||   |   |   |||   |5 6|   |||   |7  |   ||
|+---+---+---+|+---+---+---+|+---+---+---+|
||2 3|   |3 4|||3  |   |3 5|||   |5 8|5 9||
||4 9|   |9  |||   |   |8  |||   |   |   ||
|+---+---+---+|+---+---+---+|+---+---+---+|
+-------------+-------------+-------------+
|+---+---+---+|+---+---+---+|+---+---+---+|
||3 4|3 4|   |||1 3|1 3|   |||3 6|1 5|1 3||
```

```
||5  |8  |   |||4 6|4 6|   |||8  |6 8|5 6||
|+---+---+---+||7  |8  |   ||+---+---+---+|
||   |3 4|3 4||+---+---+---+||   |2 6|3 6||
||   |8 9|8 9|||   |3 4|3 6|||   |8  |9  ||
|+---+---+---+||   |6 8|8  ||+---+---+---+|
||   |   |3 8||+---+---+---+||2 3|   |1 3||
||   |   |9  |||1 2|1 3|1 3||8 9|   |5 9||
|+---+---+---+||3  |8  |8  ||+---+---+---+|
|             |+---+---+---+|             |
+-------------+-------------+-------------+
```

なお、すでに決まったが(1, 3)セルは

```
 A0 cand 1 3
3 6 7 8
```

のように4通りもある。

Hui のプログラムでは、これを決定するには次のように行う。

```
 A0 cand1 1 3
Row:   1 1 1 2 2 3 3 3 3 3 3 6 6 6 6 6 7 7 7 7 8
Column:3 3 3 3 3 3 4 4 4 6 6 7 7 7 8 8 8 9 9 9 9
Block: 2 2 2 3 3 3 3 3 3 3 4 4 4 4 4 6 6 7 7 7 7 7 8 9 9 9 9 9 9
```

この表は、次のことを表す。

1行目のセルに注目すると、1は3回、2は2回、...、8は1回現れる。

3列目のセルに注目すると、3は6回、4は3回、...、8は3回現れる。

同じブロックでは、2は3回、3は7回、...、8は1回現れる。

これから、1回だけ現れるのは8しかない。そこで8に決まる！

以上、free で入れられる場所を求めて、これに関数 ac と ar とにより定まる候補をまとめて行う関数

```
func =: (ac >. ar)@free
```

を用いて、最初のステップが決まる。

```
NB. operate func =: (ac >. ar)@free
NB. first step determined


 A1 =: func A0
 see A1
+---+---+---+
|.58|...|.94|
|..1|.52|..8|
|.2.|8..|5..|
+---+---+---+
|..5|.2.|43.|
|8..|9.4|6.2|
```

```
|26.|375|1..|
+---+---+---+
|5.2|7.9|...|
|1..|5..|72.|
|67.|2..|.45|
+---+---+---+
```

このステップは次のように続けられる。

```
   see ar@free A1
+---+---+---+
|...|...|2..|
|...|4..|...|
|..6|.9.|...|
+---+---+---+
|...|..8|...|
|...|...|.5.|
|..4|...|...|
+---+---+---+
|...|...|...|
|...|...|...|
|...|...|...|
+---+---+---+
   see ac@free A1
+---+---+---+
|...|...|...|
|...|...|3..|
|...|...|...|
+---+---+---+
|...|...|...|
|...|.1.|...|
|...|...|.89|
+---+---+---+
|...|...|...|
|...|...|...|
|...|...|...|
+---+---+---+

   A2 =: func A1
   see A2
+---+---+---+
|.58|...|294|
|..1|452|3.8|
|.26|89.|5..|
+---+---+---+
|..5|.28|43.|
|8..|914|652|
```

```
|264|375|189|
+---+---+---+
|5.2|7.9|...|
|1..|5..|72.|
|67.|2..|.45|
+---+---+---+
```

```
   A3 =: func A2
   see A3
+---+---+---+
|.58|1..|294|
|.91|452|368|
|426|89.|5..|
+---+---+---+
|.15|628|437|
|837|914|652|
|264|375|189|
+---+---+---+
|5.2|7.9|8..|
|1..|5..|72.|
|67.|2.1|945|
+---+---+---+
```

```
   A4 =: func A3
   see A4
+---+---+---+
|358|1..|294|
|791|452|368|
|426|893|571|
+---+---+---+
|915|628|437|
|837|914|652|
|264|375|189|
+---+---+---+
|542|7.9|81.|
|189|5..|72.|
|673|281|945|
+---+---+---+
```

これらをまとめて表示する。

```
   see func^:(i.5) A0
+-------------+-------------+-------------+-------------+-------------+
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
||.5.|...|.94|||.58|...|.94|||.58|...|294|||.58|1..|294|||358|1..|294||
||..1|..2|..8|||..1|.52|..8|||..1|452|3.8|||.91|452|368|||791|452|368||
||...|8..|5..|||.2.|8..|5..|||.26|89.|5..|||426|89.|5..|||426|893|571||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
```

```
||..5|.2.|.3.|||..5|.2.|43.|||..5|.28|43.|||.15|628|437|||915|628|437||
||8..|9.4|..2|||8..|9.4|6.2|||8..|914|652|||837|914|652|||837|914|652||
||.6.|.7.|1..|||26.|375|1..|||264|375|189|||264|375|189|||264|375|189||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
||..2|..9|...|||5.2|7.9|...|||5.2|7.9|...|||5.2|7.9|8..|||542|7.9|81.||
||1..|5..|7..|||1..|5..|72.|||1..|5..|72.|||1..|5..|72.|||189|5..|72.||
||67.|...|.4.|||67.|2..|.45||67.|2..|.45||67.|2.1|945|||673|281|945||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
+-------------+-------------+-------------+-------------+-------------+
```

また、それぞれのステップで、入れられた数字はつぎのようになる。

see diff funcˆ:(i.5) A0

```
+-------------+-------------+-------------+-------------+-------------+
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
||.5.|...|.94|||..8|...|...|||...|...|2..|||...|1..|...|||3..|...|...||
||..1|..2|..8|||...|.5.|...|||...|4..|3..|||.9.|...|.6.|||7..|...|...||
||...|8..|5..|||.2.|...|...|||..6|.9.|...|||4..|...|...|||...|..3|.71||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
||..5|.2.|.3.|||...|...|4..|||...|..8|...|||.1.|6..|..7|||9..|...|...||
||8..|9.4|..2|||...|...|6..|||...|.1.|.5.|||.37|...|...|||...|...|...||
||.6.|.7.|1..|||2..|3.5|...|||..4|...|.89|||...|...|...|||...|...|...||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
||..2|..9|...|||5..|7..|...|||...|...|...|||...|...|8..|||.4.|...|.1.||
||1..|5..|7..|||...|...|.2.|||...|...|...|||...|...|...|||.89|...|...||
||67.|...|.4.|||...|2..|..5|||...|...|...|||...|..1|9..|||..3|.8.|...||
|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|+---+---+---+|
+-------------+-------------+-------------+-------------+-------------+
```

最終結果は次のようになった。

see funcˆ:(6) A0

```
+---+---+---+
|358|167|294|
|791|452|368|
|426|893|571|
+---+---+---+
|915|628|437|
|837|914|652|
|264|375|189|
+---+---+---+
|542|739|816|
|189|546|723|
|673|281|945|
+---+---+---+
```

なお、Hui のプログラムでは、以上の処理では一意的には決まらないもっと難しい問題に対しては、それぞれの解の可能性のトリーを平行して解の検索を進める、いわゆる推論エンジン guess を用意している。

くわしくは、sudoku の解析の Lab システムをご覧いただきたい。