Minimum Richness Equilibrium and Sudoku

Jeffrey Horn
Northern Michigan University, jhorn@nmu.edu
Minimum Richness Equilibrium and Sudoku

Jeffrey Horn
Northern Michigan University
Department of Mathematics and Computer Science
Marquette, MI USA
jhorn@nmu.edu
http://cs.nmu.edu/~jeffhorn

MAA U.P. Regional Meeting
October 3, 2015
Lake Superior State University
Abstract

In the field of theoretical ecology the term "richness" refers to the number of species present in an ecosystem. By reducing the game of Sudoku to the problem of exact four cover ($X4C$), then reducing $X4C$ to minimum richness equilibrium (MRE), we show that MRE is in NP-complete. We further reduce MRE to minimum weight linear programming (MWLP) to arrive at a simple, polynomial-time decision process that we demonstrate to be a pretty darn good Sudoku solver!
**Background: Genetic Algorithms**

The simple GA:

- $p_x$ is the proportion of individual/species $x$ in the population.
- $f_x$ is the fitness of $x$.

Proportionate selection:

$$p_x(t+1) = p_x(t) \frac{f_x}{f}$$

Example:

$$p_A(t+1) = p_A(t) \frac{f_A}{p_A(t)f_A + p_B(t)f_B}$$
Background: The RFS Approach

The **SHARED FITNESS** \( f_{sh,x} \) of a species \( x \) depends, in a simple way, on competition from overlapping species:

\[
f_{sh,x} = \frac{1}{\text{niche\_count}(x)} = \frac{1}{\sum p_y f_{xy}}
\]

where \( p_y \) is the *proportion* of species \( y \) in the current population.

Example for two overlapping niches

\[
f_{sh,A} = \frac{f_A}{n_A f_A + n_B f_{AB}}
\]

Example for three overlapping niches

\[
f_{sh,A} = \frac{f_A}{n_A f_A + n_B f_{AB} + n_C f_{AC}}
\]

Finally, a **selection operator**, such as *proportionate selection*, uses the RFS Shared Fitnesses each generation.
• Shared Fitness:

\[ f_{sh}(x) = \left( \sum_{y \in S} p_y * f_{x,y} \right)^{-1} \]
Background: The RFS Approach


Substrate (stock material) is a finite RESOURCE to be COVERED by niches (defined by the species).

Each SPECIES covers a unique subset of the resources.

Overlapping species compete for the shared amount of resource.

E.g., Species $a$ and $b$ overlap in coverage by amount $f_{ab}$:
RFS for Shape Nesting
**Experiment 1**

**generation 1**
(one generation beyond the initial population)

- \( \text{species}_\text{count} \geq 4 \)
- \( \text{species}_\text{count} \geq 1 \)
  (shows all species)

**generation 209**
(8 cooperative species)

- \( \text{species}_\text{count} \geq 36 \)
- \( \text{species}_\text{count} \geq 2 \)
Experiment 1

generation 609
(almost 9 cooperative species)

species_count ≥ 20

→

generation 709
(9 cooperative species)

species_count ≥ 22
Experiment 2

generation: 4700

species_count ≥ 19
Co-evolutionary Shape Nesting can Outperform Commercial Software
Goal: Evolve an Exact Cover of Substrate by $K$ Species

Here $K = 16$. 
9x9 Sudoku

Insert the numerals 1-9 in each cell subject to FOUR constraints:

1. Only one numeral per cell
2. Exactly one of each numeral per row
3. Exactly one of each numeral per column
4. Exactly one of each numeral per region
Garey and Johnson (1979): X3C

[SP2] EXACT COVER BY 3-SETS (X3C)

INSTANCE: Set $X$ with $|X| = 3q$ and a collection $C$ of 3-element subsets of $X$.

QUESTION: Does $C$ contain an exact cover for $X$, i.e., a subcollection $C' \subseteq C$ such that every element of $X$ occurs in exactly one member of $C'$?

Reference: [Karp, 1972]. Transformation from 3DM.

Comment: Remains NP-complete if no element occurs in more than three subsets, but is solvable in polynomial time if no element occurs in more than two subsets [Garey and Johnson, ——]. Related EXACT COVER BY 2-SETS problem is also solvable in polynomial time by matching techniques.
9x9 Sudoku

Insert the numerals 1-9 in each cell subject to FOUR constraints:

1. Only one numeral per cell
2. Exactly one of each numeral per row
3. Exactly one of each numeral per column
4. Exactly one of each numeral per region
Sudoku as X4C

\[ f_{x,y} = 0 + \begin{cases} 
1/4 & \text{iff} & \text{SameCell}(x, y) \\
1/4 & \text{iff} & \text{SameNumeral}(x, y) \land \text{SameRow}(x, y) \\
1/4 & \text{iff} & \text{SameNumeral}(x, y) \land \text{SameColumn}(x, y) \\
1/4 & \text{iff} & \text{SameNumeral}(x, y) \land \text{SameRegion}(x, y) 
\end{cases} \text{(cumulative)}, \]
The RFSS-Evolve Algorithm

1. INITIALIZE:
   (a) Generate initial set of species (unique chromosomes) $S$;
   (b) $\forall x, y \in S$: calculate pairwise intersection\(^3\) and store as $f_{x,y} := \frac{|x \cap y|}{|x|}$;
   (c) $\forall s \in S : p_s := \frac{1}{|S|} ;// $ Uniform distribution across all species, initially.

2. LOOP: while ( termination condition\(^4\) is false ) do
2. LOOP: while (termination condition⁴ is false) do

(a) \( \forall x \in S : \text{Evaluate and store shared fitnesses as } f_{sh}(x) := \left(\sum_{\forall y \in S} p_y * f_{x,y}\right)^{-1} ; \)

(b) Calculate and store average shared fitness as \( \overline{f}_{sh} := \sum_{\forall x \in S} (p_x * f_{sh}(x)) ; \)

(c) Calculate next generation species proportions \( p' \) as \( \forall x \in S : p'_x := p_u * \frac{f_{sh}(x)}{\overline{f}_{sh}} ; \)

// This implements proportionate selection, using shared fitnesses.

(d) Move to next generation by updating species proportions as \( \forall x \in S : p_x := p'_x ; \)
### Table 2: Raw Results from Runs on USA Today 200 Sudoku Puzzle Book

<table>
<thead>
<tr>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EASY</td>
<td>MEDIUM</td>
<td>HARD</td>
</tr>
<tr>
<td>1</td>
<td>580</td>
<td>432</td>
<td>2622</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>2094</td>
<td>3596</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>365</td>
<td>2028</td>
</tr>
<tr>
<td>4</td>
<td>323</td>
<td>326</td>
<td>1987</td>
</tr>
<tr>
<td>5</td>
<td>254</td>
<td>317</td>
<td>2059</td>
</tr>
<tr>
<td>6</td>
<td>521</td>
<td>706</td>
<td>Not Solved</td>
</tr>
<tr>
<td>7</td>
<td>558</td>
<td>728</td>
<td>685</td>
</tr>
<tr>
<td>8</td>
<td>342</td>
<td>1103</td>
<td>3341</td>
</tr>
<tr>
<td>9</td>
<td>134</td>
<td>905</td>
<td>Not Solved</td>
</tr>
<tr>
<td>10</td>
<td>285</td>
<td>361</td>
<td>Not Solved</td>
</tr>
<tr>
<td>11</td>
<td>79</td>
<td>766</td>
<td>911</td>
</tr>
<tr>
<td>12</td>
<td>72</td>
<td>141</td>
<td>859</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>229</td>
<td>2611</td>
</tr>
<tr>
<td>14</td>
<td>245</td>
<td>2829</td>
<td>Not Solved</td>
</tr>
<tr>
<td>15</td>
<td>344</td>
<td>922</td>
<td>3640</td>
</tr>
<tr>
<td>16</td>
<td>356</td>
<td>280</td>
<td>Not Solved</td>
</tr>
<tr>
<td>17</td>
<td>140</td>
<td>425</td>
<td>2397</td>
</tr>
<tr>
<td>18</td>
<td>447</td>
<td>606</td>
<td>1044</td>
</tr>
<tr>
<td>19</td>
<td>513</td>
<td>489</td>
<td>775</td>
</tr>
<tr>
<td>20</td>
<td>242</td>
<td>279</td>
<td>392</td>
</tr>
<tr>
<td>21</td>
<td>638</td>
<td>634</td>
<td>3630</td>
</tr>
<tr>
<td>22</td>
<td>255</td>
<td>435</td>
<td>1066</td>
</tr>
<tr>
<td>23</td>
<td>314</td>
<td>125</td>
<td>826</td>
</tr>
</tbody>
</table>
Species Interaction Matrix

\[
\mathbf{M}_{RFS} = \text{Matrix of pairwise overlaps between species}
\]

Under RFS:

\[
\mathbf{M}_{RFS} \ast \mathbf{p} = \text{niche_count}
\]

vector of species proportions

vector of species niche counts

\[
\begin{bmatrix}
\frac{f_{E_1,E_1}}{f_{E_1,E_1}} & \frac{f_{E_1,E_2}}{f_{E_1,E_1}} & \cdots & \frac{f_{E_1,E_K}}{f_{E_1,E_1}} \\
\frac{f_{E_2,E_1}}{f_{E_2,E_1}} & \frac{f_{E_2,E_2}}{f_{E_2,E_1}} & \cdots & \frac{f_{E_2,E_K}}{f_{E_2,E_1}} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{f_{E_K,E_1}}{f_{E_K,E_1}} & \frac{f_{E_K,E_2}}{f_{E_K,E_1}} & \cdots & \frac{f_{E_K,E_K}}{f_{E_K,E_1}} \\
\frac{f_{E_1,C_1}}{f_{E_1,C_1}} & \frac{f_{E_1,C_2}}{f_{E_1,C_1}} & \cdots & \frac{f_{E_1,C_H}}{f_{E_1,C_1}} \\
\frac{f_{E_2,C_1}}{f_{E_2,C_1}} & \frac{f_{E_2,C_2}}{f_{E_2,C_1}} & \cdots & \frac{f_{E_2,C_H}}{f_{E_2,C_1}} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{f_{E_K,C_1}}{f_{E_K,C_1}} & \frac{f_{E_K,C_2}}{f_{E_K,C_1}} & \cdots & \frac{f_{E_K,C_H}}{f_{E_K,C_1}} \\
\frac{f_{E_1,C_H}}{f_{E_1,C_H}} & \frac{f_{E_2,C_H}}{f_{E_1,C_H}} & \cdots & \frac{f_{E_K,C_H}}{f_{E_1,C_H}} \\
\frac{f_{C_1,C_1}}{f_{C_1,C_1}} & \frac{f_{C_1,C_2}}{f_{C_1,C_1}} & \cdots & \frac{f_{C_1,C_H}}{f_{C_1,C_1}} \\
\frac{f_{C_2,C_1}}{f_{C_2,C_1}} & \frac{f_{C_2,C_2}}{f_{C_2,C_1}} & \cdots & \frac{f_{C_2,C_H}}{f_{C_2,C_1}} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{f_{C_H,C_1}}{f_{C_H,C_1}} & \frac{f_{C_H,C_2}}{f_{C_H,C_1}} & \cdots & \frac{f_{C_H,C_H}}{f_{C_H,C_1}}
\end{bmatrix}

\begin{bmatrix}
p_{E_1} \\
p_{E_2} \\
p_{E_K} \\
p_{C_1} \\
p_{C_2} \\
p_{C_H}
\end{bmatrix}
= \begin{bmatrix}
nicte_count(E_1) \\
nicte_count(E_1) \\
nicte_count(E_K) \\
nicte_count(C_1) \\
nicte_count(C_2) \\
nicte_count(C_H)
\end{bmatrix}
Notation

Modified Notation:

\[ f_{E_i E_j} \]
\[ p_{E_2} \]
≡ overlap between species \( E_i \) and species \( E_j \)
≡ proportion of population occupied by species \( E_2 \)

Property I

\[ f_{E_i E_j} = f_{E_j E_i} = 0, \text{ for } i \neq j \]

Property II

\[ \forall i \in (1..k) \sum_{j=1}^{h} f_{E_j C_i} = 1 \]

Exact Cover
species do not compete!
(no overlaps)

Exact Cover
species completely cover all other species
Species Interaction Matrix

Note that on main diagonal of MRFS: \( f_{xx} = 1 \)

By Property I:

\[
f_{E_i E_j} = f_{E_j E_i} = 0, \text{ for } i \neq j
\]
Species Interaction Matrix

\[
\begin{bmatrix}
1 & 0 & \ldots & 0 \\
0 & 1 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 1 \\
\end{bmatrix}
\begin{bmatrix}
f_{E_1, C_1} & f_{E_1, C_2} & \cdots & f_{E_1, C_H} \\
f_{E_2, C_1} & f_{E_2, C_2} & \cdots & f_{E_2, C_H} \\
\vdots & \vdots & \ddots & \vdots \\
f_{E_K, C_1} & f_{E_K, C_2} & \cdots & f_{E_K, C_H} \\
\end{bmatrix}
\begin{bmatrix}
p_{E_1} \\
p_{E_2} \\
\vdots \\
p_{E_K} \\
\end{bmatrix}
= 
\begin{bmatrix}
C' \\
C' \\
\vdots \\
C' \\
\end{bmatrix}
\]
Species Interaction Matrix

\[
\begin{bmatrix}
1 & 0 & \ldots & 0 \\
0 & 1 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 1 \\
\end{bmatrix}
\begin{bmatrix}
f_{E_1,C_1} & f_{E_1,C_2} & \ldots & f_{E_1,C_H} \\
f_{E_2,C_1} & f_{E_2,C_2} & \ldots & f_{E_2,C_H} \\
\vdots & \vdots & \ddots & \vdots \\
f_{E_K,C_1} & f_{E_K,C_2} & \ldots & f_{E_K,C_H} \\
\end{bmatrix}
\begin{bmatrix}
P_{E_1} \\
P_{E_2} \\
\vdots \\
P_{E_K} \\
\end{bmatrix}
= 
\begin{bmatrix}
1/K \\
1/K \\
\vdots \\
1/K \\
\end{bmatrix}
\]
### Species Interaction Matrix

\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1 \\
\end{bmatrix}
\begin{bmatrix}
f_{E_1,C_1} & f_{E_1,C_2} & \cdots & f_{E_1,C_H} \\
f_{E_2,C_1} & f_{E_2,C_2} & \cdots & f_{E_2,C_H} \\
\vdots & \vdots & \ddots & \vdots \\
f_{E_K,C_1} & f_{E_K,C_2} & \cdots & f_{E_K,C_H} \\
\end{bmatrix}
\begin{bmatrix}
K \cdot p_{E_1} \\
K \cdot p_{E_2} \\
\vdots \\
K \cdot p_{E_K} \\
\end{bmatrix}
= 
\begin{bmatrix}
1 \\
1 \\
\vdots \\
1 \\
\end{bmatrix}
\]
**XCSSS solution → MWPSLE solution**

- Assume exact cover solution of $K$ subsets.
- The vector $\mathbf{y}$ below solves the MWPSLE.

\[
\begin{bmatrix}
1 & 0 & \ldots & 0 \\
0 & 1 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 1
\end{bmatrix}
\begin{bmatrix}
E_1,c_1 \\
E_2,c_1 \\
E_3,c_1 \\
\vdots \\
E_K,c_1
\end{bmatrix}
\begin{bmatrix}
1 \\
0 \\
\vdots \\
0
\end{bmatrix}
= 
\begin{bmatrix}
1 \\
1 \\
\vdots \\
1
\end{bmatrix}
\]
XCSSS solution $\leftrightarrow$ MWPSLE solution

• Assume a non-negative vector $y$ such that
  \[ M_{RFS} \cdot y = 1 \]

• Also assume that $y$ has exactly $K$ positive components.

\[
\begin{bmatrix}
1 & a_{1,2} & \ldots & a_{1,|C|} \\
a_{2,1} & 1 & \ldots & a_{2,|C|} \\
\vdots & \vdots & \ddots & \vdots \\
a_{|C|,1} & a_{|C|,2} & \ldots & 1
\end{bmatrix}
\begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_{|C|}
\end{bmatrix}
= 
\begin{bmatrix}
1 \\
1 \\
\vdots \\
1
\end{bmatrix}
\]

• First we show all positive components of $y_i$ must be 1:
  – Since all matrix coefficients $a_{ii} = 1$, and since all $y_i$ and $a_{ii}$ are non-negative, it follows that no $y_i$ can exceed 1: $\forall i : y_i \leq 1$
  – Now since $\sum y_i = K$ then $\forall i : y_i = \{0, 1\}$ with exactly $K$ components equal to one, and all others equal to zero.
• Next we show for any two distinct positive solution components $y_i$ and $y_j$ ($i \neq j$), the corresponding $a_{ij}$ must be zero:

$$\forall i, j : (y_i = y_j = 1 \land i \neq j) \Rightarrow a_{i,j} = 0$$

– Proof by contradiction: If $a_{ij}$ were not zero, then it would be positive and would add to the left hand side of both equations $i$ and $j$. But since both $y_i$ and $y_j$ equal one, the left hand sides of equations $i$ and $j$ will then exceed one. Thus equations $i$ and $j$ will not be satisfied.
## MENDEL 2014 results

Table 2: Raw Results from Runs on USA Today 200 Sudoku Puzzle Book

<table>
<thead>
<tr>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EASY</td>
<td>MEDIUM</td>
<td>HARD</td>
</tr>
<tr>
<td>1</td>
<td>580</td>
<td>432</td>
<td>2622</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>2094</td>
<td>3596</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>365</td>
<td>2028</td>
</tr>
<tr>
<td>4</td>
<td>323</td>
<td>326</td>
<td>1987</td>
</tr>
<tr>
<td>5</td>
<td>254</td>
<td>317</td>
<td>2059</td>
</tr>
<tr>
<td>6</td>
<td>521</td>
<td>706</td>
<td>Not Solved</td>
</tr>
<tr>
<td>7</td>
<td>558</td>
<td>728</td>
<td>685</td>
</tr>
<tr>
<td>8</td>
<td>342</td>
<td>1103</td>
<td>3341</td>
</tr>
<tr>
<td>9</td>
<td>134</td>
<td>905</td>
<td>Not Solved</td>
</tr>
<tr>
<td>10</td>
<td>285</td>
<td>361</td>
<td>Not Solved</td>
</tr>
<tr>
<td>11</td>
<td>79</td>
<td>766</td>
<td>911</td>
</tr>
<tr>
<td>12</td>
<td>72</td>
<td>141</td>
<td>859</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>229</td>
<td>2611</td>
</tr>
<tr>
<td>14</td>
<td>245</td>
<td>2829</td>
<td>Not Solved</td>
</tr>
<tr>
<td>15</td>
<td>344</td>
<td>922</td>
<td>3640</td>
</tr>
<tr>
<td>16</td>
<td>356</td>
<td>280</td>
<td>Not Solved</td>
</tr>
<tr>
<td>17</td>
<td>140</td>
<td>425</td>
<td>2397</td>
</tr>
<tr>
<td>18</td>
<td>447</td>
<td>606</td>
<td>1044</td>
</tr>
<tr>
<td>19</td>
<td>513</td>
<td>489</td>
<td>775</td>
</tr>
<tr>
<td>20</td>
<td>242</td>
<td>279</td>
<td>392</td>
</tr>
<tr>
<td>21</td>
<td>638</td>
<td>634</td>
<td>3630</td>
</tr>
<tr>
<td>22</td>
<td>255</td>
<td>435</td>
<td>1066</td>
</tr>
<tr>
<td>23</td>
<td>314</td>
<td>125</td>
<td>826</td>
</tr>
</tbody>
</table>
Table 2: Raw Results from Runs on USA Today 200 Sudoku Puzzle Book

<table>
<thead>
<tr>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>EASY</td>
<td>MEDIUM</td>
<td>HARD</td>
<td>EASY</td>
</tr>
<tr>
<td>1</td>
<td>580</td>
<td>432</td>
<td>2622</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>2094</td>
<td>3596</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>365</td>
<td>2928</td>
</tr>
<tr>
<td>4</td>
<td>323</td>
<td>326</td>
<td>1987</td>
</tr>
<tr>
<td>5</td>
<td>254</td>
<td>317</td>
<td>2959</td>
</tr>
<tr>
<td>6</td>
<td>521</td>
<td>766</td>
<td>Not Solved</td>
</tr>
<tr>
<td>7</td>
<td>558</td>
<td>786</td>
<td>685</td>
</tr>
<tr>
<td>8</td>
<td>342</td>
<td>1103</td>
<td>3341</td>
</tr>
<tr>
<td>9</td>
<td>134</td>
<td>965</td>
<td>Not Solved</td>
</tr>
<tr>
<td>10</td>
<td>285</td>
<td>361</td>
<td>Not Solved</td>
</tr>
<tr>
<td>11</td>
<td>79</td>
<td>766</td>
<td>911</td>
</tr>
<tr>
<td>12</td>
<td>72</td>
<td>141</td>
<td>859</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>229</td>
<td>2811</td>
</tr>
<tr>
<td>14</td>
<td>245</td>
<td>2820</td>
<td>Not Solved</td>
</tr>
<tr>
<td>15</td>
<td>344</td>
<td>922</td>
<td>3640</td>
</tr>
<tr>
<td>16</td>
<td>356</td>
<td>280</td>
<td>Not Solved</td>
</tr>
<tr>
<td>17</td>
<td>140</td>
<td>425</td>
<td>2397</td>
</tr>
<tr>
<td>18</td>
<td>447</td>
<td>696</td>
<td>1944</td>
</tr>
<tr>
<td>19</td>
<td>513</td>
<td>489</td>
<td>775</td>
</tr>
<tr>
<td>20</td>
<td>242</td>
<td>279</td>
<td>392</td>
</tr>
<tr>
<td>21</td>
<td>638</td>
<td>634</td>
<td>3630</td>
</tr>
<tr>
<td>22</td>
<td>255</td>
<td>435</td>
<td>1066</td>
</tr>
<tr>
<td>23</td>
<td>314</td>
<td>125</td>
<td>826</td>
</tr>
<tr>
<td>24</td>
<td>260</td>
<td>326</td>
<td>1480</td>
</tr>
<tr>
<td>25</td>
<td>295</td>
<td>857</td>
<td>4555</td>
</tr>
<tr>
<td>26</td>
<td>107</td>
<td>694</td>
<td>669</td>
</tr>
<tr>
<td>27</td>
<td>57</td>
<td>802</td>
<td>2211</td>
</tr>
<tr>
<td>28</td>
<td>126</td>
<td>416</td>
<td>1025</td>
</tr>
<tr>
<td>29</td>
<td>579</td>
<td>349</td>
<td>872</td>
</tr>
<tr>
<td>30</td>
<td>141</td>
<td>855</td>
<td>Not Solved</td>
</tr>
<tr>
<td>31</td>
<td>132</td>
<td>218</td>
<td>3317</td>
</tr>
<tr>
<td>32</td>
<td>128</td>
<td>2918</td>
<td>2356</td>
</tr>
<tr>
<td>33</td>
<td>238</td>
<td>792</td>
<td>583</td>
</tr>
<tr>
<td>34</td>
<td>511</td>
<td>275</td>
<td>1147</td>
</tr>
</tbody>
</table>
Table 1: Summarized Results of Runs on USA Today 200 Sudoku Puzzle Book

<table>
<thead>
<tr>
<th>Statistic</th>
<th>EASY</th>
<th>MEDIUM</th>
<th>HARD</th>
<th>OVERALL</th>
</tr>
</thead>
<tbody>
<tr>
<td># of puzzles solved</td>
<td>68</td>
<td>68</td>
<td>50</td>
<td>186</td>
</tr>
<tr>
<td>total # of puzzles</td>
<td>68</td>
<td>68</td>
<td>62</td>
<td>198</td>
</tr>
<tr>
<td>solution rate (%)</td>
<td>100%</td>
<td>100%</td>
<td>81%</td>
<td>94%</td>
</tr>
<tr>
<td>Mean time to solution (generations)</td>
<td>272.7</td>
<td>652.0</td>
<td>2039.2</td>
<td>988.0</td>
</tr>
<tr>
<td>(standard deviation)</td>
<td>(193.3)</td>
<td>(534.5)</td>
<td>(1778.6)</td>
<td>(1213.5)</td>
</tr>
</tbody>
</table>
Decision Process

START

Does LPXC report a solution?

no

\[ \exists \text{ XC} \]

yes

Is the solution an XC?

no

Is the Det of the Interaction Matrix singular?

no

\[ \exists \text{ XC} \]

yes

Is the solution larger than an XC?

no

\[ \exists \text{ XC} \]

yes

Are the overlaps among the members of the solution the same size?

no

?
**RFSS-LP (Linear Programming)**

<table>
<thead>
<tr>
<th>PUZZLE No.</th>
<th>DIFFICULTY LEVEL</th>
<th>PUZZLE No.</th>
<th>DIFFICULTY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EASY</td>
<td>MEDIUM</td>
<td>HARD</td>
</tr>
<tr>
<td>1</td>
<td>580</td>
<td>432</td>
<td>2622</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>2091</td>
<td>3509</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>365</td>
<td>2028</td>
</tr>
<tr>
<td>4</td>
<td>323</td>
<td>326</td>
<td>1987</td>
</tr>
<tr>
<td>5</td>
<td>254</td>
<td>317</td>
<td>2059</td>
</tr>
<tr>
<td>6</td>
<td>521</td>
<td>706</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>7</td>
<td>558</td>
<td>728</td>
<td>685</td>
</tr>
<tr>
<td>8</td>
<td>342</td>
<td>1103</td>
<td>3341</td>
</tr>
<tr>
<td>9</td>
<td>134</td>
<td>905</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>10</td>
<td>285</td>
<td>361</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>11</td>
<td>79</td>
<td>766</td>
<td>911</td>
</tr>
<tr>
<td>12</td>
<td>72</td>
<td>141</td>
<td>859</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>229</td>
<td>2611</td>
</tr>
<tr>
<td>14</td>
<td>245</td>
<td>2829</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>15</td>
<td>344</td>
<td>922</td>
<td>3640</td>
</tr>
<tr>
<td>16</td>
<td>356</td>
<td>280</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>17</td>
<td>140</td>
<td>425</td>
<td>2397</td>
</tr>
<tr>
<td>18</td>
<td>447</td>
<td>696</td>
<td>1044</td>
</tr>
<tr>
<td>19</td>
<td>513</td>
<td>489</td>
<td>775</td>
</tr>
<tr>
<td>20</td>
<td>242</td>
<td>279</td>
<td>392</td>
</tr>
<tr>
<td>21</td>
<td>638</td>
<td>634</td>
<td>3630</td>
</tr>
<tr>
<td>22</td>
<td>255</td>
<td>435</td>
<td>1066</td>
</tr>
<tr>
<td>23</td>
<td>314</td>
<td>125</td>
<td>826</td>
</tr>
<tr>
<td>24</td>
<td>269</td>
<td>326</td>
<td>1480</td>
</tr>
<tr>
<td>25</td>
<td>295</td>
<td>897</td>
<td>4555</td>
</tr>
<tr>
<td>26</td>
<td>107</td>
<td>694</td>
<td>669</td>
</tr>
<tr>
<td>27</td>
<td>57</td>
<td>802</td>
<td>2211</td>
</tr>
<tr>
<td>28</td>
<td>126</td>
<td>416</td>
<td>1025</td>
</tr>
<tr>
<td>29</td>
<td>579</td>
<td>349</td>
<td>872</td>
</tr>
<tr>
<td>30</td>
<td>141</td>
<td>855</td>
<td>Not-Solved</td>
</tr>
<tr>
<td>31</td>
<td>132</td>
<td>218</td>
<td>3317</td>
</tr>
<tr>
<td>32</td>
<td>128</td>
<td>2918</td>
<td>2356</td>
</tr>
<tr>
<td>33</td>
<td>238</td>
<td>792</td>
<td>583</td>
</tr>
<tr>
<td>34</td>
<td>541</td>
<td>275</td>
<td>1147</td>
</tr>
</tbody>
</table>
RFSS-LP (INTEGER Programming)

Table 2: Raw Results from Runs on USA Today 200 Sudoku Puzzle Book

<table>
<thead>
<tr>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
<th>PUZZLE No. (within level)</th>
<th>DIFFICULTY LEVEL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EASY</td>
<td>MEDIUM</td>
<td>HARD</td>
</tr>
<tr>
<td>1</td>
<td>580</td>
<td>432</td>
<td>2622</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>2094</td>
<td>3596</td>
</tr>
<tr>
<td>3</td>
<td>90</td>
<td>365</td>
<td>2028</td>
</tr>
<tr>
<td>4</td>
<td>323</td>
<td>326</td>
<td>1987</td>
</tr>
<tr>
<td>5</td>
<td>254</td>
<td>317</td>
<td>2059</td>
</tr>
<tr>
<td>6</td>
<td>521</td>
<td>706</td>
<td>Not Solved</td>
</tr>
<tr>
<td>7</td>
<td>558</td>
<td>728</td>
<td>685</td>
</tr>
<tr>
<td>8</td>
<td>342</td>
<td>1103</td>
<td>3341</td>
</tr>
<tr>
<td>9</td>
<td>134</td>
<td>905</td>
<td>Not Solved</td>
</tr>
<tr>
<td>10</td>
<td>285</td>
<td>361</td>
<td>Not Solved</td>
</tr>
<tr>
<td>11</td>
<td>79</td>
<td>766</td>
<td>911</td>
</tr>
<tr>
<td>12</td>
<td>72</td>
<td>141</td>
<td>859</td>
</tr>
<tr>
<td>13</td>
<td>58</td>
<td>229</td>
<td>2611</td>
</tr>
<tr>
<td>14</td>
<td>245</td>
<td>2829</td>
<td>Not Solved</td>
</tr>
<tr>
<td>15</td>
<td>344</td>
<td>922</td>
<td>3640</td>
</tr>
<tr>
<td>16</td>
<td>356</td>
<td>280</td>
<td>Not Solved</td>
</tr>
<tr>
<td>17</td>
<td>140</td>
<td>425</td>
<td>2397</td>
</tr>
<tr>
<td>18</td>
<td>447</td>
<td>696</td>
<td>1044</td>
</tr>
<tr>
<td>19</td>
<td>513</td>
<td>489</td>
<td>775</td>
</tr>
<tr>
<td>20</td>
<td>242</td>
<td>279</td>
<td>392</td>
</tr>
<tr>
<td>21</td>
<td>638</td>
<td>634</td>
<td>3630</td>
</tr>
<tr>
<td>22</td>
<td>255</td>
<td>435</td>
<td>1066</td>
</tr>
<tr>
<td>23</td>
<td>314</td>
<td>125</td>
<td>826</td>
</tr>
<tr>
<td>24</td>
<td>269</td>
<td>326</td>
<td>1480</td>
</tr>
<tr>
<td>25</td>
<td>295</td>
<td>897</td>
<td>4555</td>
</tr>
<tr>
<td>26</td>
<td>107</td>
<td>694</td>
<td>669</td>
</tr>
<tr>
<td>27</td>
<td>57</td>
<td>802</td>
<td>2211</td>
</tr>
<tr>
<td>28</td>
<td>126</td>
<td>416</td>
<td>1025</td>
</tr>
<tr>
<td>29</td>
<td>579</td>
<td>349</td>
<td>872</td>
</tr>
<tr>
<td>30</td>
<td>141</td>
<td>855</td>
<td>Not Solved</td>
</tr>
<tr>
<td>31</td>
<td>132</td>
<td>218</td>
<td>3317</td>
</tr>
<tr>
<td>32</td>
<td>128</td>
<td>2918</td>
<td>2356</td>
</tr>
<tr>
<td>33</td>
<td>238</td>
<td>792</td>
<td>583</td>
</tr>
<tr>
<td>34</td>
<td>511</td>
<td>275</td>
<td>1147</td>
</tr>
</tbody>
</table>
Supersudoku
RFSS Solution
Hypersudoku

1. Only one numeral per cell
2. Exactly one of each numeral per row
3. Exactly one of each numeral per column
4. Exactly one of each numeral per region
5. Exactly one of each numeral per shaded region
Implications

• $M_{RFS}$ is $|C|$ by $|C|$ in size, while $M_{SM}$ is $|C|$ by $|X|$ (and $|X|$ can grow much exponentially in $|C|$, and vice versa)

• Are the linear approaches practical?

• Note that these NP-complete problems (e.g., X3C, MWPSLE, 0-1 Integer Programming) are solvable in polynomial time if the matrix ($M_{RFS}$ or $M_{SM}$) is non-singular. (Can solve some singular matrices.)

• Perhaps $M_{RFS}$ is non-singular when $M_{SM}$ is, and vice versa.

• $M_{RFS}$ works with arbitrary $|X|$ (XrC), even with continuous sets. Only requires set intersections.