Dipartimento di Matematica

R. Fontana, F. Rapallo, M. P. Rogantin

Markov bases for sudoku grids

Rapporto interno N. 4, marzo 2010

Politecnico di Torino

Corso Duca degli Abruzzi, 24-10129 Torino-Italia

Markov bases for sudoku grids

Roberto Fontana, Fabio Rapallo and Maria Piera Rogantin

1 Introduction and preliminary material

In recent years, sudoku has become a very popular game. In its most common form, the objective of the game is to complete a 9×9 grid with the digits from 1 to 9. Each digit must appear once and only once in each column, each row and each of the nine 3×3 boxes. It is known that the sudoku grids are special cases of Latin squares in the class of *gerechte designs*, see [2]. In [7] the connections between sudoku grids and experimental designs are extensively studied in the framework of Algebraic Statistics. In this paper we show how to represent sudoku games in terms of $0-1$ contingency tables. The connections between contingency tables, design of experiments, and the use of some techniques from Algebraic Statistics allows us to study and describe the set of all sudoku grids. Although the methodology is simple and can be easily stated for general $p^2 \times p^2$ ($p \geq 2$) sudoku grids, the computations are very intensive and then limited to the 4×4 case. However, we expect that our work could form a prototype to understand the connections between designed experiments and contingency tables for the general case.

A sudoku grid can be viewed as a particular subset of cardinality $p^2 \times p^2$ of the $p^2 \times p^2 \times p^2$ possible assignments of a digit between 1 and p^2 (or more generally p^2 symbols) to the cells of the grid. Under the point of view of design of experiments, a sudoku can be considered as a fraction of a full factorial design with four factors R, C, B, S, corresponding to rows, columns, boxes and symbols, with p^2 lev-

Roberto Fontana
DIMAT Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino Italy, e-mail: roberto.fontana@polito.it

Fabio Rapallo
DISTA University of Eastern Piedmont, Viale T. Michel 11, 15121 Alessandria Italy e-mail: fabio.rapallo@mfn.unipmn.it

Maria Piera Rogantin
DIMA Università di Genova, Via Dodecaneso 35, 16146 Genova Italy e-mail: rogantin@dima.unige.it

els each. The three position factors R, C and B are dependent; in fact a row and a column specify a box, but the polynomial relation between these three factors is fairly complicated. A simpler approach consists in splitting the row factor R into two pseudo-factors R_1 and R_2 with p levels each, and analogously the column factor C into C_1 and C_2. Then the box factor B corresponds to R_1 and C_1. The factors R_1 and C_1 are named "the band" and "the stack" respectively, and the factors R_2 and C_2 are named as "the row within a band" and "the column within a stack". Finally, given a digit k between 1 and p^2, the factors S_1 and S_2 provide the base-p representation of $k-1$. It should be noted that two factors for symbols are introduced only for symmetry of representation.

Hence, the full factorial has six factors R_1, R_2, C_1, C_2, S_1 and S_2, with p levels each. In this work we keep our exposition simpler using the integer coding for the levels $0, \ldots, p-1$.

As an example, in a 4×4 sudoku, if the symbol 3 (coded with 10, the binary representation of 2) is in the second row within the first band ($R_1 = 0, R_2 = 1$) and in the first column of the second stack ($C_1 = 1, C_2 = 0$), the corresponding point of the design is $(0,1,1,0,1,0)$.

As introduced in [7], a sudoku grid, as fraction of a full factorial design, is specified through its indicator polynomial function, and a move between two grids is also a polynomial. With such a methodology, three classes of moves are described: permutations of symbols, bands, rows within a band, stacks, columns within a stack, denoted by $\mathscr{M}_1$, transposition between rows and columns, denoted by $\mathscr{M}_2$, and moves acting on special parts of the sudoku grid, denoted by $\mathscr{M}_3$. While all the moves in $\mathscr{M}_1$ and $\mathscr{M}_2$ can be applied to all the sudoku, a move in $\mathscr{M}_3$ can be applied only to those sudoku having a special pattern. An example of a move in $\mathscr{M}_3$ is illustrated in Figure 1. This move acts on two parts of the grid defined by the intersection of a stack with two rows belonging to different boxes, and it exchanges the two symbols contained in it. But such a move is possible only when the symbols in each selected part are the same. Similar moves are defined for bands and columns, respectively. The relevance of the moves in $\mathscr{M}_3$ will be discussed in the next sections.

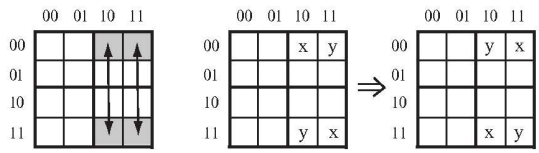

Fig. 1 A move in the class $\mathscr{M}_3$.

2 Moves and Markov bases for sudoku grids

The application of statistical techniques for contingency tables in the framework of the design of experiments, and in particular to sudoku grids, is not straightforward. In fact, a sudoku grid is not a contingency table as it contains labels instead of counts. To describe a sudoku grid as a contingency table, we need to consider a $p \times p \times p \times p \times p \times p$ table $\mathbf{n}$, with 6 indices r_1, r_2, c_1, c_2, s_1 and s_2, each ranging between 0 and $p-1$. The table $\mathbf{n}$ is a $0-1$ table with $n_{r_1 r_2 c_1 c_2 s_1 s_2} = 1$ if and only if the (r_1, r_2, c_1, c_2) cell of the grid contains the symbol (s_1, s_2) and is 0 otherwise. Notice that $n_{r_1 r_2 c_1 c_2 s_1 s_2}$ is the value of the indicator function of the fraction computed in the design point $(r_1, r_2, c_1, c_2, s_1, s_2)$. This approach has been already sketched in [7]. A similar approach is also described in [1] for different applications.

A $0-1$ contingency table must satisfy suitable constraints in order to represent a sudoku grid. The validity conditions are expressed as follows. Each sudoku grid must have one and only one symbol in each cell, in each row, in each column and in each box. The four constraints translate into the following linear conditions on $\mathbf{n}$:

$$\sum_{s_1,s_2=0}^{p-1} n_{r_1 r_2 c_1 c_2 s_1 s_2} = 1 \ \forall r_1, r_2, c_1, c_2 \qquad \sum_{c_1,c_2=0}^{p-1} n_{r_1 r_2 c_1 c_2 s_1 s_2} = 1 \ \forall \ r_1, r_2, s_1, s_2$$
$$\sum_{r_1,r_2=0}^{p-1} n_{r_1 r_2 c_1 c_2 s_1 s_2} = 1 \ \forall \ c_1, c_2, s_1, s_2 \qquad \sum_{r_2,c_2=0}^{p-1} n_{r_1 r_2 c_1 c_2 s_1 s_2} = 1 \ \forall \ r_1, c_1, s_1, s_2 \ .$$

Therefore we have a system with $4p^4$ linear conditions on $\mathbf{n}$ and the valid sudoku grids are just its integer non-negative solutions.

Given an integer linear system of equations, an integer (possibly negative) table $\mathbf{m}$ is a move if, for each non-negative solution $\mathbf{n}$ of the system, $\mathbf{n}+\mathbf{m}$ and $\mathbf{n}-\mathbf{m}$ are again solutions of the system, when non-negative.

As introduced in [5], a Markov basis is a finite set of moves $\mathscr{B} = \{\mathbf{m}_1, \ldots, \mathbf{m}_k\}$ which makes connected all the non-negative integer solutions of the above system of equations. Given any two sudoku $\mathbf{n}$ and $\mathbf{n}'$, there exists a sequence of moves $\mathbf{m}_{i_1}, \ldots, \mathbf{m}_{i_H}$ in $\mathscr{B}$ and a sequence of signs $\varepsilon_{i_1}, \ldots, \varepsilon_{i_H}$ ($\varepsilon_{i_j} = \pm 1$ for all j) such that

$$\mathbf{n}' = \mathbf{n} + \sum_{j=1}^{H} \varepsilon_{i_j} \mathbf{m}_{i_j}$$

and all the intermediate steps

$$\mathbf{n} + \sum_{j=1}^{h} \varepsilon_{i_j} \mathbf{m}_{i_j} \quad \text{for all} \ \ h = 1, \ldots, H$$

are again non-negative integer solutions of the linear system.

While all the linear constraints in our problem have constant term equal to 1, it is known that the computation of a Markov basis is independent on the constant term

of the linear system, see e.g. [5]. Therefore, we can select the subset $\mathscr{B}^f_+$ of $\mathscr{B}$ of the moves $\mathbf{m}$ of $\mathscr{B}$ that can be added at least to one sudoku grid, $\mathbf{n}$, in order to obtain again a valid sudoku grid, $\mathbf{n}+\mathbf{m}$. In the same way, we denote with $\mathscr{B}^f_-$ the subset of $\mathscr{B}$ formed by the elements that can be subtracted by at least one sudoku grid to obtain again a valid sudoku grid. It is immediate to check that $\mathscr{B}^f_+ = \mathscr{B}^f_-$. Thus, we denote this set simply with $\mathscr{B}^f$ and we refer to the moves in $\mathscr{B}^f$ as to the *feasible moves*.

The approach with Markov bases enables us to connect each pair of sudoku grids and to generate all the sudoku grids starting from a given one.

The actual computation of a Markov basis needs polynomial algorithms and symbolic computations. We refer to [6] for more details on Markov bases and how to compute them. Although in many problems involving contingency tables the Markov bases are small sets of moves, easy to compute and to handle, in the case of sudoku grids we have a large number of moves. Currently, the problem is computationally not feasible already in the case of classical 9×9 grids.

3 The 4×4 sudoku

In this section we consider the case of 4×4 sudoku, i. e., $p=2$. Using `4ti2` [11], we obtain the Markov basis $\mathscr{B}$. It contains $34,920$ elements while it is known that there are only 288 4×4 sudoku grids, listed in [7]. Using such list and some ad-hoc modules written in `SAS-IML` [10], we have explored this Markov basis finding some interesting facts.

- *Feasible moves.* Among the $34,920$ moves of the Markov basis $\mathscr{B}$ there are only $2,160$ feasible moves. To provide a term of comparison, we remind that the cardinality of the set of all the differences between two valid sudoku is $288\cdot 287/2 = 41,328$ and we have checked that $39,548$ of them are different.
- *Classification of moves.* If we classify each of the $2,160$ moves generated by $\mathscr{B}^f$ according to both the number of sudoku that can use it and the number of points of the grids that are changed by the move itself, we obtain the following table.

# of Sudoku that	# of Points moved				Total
can use the move	4	8	10	12	
1	0	0	1,536	192	1,728
2	0	336	0	0	336
8	96	0	0	0	96
Total	96	336	1,536	192	2,160

  – The 96 moves that change 4 points and can be used 8 times are all of type $\mathscr{M}_3$, like the one reported in Figure 1. We have also verified that these moves are enough to connect all the sudoku grids.

– The 336 moves that changes 8 points and can be used by 2 tables correspond to an exchange of two symbols, rows and columns. For example, two of them are:

	4	3	
	3		4
4			3
3		4	

$\Rightarrow$

	3	4	
	4		3
3			4
4		3	

4	2	3	1
3	1	2	4

$\Rightarrow$

3	1	2	4
4	2	3	1

– The remaining $1,728$ moves are suitable compositions of the previous ones. For instance, the move

3		4	1
	1	3	2
2	3		4
1	4	2	

$\Rightarrow$

1		3	4
	3	2	1
3	4		2
2	1	4	

is the composition of a permutation of two symbols, a permutation of two rows and two moves of type $\mathscr{M}_3$.

4 Partially filled 4×4 grids

Usually, a classical sudoku game is given as a partially filled grid that must be completed placing, in the empty cells, the right symbols. In framework of the design of experiments this procedure corresponds to augment an existing design under suitable properties, in this case those of the *gerechte designs*. For a people who prepares the sudoku grids, it is important to know where to place the givens and which symbols to put in them in order to obtain a grid with an unique completion. The non-empty cells in a sudoku game are shortly referred as *givens*. In this section, we use the Markov bases to study all possible givens for the 4×4 grids.

Notice that the definitions of Markov bases and feasible moves lead us to the following immediate result. When the Markov basis corresponding to a configuration of givens has no moves, then the partially filled grid can be completed in a unique way.

When some cells of the grids contain givens, we have to determine a Markov basis which does not act on such cells. This problem is then reduced to the computation of Markov bases for tables with structural zeros, as each given fixes 4 cells of the table $\mathbf{n}$. Theoretically, the computation of Markov bases for tables with structural zeros is a known problem, see e.g. [8]. A way to easily obtain all Markov bases for the partially filled sudoku should consist in the computation of a special Markov basis for the complete problem, known as *Universal Markov basis*, see [9]. An Universal Markov basis is a special Markov basis consistent with all configurations of structural zeros. More precisely, a Markov basis for a configuration with

structural zeros is formed by the moves of the Universal Markov basis for the complete problem by removing the moves which involve the fixed cells. In our problem, this means that the Universal Markov basis for the complete grid would be sufficient also for all partially filled grids. Notice that the Markov basis computed with `4ti2` and presented above is not Universa. Unfortunately, the dimensions of the problem make the computation of the Universal Markov bases currently unfeasible.

A different approach is to compute and analyze the Markov basis for all the possible choices C of the cells of the grid that should not be modified. For 4×4 sudoku it means that we should run `4ti2` over 2^{16} configurations corresponding to all the possible subsets of the cells of the grid. To reduce the computational effort we have exploited some symmetries of the sudoku grids, creating a partition of all the possible C and computing the Markov basis $\mathscr{B}_C$ only for one representative of each class. An approach to sudoku based on symmetries is also presented in [4].

The considered symmetries correspond to moves in $\mathscr{M}_1$ (permutations of bands, of rows within a band, of stacks and of columns within a stack) and in $\mathscr{M}_2$ (transposition) described in Section 1, moves that can be applied to all the sudoku. We describe now the construction of the classes of equivalence.

Let π_{e_1,e_2,e_3,e_4} be the transformation acting on the position of a cell:

$$\pi_{e_1,e_2,e_3,e_4}(r_1,r_2,c_1,c_2) = (r_1,r_2,c_1,c_2) + (e_1,e_2,e_3,e_4) \bmod 2 \quad \text{with } e_i \in \{0,1\} \ .$$

The permutation of a band corresponds to $(e_1,e_2,e_3,e_4) = (1,0,0,0)$, the permutation of the rows within both the bands corresponds to $(e_1,e_2,e_3,e_4) = (0,1,0,0)$, and the composition of both the permutations corresponds to $(e_1,e_2,e_3,e_4) = (1,1,0,0)$. Analogously for stacks and columns.

Let γ_e be the transposition of the position of a cell:

$$\gamma_e(r_1,r_2,c_1,c_2) = (c_1,c_2,r_1,r_2)(e=1) + (r_1,r_2,c_1,c_2)(e=0) \quad \text{with } e \in \{0,1\}$$

where (A) is 1 if the expression A is true and 0 otherwise.

Given $\mathbf{e} = (e_0,e_1,e_2,e_3,e_4,e_5)$, let $\tau_{\mathbf{e}}$ be the composition:

$$\tau_{\mathbf{e}} = \tau_{e_0,e_1,e_2,e_3,e_4,e_5} = \gamma_{e_0} \circ \pi_{e_1,e_2,e_3,e_4} \circ \gamma_{e_5} \quad \text{with } e_i \in \{0,1\} \ .$$

We notice that the transformation $\tau_{0,0,0,0,0,0}$ is the identity and that the transposition γ_e is considered both as the first and the last term in $\tau_{\mathbf{e}}$ because, in general, it is not commutative with π_{e_1,e_2,e_3,e_4}. We also point out that the 64 transformations $\tau_{\mathbf{e}}$ do not necessarily cover all the possible ones but they lead to significant reduction of the problem.

Given a subset of cells C, we denote by $\tau_{\mathbf{e}}(C)$ the transformation of all the cells of C. We say that two choices C_k and D_k, with k fixed cells, are equivalent if there exists a vector $\mathbf{e}$ such that:

$$D_k = \tau_{\mathbf{e}}(C_k)$$

and we write $C_k \sim D_k$. In Figure 2 a class of equivalence of grids is shown.

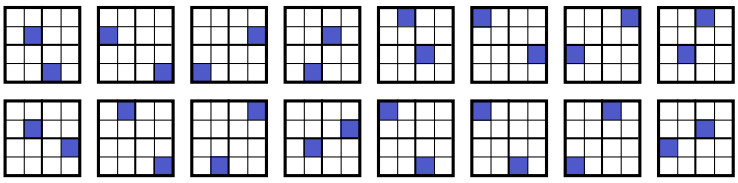

Fig. 2 The 16 grids of an equivalence class with 2 fixed cells, containing the givens.

Now we show that it is enough to compute the Markov basis only for one representative of each class.

Given a sudoku table $\mathbf{n}$ we denote by $\tilde{\tau}_{\mathbf{e}}(\mathbf{n})$ the transformation $\tau_{\mathbf{e}}$ applied to all the cells of the sudoku:

$$\tilde{\tau}_{\mathbf{e}}(n_{r_1,r_2,c_1,c_2,s_1,s_2}) = n_{\tau_{\mathbf{e}}(r_1,r_2,c_1,c_2),s_1,s_2} \forall\, r_1,r_2,c_1,c_2 \ .$$

In the same way $\tilde{\tau}_{\mathbf{e}}$ can be applied to a move $\mathbf{m}$.

Let C and D be in the same class of equivalence, $D = \tau_{\mathbf{e}}(C)$, and $\mathscr{B}_C^f$ and $\mathscr{B}_D^f$ be the corresponding sets of feasible moves in the Markov bases obtained from C and D. Then:

$$\mathscr{B}_D^f = \tilde{\tau}_{\mathbf{e}}\left(\mathscr{B}_C^f\right) \quad \text{with } \tilde{\tau}_{\mathbf{e}}\left(\mathscr{B}_C^f\right) = \left\{\tilde{\tau}_{\mathbf{e}}(\mathbf{m}),\ \mathbf{m} \in \mathscr{B}_C^f\right\} \ .$$

In fact, given $\mathbf{m} \in \mathscr{B}_C^f$, it follows that there exist a sudoku $\mathbf{n}$ and a sign ε such that $\mathbf{n} + \varepsilon\ \mathbf{m}$ is still a sudoku and:

$$\begin{aligned}
\tilde{\tau}_{\mathbf{e}}(\mathbf{n} + \varepsilon\ \mathbf{m}) &= \tilde{\tau}_{\mathbf{e}}\left((n + \varepsilon\ m)_{r_1,r_2,c_1,c_2,s_1,s_2}\right) \\
&= (n + \varepsilon\ m)_{\tau_{\mathbf{e}}(r_1,r_2,c_1,c_2),s_1,s_2} \\
&= n_{\tau_{\mathbf{e}}(r_1,r_2,c_1,c_2),s_1,s_2} + \varepsilon\ m_{\tau_{\mathbf{e}}(r_1,r_2,c_1,c_2),s_1,s_2} \\
&= \tilde{\tau}_{\mathbf{e}}(\mathbf{n}) + \varepsilon\ \tilde{\tau}_{\mathbf{e}}(\mathbf{m}) \ .
\end{aligned}$$

Therefore $\tilde{\tau}_{\mathbf{e}}(\mathbf{m})$ is a feasible move for the sudoku $\tilde{\tau}_{\mathbf{e}}(\mathbf{n})$. Moreover as $\mathbf{m}$ does not act on the cells C, $\tilde{\tau}_{\mathbf{e}}(\mathbf{m})$ does not act on the cells $\tau_{\mathbf{e}}(C)$. It follows that $\tilde{\tau}_{\mathbf{e}}(\mathbf{m})$ is in $\mathscr{B}^f_{\tau_{\mathbf{e}}(C)}$.

This methodology allows us to significantly reduce the computation, approximately of 96%, as summarized in the following table, where k is the number of fixed cells and $n_{\text{eq.cl.}}$ is the number of equivalence classes.

k	$1-15$	$2-14$	$3-13$	$4-12$	$5-11$	$6-10$	$7-9$	8	Total
$\binom{16}{k}$	16	120	560	1,820	4,368	8,008	11,440	12,870	65,534
$n_{\text{eq.cl.}}$	1	9	21	78	147	291	375	456	2,300

In view of this reduction, we have first computed a Markov basis $\mathscr{B}_C$ for one representative C for each of the 2,300 equivalence classes of the subsets of cells,

using `4ti2`. Then, using some ad hoc modules in SAS-IML, we have determined the feasible moves, $\mathscr{B}_C^f$ and the number of sudoku $\#\mathscr{S}_C$ that can use at least one move. The following table displays the results of all the $2,300$ runs, classified by the number of fixed cells k and the cardinality of $\mathscr{S}_C$.

$\#\mathscr{S}_C \setminus k$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	2	21	92	221	271	250	141	76	21	9	1
24	0	0	0	0	0	8	7	5	0	0	0	0	0	0	0
72	0	0	0	0	3	3	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	50	158	186	96	40	6	2	0	0	0
120	0	0	0	0	4	22	4	0	0	0	0	0	0	0	0
168	0	0	0	7	29	86	56	18	3	0	0	0	0	0	0
192	0	0	0	2	18	61	50	24	5	1	0	0	0	0	0
216	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0
240	0	0	0	4	43	16	2	0	0	0	0	0	0	0	0
264	0	0	0	16	8	2	0	0	0	0	0	0	0	0	0
288	1	9	21	49	33	22	6	2	0	0	0	0	0	0	0
Total	1	9	21	78	147	291	375	456	375	291	147	78	21	9	1

We can highlight some interesting facts.

- With $1,2$ or 3 fixed cells all the 288 sudoku can use at least one move, and therefore no choice of givens determines the completion of the grid univocally.
- With 4 fixed cells there are 49 patterns (or more precisely equivalent classes of patterns) that do not determine the completion of the grid univocally. Moreover for 7 patterns there are $288 - 168 = 120$ givens that determine the completion of the grid univocally; similarly for 2 patterns there are $288 - 192 = 96$ givens that determine the completion of the grid univocally, and so on. Here we have the verification that the minimum number of givens for the uniqueness of the completion is 4.
- With 5 fixed cells there are 2 patterns for which any choice of givens determines the completion of the grid univocally.
- With 8 fixed cells there are 2 patterns for which any choice of givens do not determine the completion of the grid univocally. Nevertheless for each pattern with 9 fixed cells there is a choice of givens which makes unique the completion of the grids. Then, the maximum number of fixed cell for which any choice of givens do not determine the completion of the grid univocally is 8.
- With 12 fixed cells there are 2 patterns for which 96 choices of givens do not determine the completion of the grid univocally.
- With $13,14$ and 15 fixed cells any choice of givens determines the completion of the grid univocally.

Figure 3.a shows that the same pattern of 4 cells (the shades ones) leads to a unique solution, if the givens are chosen like in the left part of the figure, or to more

than one solution, if the givens are chosen like in the right part of the figure. Figure 3.b is analogous to Figure 3.a but considers a pattern of 12 cells.

a)

1	2	4	3
3	4	2	1
4	1	3	2
2	3	1	4

2	4	1	3
1	3	4	2
4	2	3	1
3	1	2	4

1	2	3	4
4	3	1	2
2	1	4	3
3	4	2	1

b)

2	4	1	3
1	3	4	2
4	2	3	1
3	1	2	4

3	1	4	2
2	4	3	1
4	2	1	3
1	3	2	4

1	3	4	2
2	4	3	1
4	2	1	3
3	1	2	4

Fig. 3 Fixed a pattern, different choices of givens produce or not the uniqueness. Patterns with 4 and 12 fixed cells.

Figure 4 shows a pattern of 5 cells for which any choice of the givens corresponds to a unique solution.

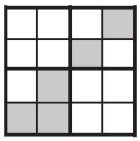

Fig. 4 A pattern of 5 cells for which any choice of the givens produces a unique solution.

5 Further developments

The use of Markov bases has allowed to study the moves between 4×4 sudoku grids, with nice properties for the study of partially filled grids. However, our theory has computational limitations when applied to standard 9×9 grids. Therefore, further work is needed to make our methodology and algorithms actually feasible for 9×9 grids. In particular, we will investigate the following points.

(a) To simplify the computation of Markov bases, using the special properties of the sudoku grids. For instance, some results in this direction is already known for design matrices with symmetries, see [1], and for contingency tables with strictly positive margins, see [3].

(b) To characterize the feasible moves theoretically. In fact, in our computations the selection of the feasible moves and the results in Section 4 are based on the

knowledge of the complete list of sudoku grids. This approach is then unfeasible in the 9×9 case.

(c) To make easy the computation of the Universal Markov basis for our problem, in order to avoid explicit computations for the study of the sudoku grids with givens.

References

1. Aoki, S., Takemura, A.: The largest group of invariance for Markov bases and toric ideals. J. Symb. Comput. **43**(5), 342–358 (2008)
2. Bailey, R.A., Cameron, P.J., Connelly, R.: Sudoku, Gerechte Designs, Resolutions, Affine Space, Spreads, Reguli, and Hamming Codes. Amer. Math. Monthly **115**(5), 383–404 (2008)
3. Chen, Y., Dinwoodie, I.H., Yoshida, R.: Markov chains, quotient ideals, and connectivity with positive margins. In: P. Gibilisco, E. Riccomagno, M.P. Rogantin, H.P. Wynn (eds.) Algebraic and Geometric Methods in Statistics, pp. 99–110. Cambridge University Press (2010)
4. Dahl, G.: Permutation matrices related to Sudoku. Linear Algebra Appl. **430**(8-9), 2457–2463 (2009)
5. Diaconis, P., Sturmfels, B.: Algebraic algorithms for sampling from conditional distributions. Ann. Statist. **26**(1), 363–397 (1998)
6. Drton, M., Sturmfels, B., Sullivant, S.: Lectures on Algebraic Statistics. Birkhauser, Basel (2009)
7. Fontana, R., Rogantin, M.P.: Indicator function and sudoku designs. In: P. Gibilisco, E. Riccomagno, M.P. Rogantin, H.P. Wynn (eds.) Algebraic and Geometric Methods in Statistics, pp. 203–224. Cambridge University Press (2010)
8. Rapallo, F.: Markov bases and structural zeros. J. Symb. Comput. **41**(2), 164–172 (2006)
9. Rapallo, F., Rogantin, M.P.: Markov chains on the reference set of contingency tables with upper bounds. Metron **65**(1), 35–51 (2007)
10. SAS Institute Inc.: SAS/IML®9.1 User's Guide. Cary, NC: SAS Institute Inc. (2004)
11. 4ti2 team: 4ti2—a software package for algebraic, geometric and combinatorial problems on linear spaces. Available at www.4ti2.de (2007)