The Journal of Logic and Algebraic Programming 79 (2010) 722–739

Contents lists available at ScienceDirect

The Journal of Logic and Algebraic Programming

journal homepage: www.elsevier.com/locate/jlap

Relation-algebraic specification and solution of special university timetabling problems

Rudolf Berghammer *, Britta Kehden

Institut für Informatik, Universität Kiel, Olshausenstraße 40, 24098 Kiel, Germany

ARTICLE INFO

Article history:
Available online 17 July 2010

Keywords:
Relation algebra
Timetabling
Formal program development
Tool support
RelView

ABSTRACT

In this paper, we are concerned with a special timetabling problem. It was posed to us by the administration of our university and stems from the adoption of the British-American system of university education in Germany. This change led to the concrete task of constructing a timetable that enables the undergraduate education of secondary school teachers within three years in the "normal case" and within four years in the case of exceptional combinations of subjects. We develop two relation-algebraic models of the timetabling problem and in each case algorithms for computing solutions. The latter easily can be implemented in the Kiel RelView tool showing that RelView can be used for timetabling.

1. Introduction

The study of relations has its roots in the second half of the 19th century with the pioneering works of Boole and de Morgan. Later on, Peirce and Schröder investigated the algebra of relations. The modern axiomatic development of relation algebra starts with the fundamental work of Tarski (cf. [16]) and his co-workers. In the last two decades this formalization has widely been used by many mathematicians and computer scientists as a very convenient base for formally dealing with fundamental concepts like graphs, orders, lattices, games and combinatorics in mathematics and data bases, Petri nets, preference and scaling, algorithmics, data types and semantics of programming languages in computer science. A lot of examples and references to relevant literature can be found in [5–8,15] and the proceedings of the international conferences "Relational Methods in Computer Science".

The construction of timetables for educational institutions and other purposes (see, e.g., [12] or [9], Section 5.6, for an overview or the proceedings of the international conferences "Practice and Theory of Automated Timetables" for many details) is also an area where relation algebra successfully has been applied. In [13,14] a relation-algebraic specification of an abstract timetabling problem is presented that covers a lot of concrete cases. It uses two input relations, viz. A that specifies whether a meeting can take place in a time slot and P that specifies whether a participant takes part in a meeting. Then a solution of the timetabling problem is a relation S between the meetings and the time slots that is univalent and total (i.e., a mapping in the relation-algebraic sense of [15], Section 4.2) and fulfils $S \subseteq A$ and $(PP^{\mathsf{T}} \cap \overline{\mathrm{I}})S \subseteq \overline{S}$. The first inclusion says that if S assigns a meeting m to time slot t, then m can take place in t, and the second inclusion ensures that if different meetings m and m' are in conflict, then m and m' are assigned to different time slots. In [10,11] this specification is reformulated in such a way that instead of A and S their corresponding vectors on the direct product of the meetings and the time slots are used. Interpreting a relation column-wisely as a list of vectors, this allows to combine relation algebra and randomized search heuristics and results in programs of the Kiel RelView tool (see [1,3]) for computing solutions.

In this paper, we combine again relation algebra and the computer system RelView to model another abstract timetabling problem and to compute solutions. The problem was posed to us by the administration of the University of Kiel and

* Corresponding author.
E-mail addresses: rub@informatik.uni-kiel.de (R. Berghammer), bk@informatik.uni-kiel.de (B. Kehden).

1567-8326/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jlap.2010.07.013

stems from Germany's agreement to the so-called Bologna accord. A consequence of this fact is the current change from the classical German university education system to the British-American undergraduate–graduate system with Bachelor and Master degrees. In particular with regard to the undergraduate education of secondary school teachers this change causes some difficulties. A very serious one is to enable a three years duration of study without to abolish Germany's tradition of (at least) two different subjects. Exactly this demand is the background of the administration's timetabling problem. Given its informal description, its input data and some additional desirable properties of possible solutions, we have been asked to construct a timetable that enables a three years duration of undergraduate-study in the case of the most selected combinations of subjects and a four years duration of study in the case of exceptional combinations of subjects.

The original approach of our university administration bases on the three categories "very frequently", "less common" and "hardly ever selected" of combinations of subjects and a rotation of time slots realized by a division of the subjects into groups and blocks. In Section 3, first, we will present the informal description we have obtained. Guided by [10,13,14], we then will show how it can be transformed into a formal relation-algebraic model of a new kind of an abstract timetabling problem. Using the latter as starting point, finally, we will formally develop an algorithm for testing solvability of the timetabling problem and for obtaining solutions if such exist. In essence, the algorithm is given by a relation-algebraic expression that immediately can be translated into the programming language of the RelView tool. So, RelView can be applied to timetabling.

With regard to its mathematical substance, the relation-algebraic model resulting from the administration's original approach turned out to be very attractive. Especially it allows – as we will demonstrate in Section 4 – to define the notion of isomorphic solutions and to compute, besides all solutions as done by the algorithm of Section 3, all solutions up to isomorphism only. This is very advantageous when solutions have to be evaluated and compared. However, in concrete applications the model proved to be cumbersome. Furthermore, our RelView experiments showed that a trisection of the combinations is unnecessary in practice since, to obtain at least one solution of the timetabling problem, in all realistic cases the categories had to be modified in such a way that "less common" became almost empty. As a consequence, we developed a more simple alternative to the administration's model that works with two categories only and, guided by the original approach, an algorithm for computing solutions in case of the new model, too. All this is presented in Sections 5. In Section 6, we sketch a second method for obtaining solutions in case of the alternative model that bases on a concept of graph theory, viz. maximum stable sets.

A disadvantage of the new model and its algorithms is that for solvable timetabling problems the number of computed solutions may become very large, but in essence only a few of them are non-isomorphic, i.e., really of interest. This makes it difficult to compare solutions and to select a specific solution that fulfills additional properties. However, as we will show in Section 7, the great advantage of the new model is that it allows a considerable reduction of the problem size. This enables RelView to compute all solutions of the concrete timetabling problem posed to us by the administration of our university within a few seconds only.

2. Relation-algebraic preliminaries

In this section, we provide the relation-algebraic material used in the remainder of the paper. For more details concerning relation algebra, see [6,15] for example.

We denote the set (or type) of all relations with domain X and range Y by $[X \leftrightarrow Y]$ instead of $2^{X \times Y}$ and write $R : X \leftrightarrow Y$ instead of $R \in [X \leftrightarrow Y]$. If the sets X and Y are finite, we may consider R as a Boolean matrix. This specific interpretation is well suited for many purposes and also one of the possibilities to depict relations in RelView; cf. [1,3]. Therefore, we use in this paper often matrix notation and terminology. Especially, we speak about rows, columns and entries of relations, and write $R_{x,y}$ instead of $\langle x, y\rangle \in R$ or $x\,R\,y$.

We assume the reader to be familiar with the basic operations on relations, viz. R^{T} (transposition, conversion), $\overline{R}$ (complement), $R \cup S$ (join), $R \cap S$ (meet), RS (composition, multiplication), $R \subseteq S$ (inclusion), and the special relations O (empty relation), L (universal relation) and I (identity relation). Each type $[X \leftrightarrow Y]$ forms with the operations $\overline{}$, $\cup$, $\cap$, the ordering $\subseteq$ and the constants O and L a complete Boolean lattice. Further well-known rules are, for instance, $R^{\mathsf{T}^{\mathsf{T}}} = R$, $\overline{R^{\mathsf{T}}} = \overline{R}^{\mathsf{T}}$ and that $R \subseteq S$ implies $R^{\mathsf{T}} \subseteq S^{\mathsf{T}}$. The theoretical framework for these rules and many others to hold is that of an (axiomatic, typed) relation algebra. For each type respectively pair / triple of types we have those of the set-theoretic relations as constants and operations of this algebraic structure. The axioms of a relation algebra are the axioms of a complete Boolean lattice for complement, meet, join, ordering, empty relation and universal relation, the associativity and neutrality of identity relations for composition, the equivalence of $QR \subseteq S$, $Q^{\mathsf{T}}\,\overline{S} \subseteq \overline{R}$ and $\overline{S}\,R^{\mathsf{T}} \subseteq \overline{Q}$ (*Schröder rule*) and that $R \neq \mathsf{O}$ implies $\mathsf{L}R\mathsf{L} = \mathsf{L}$ (*Tarski rule*). From the latter axiom we obtain that either $\mathsf{L}R\mathsf{L} = \mathsf{L}$ or $\mathsf{L}R\mathsf{L} = \mathsf{O}$ and that relational inclusion can be described via

$$R \subseteq S \iff \overline{\mathsf{L}(R \cap \overline{S})\mathsf{L}} = \mathsf{L}. \tag{1}$$

Typing the universal relations of the left-hand side of $\overline{\mathsf{L}(R \cap \overline{S})\mathsf{L}} = \mathsf{L}$ in (1) in such a way that the universal relation of the equation's right-hand side has a singleton set **1** as domain and range and using the only two relations of $[\mathbf{1} \leftrightarrow \mathbf{1}]$ as model for the Booleans, it is possible to translate every Boolean combination φ of relation-algebraic inclusions into a relation-algebraic expression e such that φ holds if and only if $e = \mathsf{L}$. This follows from the fact that on $[\mathbf{1} \leftrightarrow \mathbf{1}]$ the relational operations $\overline{}$, $\cup$ and $\cap$ directly correspond to the logical connectives $\neg$, $\vee$ and $\wedge$.

There are some relation-algebraic possibilities to model sets. Our first modeling uses *(column) vectors*, which are relations v with $v = v\mathsf{L}$. Since for a vector the range is irrelevant, we consider mostly vectors $v : X \leftrightarrow \mathbf{1}$ with the singleton set $\mathbf{1} = \{\perp\}$ as range and omit in such cases the subscript $\perp$, i.e., write v_x instead of $v_{x,\perp}$ and say then that the x-entry of v is 1. Such a vector can be considered as a Boolean matrix with exactly one column, i.e., as a Boolean column vector, and *represents* the subset $\{x \in X \mid v_x\}$ of X. Sets of vectors are closed under forming complements, joins, meets and left-compositions Rv. As a consequence, for vectors property (1) simplifies to

$$v \subseteq w \iff \overline{\mathsf{L}(v \cap \overline{w})} = \mathsf{L}. \tag{2}$$

A non-empty vector v is a *point* if $vv^{\mathsf{T}} \subseteq \mathsf{I}$, i.e., it is *injective*. This means that it represents a singleton subset of its domain or an element from it if we identify a singleton set $\{x\}$ with the element x. In the matrix model, hence, a point $v : X \leftrightarrow \mathbf{1}$ is a Boolean column vector in which exactly one entry is 1.

Given $y \in Y$, with $R^{(y)}$ we denote the y-column of the relation $R : X \leftrightarrow Y$. That is, $R^{(y)}$ has type $[X \leftrightarrow \mathbf{1}]$ and for all $x \in X$ are $R^{(y)}_x$ and $R_{x,y}$ equivalent. To compare the columns of two relations R and S with the same domain X and possible different ranges Y and Y', we use the symmetric quotient

$$syq(R, S) = \overline{R^{\mathsf{T}}\,\overline{S}} \;\cap\; \overline{\overline{R}^{\mathsf{T}} S} \tag{3}$$

of them. The type of $syq(R, S)$ is $[Y \leftrightarrow Y']$, and transforming (3) into a component-wise notation we have for all $y \in Y$ and $y' \in Y'$ that $syq(R, S)_{y,y'}$ if and only if $R^{(y)} = S^{(y')}$.

As a second way to deal with sets, we will apply the relation-level equivalents of the set-theoretic symbol $\in$, that is, *membership-relations* $\mathsf{M} : X \leftrightarrow 2^X$. These specific relations are defined by demanding for all elements $x \in X$ and sets $Y \in 2^X$ that $\mathsf{M}_{x,Y}$ if and only if $x \in Y$. A simple Boolean matrix implementation of membership-relations requires an exponential number of bits. However, in [2,3] an implementation of $\mathsf{M} : X \leftrightarrow 2^X$ using binary decision diagrams (BDDs) is presented, where the number of BDD-vertices is linear in the size of the base set X. This implementation is part of RelView.

Finally, we will use injective mappings for modeling sets. Given an injective function $\iota : Y \to X$ in the usual mathematical sense, we may consider Y as a subset of X by identifying it with its image under ι. If Y is actually a subset of X and ι is given as a relation of type $[Y \leftrightarrow X]$ such that $\iota_{y,x}$ if and only if $y = x$ for all $y \in Y$ and $x \in X$, then the vector $\iota^{\mathsf{T}}\mathsf{L} : X \leftrightarrow \mathbf{1}$ represents Y as a subset of X in the sense above. Clearly, the transition in the other direction is also possible, i.e., the generation of a relation $inj(v) : Y \leftrightarrow X$ from the vector representation $v : X \leftrightarrow \mathbf{1}$ of the subset Y of X such that for all $y \in Y$ and $x \in X$ we have $inj(v)_{y,x}$ if and only if $y = x$. We obtain $inj(v)$ by removing from $\mathsf{I} : X \leftrightarrow X$ all rows which correspond to a 0-entry in v. The relation $inj(v)$ is an injective mapping in the relation-algebraic sense. A combination of such relations with membership-relations allows a *column-wise representation* of sets of subsets. More specifically, if the vector $v : 2^X \leftrightarrow \mathbf{1}$ represents a subset $\mathcal{S}$ of 2^X in the sense above, i.e., $\mathcal{S}$ equals the set $\{S \in 2^X \mid v_S\}$, then for all $x \in X$ and $Y \in \mathcal{S}$ we get the equivalence of $(\mathsf{M}\, inj(v)^{\mathsf{T}})_{x,Y}$ and $x \in Y$. This means, that the elements of $\mathcal{S}$ are represented precisely by the columns of the relation $\mathsf{M}\, inj(v)^{\mathsf{T}} : X \leftrightarrow \mathcal{S}$.

Given a direct product $X \times Y$ of sets X and Y, there are the natural projections which decompose a pair $u = \langle u_1, u_2\rangle$ into its first component u_1 and its second component u_2. (Throughout this paper, pairs u are assumed to be of the form $u = \langle u_1, u_2\rangle$, i.e., the first component of u is denoted by u_1 and the second component by u_2.) For a relation-algebraic approach, it is very useful to consider instead of these two functions the two corresponding *projection relations* $\pi : X\times Y \leftrightarrow X$ and $\rho : X\times Y \leftrightarrow Y$ such that, given any $u \in X\times Y$, it holds $\pi_{u,x}$ if and only if $u_1 = x$ and $\rho_{u,y}$ if and only if $u_2 = y$. Projection relations algebraically allow to specify the *parallel composition* $R \parallel S : X\times X' \leftrightarrow Y\times Y'$ of relations $R : X \leftrightarrow Y$ and $S : X' \leftrightarrow Y'$ in such a way that $(R \parallel S)_{u,v}$ is equivalent to R_{u_1,v_1} and S_{u_2,v_2} for all $u \in X\times X'$ and $v \in Y\times Y'$. We get this property via the definition

$$R \parallel S = \pi R\sigma^{\mathsf{T}} \cap \rho S\tau^{\mathsf{T}}, \tag{4}$$

where $\pi : X\times X' \leftrightarrow X$ and $\rho : X\times X' \leftrightarrow X'$ are the two projection relations of $X \times X'$ and $\sigma : Y\times Y' \leftrightarrow Y$ and $\tau : Y\times Y' \leftrightarrow Y'$ are those of $Y \times Y'$.

We end this section with two functions (in the usual mathematical sense) which establish a Boolean lattice isomorphism between the two Boolean lattices $[X \leftrightarrow Y]$ and $[X\times Y \leftrightarrow \mathbf{1}]$. The direction from $[X \leftrightarrow Y]$ to $[X\times Y \leftrightarrow \mathbf{1}]$ is given by the isomorphism *vec*, where

$$vec(R) = (\pi R \cap \rho)\mathsf{L} \tag{5}$$

defines the vector $vec(R)$ corresponding to the relation R, and that from $[X\times Y \leftrightarrow \mathbf{1}]$ to $[X \leftrightarrow Y]$ by the inverse isomorphism *rel*, where

$$rel(v) = \pi^{\mathsf{T}}(\rho \cap v\mathsf{L}^{\mathsf{T}}) \tag{6}$$

defines the relation $rel(v)$ corresponding to the vector v. In the two equations (5) and (6) $\pi : X\times Y \leftrightarrow X$ and $\rho : X\times Y \leftrightarrow Y$ are the projection relations of the underlying direct product and L is a universal vector of type $[Y \leftrightarrow \mathbf{1}]$. Using a component-wise notation, these definitions say that for all $x \in X$ and $y \in Y$ we have $R_{x,y}$ if and only if $vec(R)_{\langle x,y\rangle}$ and $v_{\langle x,y\rangle}$ if and only if $rel(v)_{x,y}$. Decisive for our latter applications is

$$vec(QSR) = (Q \parallel R^{\mathsf{T}})vec(S). \tag{7}$$

Property (7) is proved in [10,11] using the definition (4) and the relation-algebraic axiomatization of the projection relations of direct products given, for example, in [15]. Two immediate consequences of (7) are the special cases $vec(QS) = (Q \parallel \mathsf{I})vec(S)$ and $vec(SR) = (\mathsf{I} \parallel R^{\mathsf{T}})vec(S)$.

3. Timetabling using the administration's original approach

The background of the timetabling problem of this paper is as follows: presently at our university (the CAU Kiel) there exist 34 different subjects for the undergraduate education of secondary school teachers (and, to be correct, some others professions which corresponds to the former education in these subjects ending with a Magister degree). According to the examination regulations, each student has to select two subjects. Experience with the classical system has shown that all possible combinations roughly can be divided into three categories, viz. the very frequently ones, the less common ones and those which are hardly ever selected. The goal is to construct a timetable that enables a three years duration of study for combinations of the first category and a four years duration of study for combinations of the second category. Concretely this means that there are no conflicts between the courses of the two subjects if they belong to the first category during the entire duration of study and for the second category conflicts at the most appear in one of three years, which enforces a fourth year of study. As a further goal, the number of conflicts should be very small. To this purpose, the university administration divided the 34 subjects into 9 groups, denoted by $A, B, \ldots, H, I$, and then the groups in turn into three blocks 1, 2 and 3 as shown in the following three tables via the block- and the group-columns:

group	year 1	year 2	year 3
A	t_1	t_1	t_1
B	t_2	t_2	t_2
C	t_3	t_3	t_3

block 1

group	year 1	year 2	year 3
D	t_1	t_2	t_3
E	t_2	t_3	t_1
F	t_3	t_1	t_2

block 2

group	year 1	year 2	year 3
G	t_1	t_3	t_2
H	t_2	t_1	t_3
I	t_3	t_2	t_1

block 3

The meaning of the three year-columns of the tables is as follows. First, each week is divided into three disjoint time slots of equal size, denoted by t_1, t_2 and t_3, and this partitioning remains constant over a long period. For each academic year then each course of the undergraduate-education of secondary school teachers is assigned to a time slot in such a way that all courses of a subject take place in the same time slot. The table on the left indicates that for the first block this assignment remains constant over three academic years. For instance, every year all courses of a subject from group A take place in time slot t_1. For the other blocks, by contrast, the assignment of courses to time slots cyclically changes, as shown in the remaining two tables. To give also here an example, all courses of a subject from group D take place in time slot t_n in year n, $1 \leq n \leq 3$.

An immediate consequence of the administration's approach is that the duration of study is three years if and only if the two subjects of the combination belong to different groups of the same block. Four years suffice to take part in the combination's courses if the subjects belong to groups of different blocks. Now, from our administration we obtained the classification of the combinations and our task was to compute a function from the set of subjects to the set of groups with the following properties:

(a) If two subjects are mapped to the same group, then they form a combination of the third category.
(b) If two subjects form a combination of the first category, then their groups belong to the same block.

Both (a) and (b) namely imply that all combinations of the most important first category belong to different groups of the same block. In case that the desired function does not exist, we have been asked to compute at least a partial function for which (a) and (b) hold. Thus, the administration expected to obtain enough information that allows to experiment with the partitioning of the combinations such that, finally, one is found that allows a solution of the timetabling problem but still is reasonable with respect to the frequency of the combination's choices.

To formalize the just presented informal description to a general abstract timetabling problen, we assume $\mathfrak{S}$ to denote a finite set of subjects, $\mathfrak{G}$ to denote a finite set of groups and $\mathfrak{B}$ to denote a finite set of blocks. (In the above described concrete case we have $|\mathfrak{S}| = 34$, $|\mathfrak{G}| = 9$ and $|\mathfrak{B}| = 3$.) For modeling the partitioning of groups into blocks, we furthermore assume a relation $Q : \mathfrak{G} \leftrightarrow \mathfrak{B}$ such that for all $g \in \mathfrak{G}$ and $b \in \mathfrak{B}$ we have $Q_{g,b}$ if and only if group g belongs to block b. Then the reflexive and symmetric relation $B = QQ^{\mathsf{T}} : \mathfrak{G} \leftrightarrow \mathfrak{G}$ fulfils

$$B_{g,g'} \iff g \text{ and } g' \text{ belong to the same block} \tag{8}$$

for all groups $g, g' \in \mathfrak{G}$. And, finally, we assume a specification of the partition of the set of all possible combinations of subjects into the three categories "very frequently", "less common" and "hardly ever selected" by two relations, viz. $J : \mathfrak{S} \leftrightarrow \mathfrak{S}$ such that

$$J_{s,s'} \iff s \neq s' \text{ and } s, s' \text{ is a combination of the first category} \tag{9}$$

for all subjects $s, s' \in \mathfrak{S}$, and $N : \mathfrak{S} \leftrightarrow \mathfrak{S}$ such that

$$N_{s,s'} \iff s = s' \text{ or } s, s' \text{ is a combination of the third category} \tag{10}$$

for all subjects $s, s' \in \mathfrak{S}$. Then $\overline{J \cup N}$ relates two subjects if and only if they are different and form a combination of the second category. Note that also J and N are symmetric, J is irreflexive, and N is reflexive. The reflexivity of N is motivated by the informal requirement that the duration of study is three years if and only if the two subjects of the combination belong to different groups of the same block. The relations of (8) to (10) constitute the input of the university timetabling problem. Based on them, now we relation-algebraically specify its output.

Definition 3.1. Given the three input relations $B : \mathfrak{G} \leftrightarrow \mathfrak{G}$, $J : \mathfrak{S} \leftrightarrow \mathfrak{S}$ and $N : \mathfrak{S} \leftrightarrow \mathfrak{S}$, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{G}$ is a solution of the university timetabling problem if $\overline{N}S \subseteq \overline{S}$, $JS \subseteq \overline{S\overline{B}}$, $S^{\mathsf{T}}S \subseteq \mathsf{I}$ and $\mathsf{L} \subseteq S\mathsf{L}$.

The four inclusions of Definition 3.1 are a relation-algebraic formalization of the above informal requirements. In the case of $\overline{N}S \subseteq \overline{S}$ this is shown by the following calculation. It starts with the logical formalization of property (a) and transforms it step-by-step into the first inclusion of Definition 3.1, thereby replacing logical constructions by their relation-algebraic counterparts.

$$\begin{aligned}
\forall s, s', g : S_{s,g} \wedge S_{s',g} \rightarrow N_{s,s'} &\iff \neg\exists s, s', g : S_{s,g} \wedge S_{s',g} \wedge \overline{N}_{s,s'} \\
&\iff \neg\exists s, g : S_{s,g} \wedge (\overline{N}S)_{s,g} \\
&\iff \forall s, g : (\overline{N}S)_{s,g} \rightarrow \overline{S}_{s,g} \\
&\iff \overline{N}S \subseteq \overline{S}.
\end{aligned}$$

In the same way the second inclusion $JS \subseteq \overline{S\overline{B}}$ of Definition 3.1 is obtained from the logical formalization of property (b). The remaining two inclusions of Definition 3.1 relation-algebraically specify S to be a univalent (third inclusion) and total (fourth inclusion) relation, i.e., to be a mapping (in the relation-algebraic sense) from the set of subjects to the set of groups.

Based on an idea presented in [10], the above non-algorithmic relation-algebraic specification of a solution S of the university timetabling problem now will be reformulated in such a way that instead of S its corresponding vector $vec(S)$ is used. This change of representation, finally, will lead to an algorithmic specification. The following theorem is the key of the approach. In its inclusion $\mathsf{L} \subseteq \pi^{\mathsf{T}}v$, by $\pi : \mathfrak{S}\times\mathfrak{G} \leftrightarrow \mathfrak{S}$ we denote the first projection relation of $\mathfrak{S} \times \mathfrak{G}$.

Theorem 3.1. *Assume the three input relations $B : \mathfrak{G} \leftrightarrow \mathfrak{G}$, $J : \mathfrak{S} \leftrightarrow \mathfrak{S}$ and $N : \mathfrak{S} \leftrightarrow \mathfrak{S}$, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{G}$ and a vector $v : \mathfrak{S}\times\mathfrak{G} \leftrightarrow \mathbf{1}$ such that $v = vec(S)$. Then S is a solution of the university timetabling problem if and only if $(\overline{N} \parallel \mathsf{I})v \subseteq \overline{v}$, $(J \parallel \mathsf{I})v \subseteq \overline{(\mathsf{I} \parallel \overline{B})v}$, $(\mathsf{I} \parallel \overline{\mathsf{I}})v \subseteq \overline{v}$ and $\mathsf{L} \subseteq \pi^{\mathsf{T}}v$.*

Proof. We show that for all n, $1 \leq n \leq 4$, the nth inclusion of Definition 3.1 is equivalent to the nth inclusion of the theorem. The following calculation proves the equivalence for the case $n = 1$:

$$\begin{aligned}
\overline{N}S \subseteq \overline{S} &\iff vec(\overline{N}S) \subseteq vec(\overline{S}) && vec \text{ isomorphism} \\
&\iff (\overline{N} \parallel \mathsf{I})vec(S) \subseteq vec(\overline{S}) && \text{due to (7)} \\
&\iff (\overline{N} \parallel \mathsf{I})vec(S) \subseteq \overline{vec(S)} && vec \text{ isomorphism} \\
&\iff (\overline{N} \parallel \mathsf{I})v \subseteq \overline{v} && v = vec(S)
\end{aligned}$$

The equivalence of the second inclusions is shown as follows:

$$
\begin{array}{rcll}
JS \subseteq \overline{S\overline{B}} & \iff & vec(JS) \subseteq vec(\overline{S\overline{B}}) & vec \text{ isomorphism} \\
& \iff & vec(JS) \subseteq \overline{vec(S\overline{B})} & vec \text{ isomorphism} \\
& \iff & (J \parallel \mathsf{I})vec(S) \subseteq \overline{(\mathsf{I} \parallel \overline{B}^{\mathsf{T}})vec(S)} & \text{due to (7)} \\
& \iff & (J \parallel \mathsf{I})vec(S) \subseteq \overline{(\mathsf{I} \parallel \overline{B})vec(S)} & B \text{ is symmetric} \\
& \iff & (J \parallel \mathsf{I})v \subseteq \overline{(\mathsf{I} \parallel \overline{B})v} & v = vec(S)
\end{array}
$$

The subsequent calculation establishes the equivalence of the two inclusions concerning univalence of S:

$$
\begin{array}{rcll}
S^{\mathsf{T}}S \subseteq \mathsf{I} & \iff & S\bar{\mathsf{I}} \subseteq \bar{S} & [15, \text{Prop. 4.2.1}] \\
& \iff & vec(S\bar{\mathsf{I}}) \subseteq vec(\bar{S}) & vec \text{ isomorphism} \\
& \iff & (\mathsf{I} \parallel \bar{\mathsf{I}}^{\mathsf{T}})vec(S) \subseteq vec(\bar{S}) & \text{due to (7)} \\
& \iff & (\mathsf{I} \parallel \bar{\mathsf{I}}^{\mathsf{T}})vec(S) \subseteq \overline{vec(S)} & vec \text{ isomorphism} \\
& \iff & (\mathsf{I} \parallel \bar{\mathsf{I}})v \subseteq \bar{v} & \bar{\mathsf{I}} \text{ is symmetric, } v = vec(S)
\end{array}
$$

It remains to verify the last inclusions to be equivalent. Here we have:

$$
\begin{array}{rcll}
\mathsf{L} \subseteq S\mathsf{L} & \iff & vec(\mathsf{L}) \subseteq vec(S\mathsf{L}) & vec \text{ isomorphism} \\
& \iff & \mathsf{L} \subseteq (\mathsf{I} \parallel \mathsf{L}^{\mathsf{T}})vec(S) & vec \text{ isomorphism(7)} \\
& \iff & \mathsf{L} \subseteq (\pi\pi^{\mathsf{T}} \cap \rho\mathsf{L}^{\mathsf{T}}\rho^{\mathsf{T}})vec(S) & \text{due to (4)} \\
& \iff & \mathsf{L} \subseteq (\pi\pi^{\mathsf{T}} \cap \mathsf{L})vec(S) & \rho \text{ is total} \\
& \iff & \mathsf{L} \subseteq \pi\pi^{\mathsf{T}}v & v = vec(S) \\
& \iff & \mathsf{L} \subseteq \pi^{\mathsf{T}}v &
\end{array}
$$

The direction "$\Rightarrow$" of the last step follows from the surjectivity and univalence of π since this implies $\mathsf{L} = \pi^{\mathsf{T}}\mathsf{L} \subseteq \pi^{\mathsf{T}}\pi\pi^{\mathsf{T}}v \subseteq \mathsf{I}\pi^{\mathsf{T}}v = \pi^{\mathsf{T}}v$, and the direction "$\Leftarrow$" is a consequence of the totality of π, since $\mathsf{L} \subseteq \pi\mathsf{L} \subseteq \pi\pi^{\mathsf{T}}v$. $\Box$

Now, we are in a position to present a relation-algebraic expression that depends on a vector $v = vec(S)$ and evaluates to the universal relation of type $[\mathbf{1} \leftrightarrow \mathbf{1}]$ if and only if v corresponds to a solution S of the university timetabling problem. In the equation of the following theorem, this expression constitutes the left-hand side.

Theorem 3.2. *Assume again the relations B, J, N, S, v and π as in Theorem 3.1. Then S is a solution of the university timetabling problem if and only if*

$$\overline{\mathsf{L}(((\overline{N} \parallel \mathsf{I})v \cap v) \cup ((J \parallel \mathsf{I})v \cap (\mathsf{I} \parallel \overline{B})v) \cup ((\mathsf{I} \parallel \bar{\mathsf{I}})v \cap v) \cup \mathsf{L}\overline{\pi^{\mathsf{T}}v})} = \mathsf{L}.$$

Proof. Property (2) of Section 2 implies the following equivalences:

$$
\begin{array}{rcl}
(\overline{N} \parallel \mathsf{I})v \subseteq \bar{v} & \iff & \overline{\mathsf{L}((\overline{N} \parallel \mathsf{I})v \cap v)} = \mathsf{L}, \\
(J \parallel \mathsf{I})v \subseteq \overline{(\mathsf{I} \parallel \overline{B})v} & \iff & \overline{\mathsf{L}((J \parallel \mathsf{I})v \cap (\mathsf{I} \parallel \overline{B})v)} = \mathsf{L}, \\
(\mathsf{I} \parallel \bar{\mathsf{I}})v \subseteq \bar{v} & \iff & \overline{\mathsf{L}((\mathsf{I} \parallel \bar{\mathsf{I}})v \cap v)} = \mathsf{L}, \\
\mathsf{L} \subseteq \pi^{\mathsf{T}}v & \iff & \overline{\mathsf{L}\overline{\pi^{\mathsf{T}}v}} = \mathsf{L}.
\end{array}
$$

Combining this with Theorem 3.1, we get that S is a solution of the university timetabling problem if and only if

$$\overline{\mathsf{L}((\overline{N} \parallel \mathsf{I})v \cap v)} \cap \overline{\mathsf{L}((J \parallel \mathsf{I})v \cap (\mathsf{I} \parallel \overline{B})v)} \cap \overline{\mathsf{L}((\mathsf{I} \parallel \bar{\mathsf{I}})v \cap v)} \cap \overline{\mathsf{L}\overline{\pi^{\mathsf{T}}v}} = \mathsf{L}.$$

Next, we apply a de Morgan law and transform this equation into

$$\overline{\mathsf{L}((\overline{N} \parallel \mathsf{I})v \cap v) \cup \mathsf{L}((J \parallel \mathsf{I})v \cap (\mathsf{I} \parallel \overline{B})v) \cup \mathsf{L}((\mathsf{I} \parallel \bar{\mathsf{I}})v \cap v) \cup \mathsf{L}\overline{\pi^{\mathsf{T}}v}} = \mathsf{L}.$$

Finally, we replace the universal relation $\mathsf{L} : \mathbf{1} \leftrightarrow \mathfrak{G}$ of $\mathsf{L}\,\overline{\pi^{\mathsf{T}} v}$ by a composition $\mathsf{L}\mathsf{L}$, where the first L has type $[\mathbf{1} \leftrightarrow \mathfrak{S} \times \mathfrak{G}]$ and the second L has type $[\mathfrak{S} \times \mathfrak{G} \leftrightarrow \mathfrak{G}]$. This adaption of types allows to apply a distributivity law, which yields the desired result. □

Considering now v as a variable, the left-hand side of the equation of Theorem 3.2 leads to the following function Φ on relations, where the first universal relation L has type $[\mathbf{1} \leftrightarrow \mathfrak{S} \times \mathfrak{G}]$, the second L has type $[\mathfrak{S} \times \mathfrak{G} \leftrightarrow \mathfrak{G}]$ and X is the name of the variable.

$$\Phi(X) = \overline{\mathsf{L}(((\overline{N} \parallel \mathsf{I})X \cap X) \cup ((J \parallel \mathsf{I})X \cap (\mathsf{I} \parallel \overline{B})X) \cup ((\mathsf{I} \parallel \bar{\mathsf{I}})X \cap X) \cup \mathsf{L}\,\overline{\pi^{\mathsf{T}} X})}.$$

When applied to a vector $v : \mathfrak{S} \times \mathfrak{G} \leftrightarrow \mathbf{1}$, this function returns $\mathsf{L} : \mathbf{1} \leftrightarrow \mathbf{1}$ if and only if v corresponds to a solution of the university timetabling problem, and $\mathsf{O} : \mathbf{1} \leftrightarrow \mathbf{1}$ otherwise. A specific feature of Φ is that it is defined using the variable X, constant relations, complements, joins, meets and left-compositions only. Hence, it is a vector predicate in the sense of [10,11]. With the aid of the membership-relation $\mathsf{M} : \mathfrak{S} \times \mathfrak{G} \leftrightarrow [\mathfrak{S} \leftrightarrow \mathfrak{G}]$ we, therefore, obtain a vector

$$t = \Phi(\mathsf{M})^{\mathsf{T}} \tag{11}$$

of type $[[\mathfrak{S} \leftrightarrow \mathfrak{G}] \leftrightarrow \mathbf{1}]$ such that for all relations $X : \mathfrak{S} \leftrightarrow \mathfrak{G}$ we have t_X if and only if the X-column of M, considered as a vector $\mathsf{M}^{(X)} : \mathfrak{S} \times \mathfrak{G} \leftrightarrow \mathbf{1}$, corresponds to a solution of the university timetabling problem. From (11) a column-wise representation of all vectors which yield by their corresponding relations all solutions of our university timetabling problem may be obtained using the technique described in Section 2. But the vector t also allows to compute one (or even all) single solution(s) in the sense of Definition 3.1. The procedure is rather simple: first, a point $p \subseteq t$ is selected (for instance, in RelView via the pre-defined operation `point`). Because of the above property, then the vector $\mathsf{M}p : \mathfrak{S} \times \mathfrak{G} \leftrightarrow \mathbf{1}$ corresponds to a solution of our timetabling problem. Now, the solution itself is obtained as relation of type $[\mathfrak{S} \leftrightarrow \mathfrak{G}]$ by applying the function *rel*, i.e., by $rel(\mathsf{M}p)$.

Each of the relational functions we have presented so far easily can be translated into the programming language of RelView. Using the tool, we have solved the original problem posed to us by the university administration. However, the input and output relations are too big to be presented here. Therefore, in the following we consider a smaller example to demonstrate our approach.

Example 3.1. We consider a set $\mathfrak{S}$ of only 10 subjects, namely mathematics (Ma), german (Ge), english (En), history (Hi), physics (Ph), chemistry (Che), biology (Bio), geography (Geo), arts (Ar) and physical education (Pe), which have to be distributed to the six groups A, B, C, D, E and F. The groups are divided into the blocks 1 and 2 via a relation Q and this immediately leads to the relation $B = QQ^{\mathsf{T}} : \mathfrak{G} \leftrightarrow \mathfrak{G}$ that specifies whether two groups belong to the same block. As (Boolean) RelView-matrices Q and B look as follows, where a black square means entry 1 and a white square means entry 0:

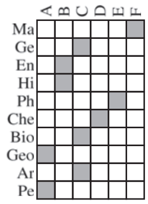
$$D = \begin{array}{c|cc|} & 1 & 2 \\ \hline \mathrm{A} & 1 & 0 \\ \mathrm{B} & 1 & 0 \\ \mathrm{C} & 1 & 0 \\ \mathrm{D} & 0 & 1 \\ \mathrm{E} & 0 & 1 \\ \mathrm{F} & 0 & 1 \\ \hline \end{array} \qquad B = \begin{array}{c|cccccc|} & \mathrm{A} & \mathrm{B} & \mathrm{C} & \mathrm{D} & \mathrm{E} & \mathrm{F} \\ \hline \mathrm{A} & 1 & 1 & 1 & 0 & 0 & 0 \\ \mathrm{B} & 1 & 1 & 1 & 0 & 0 & 0 \\ \mathrm{C} & 1 & 1 & 1 & 0 & 0 & 0 \\ \mathrm{D} & 0 & 0 & 0 & 1 & 1 & 1 \\ \mathrm{E} & 0 & 0 & 0 & 1 & 1 & 1 \\ \mathrm{F} & 0 & 0 & 0 & 1 & 1 & 1 \\ \hline \end{array}$$

We further consider the first two tables at the beginning of this section, that assign one time slot to every group $A, B, \ldots, F$ for each of the three years. The three relations J, N and B, where J and N are shown in the following pictures as RelView-matrices, constitute the input of our exemplary timetabling problem. From the two RelView-matrices we see. for instance, that the subjects mathematics and physics constitute an often selected combination and the subjects history and chemistry are hardly ever combined.

$$J = \begin{array}{c|cccccccccc|} & \mathrm{Ma} & \mathrm{Ge} & \mathrm{En} & \mathrm{Hi} & \mathrm{Ph} & \mathrm{Che} & \mathrm{Bio} & \mathrm{Geo} & \mathrm{Ar} & \mathrm{Pe} \\ \hline \mathrm{Ma} & 0&0&0&0&1&1&0&0&0&0 \\ \mathrm{Ge} & 0&0&1&0&0&0&0&0&0&0 \\ \mathrm{En} & 0&1&0&0&0&0&0&0&0&1 \\ \mathrm{Hi} & 0&0&0&0&0&0&0&1&1&0 \\ \mathrm{Ph} & 1&0&0&0&0&1&0&0&0&0 \\ \mathrm{Che} & 1&0&0&0&1&0&0&0&0&0 \\ \mathrm{Bio} & 0&0&0&0&0&0&0&1&0&1 \\ \mathrm{Geo} & 0&0&0&1&0&0&1&0&0&0 \\ \mathrm{Ar} & 0&0&0&1&0&0&0&0&0&0 \\ \mathrm{Pe} & 0&0&1&0&0&0&1&0&0&0 \\ \hline \end{array} \qquad N = \begin{array}{c|cccccccccc|} & \mathrm{Ma} & \mathrm{Ge} & \mathrm{En} & \mathrm{Hi} & \mathrm{Ph} & \mathrm{Che} & \mathrm{Bio} & \mathrm{Geo} & \mathrm{Ar} & \mathrm{Pe} \\ \hline \mathrm{Ma} & 1&1&0&0&0&0&1&1&1&1 \\ \mathrm{Ge} & 1&1&0&0&1&1&1&0&1&0 \\ \mathrm{En} & 0&0&1&1&0&1&0&1&1&0 \\ \mathrm{Hi} & 0&0&1&1&1&1&0&0&0&1 \\ \mathrm{Ph} & 0&1&0&1&1&0&0&1&1&0 \\ \mathrm{Che} & 0&1&1&1&0&1&1&0&1&1 \\ \mathrm{Bio} & 1&1&0&0&0&1&1&0&1&0 \\ \mathrm{Geo} & 1&0&1&0&1&0&0&1&0&1 \\ \mathrm{Ar} & 1&1&1&0&1&1&1&0&1&1 \\ \mathrm{Pe} & 1&0&0&1&0&1&0&1&1&1 \\ \hline \end{array}$$

We have used the RelView tool to generate in a first step the membership-relation $\mathsf{M} : \mathfrak{S} \times \mathfrak{G} \leftrightarrow [\mathfrak{S} \leftrightarrow \mathfrak{G}]$ of size 60×2^{60} for this example and, afterwards, to determine the vector $t = \Phi(\mathsf{M})^{\mathsf{T}}$ of length 2^{60} by translating the definition of Φ into its programming language. The result showed that t has exactly 144 1-entries, which means that there are exactly 144 solutions for the given timetabling problem, represented by 144 columns of the membership-relation M. Selecting a point p contained in t and defining v as composition $\mathsf{M}p$, a vector of length 60 and its corresponding relation $S = rel(v) : \mathfrak{S} \leftrightarrow \mathfrak{G}$

of size 10×6 have been computed such that the latter is a solution of our timetabling problem. Here is the RelView-picture of the solution S:

$S =$ 

Using the composition $\mathsf{M}\, inj(t)^{\mathsf{T}}$ we even have been able to compute the list of all solutions, represented as a relation with 60 rows and 144 columns. This relation is too large to be depicted here.

4. Computing solutions up to isomorphism

If the university timetabling problem of Section 3 is solvable, there often exist a large number of solutions. To be able to evaluate and compare the solutions, it is useful to examine them for isomorphism and consider only one solution of a large set of very similar ones. In this section, we will show how this can be achieved. First, we will present a reasonable definition of isomorphism between solutions, based on the sets of combinable and restricted combinable pairs of subjects. For a given solution S, we call two subjects *combinable* if they can be studied without overlappings, which means that S assigns the subjects to different groups of the same block. Two subjects that are assigned to groups of different blocks are called *restricted combinable*. The following theorem gives relation-algebraic expressions that specify the combinable and restricted combinable pairs of subjects, respectively.

Theorem 4.1. *Assume the input $B : \mathfrak{G} \leftrightarrow \mathfrak{G}$ and the solution $S : \mathfrak{S} \leftrightarrow \mathfrak{G}$ of the university timetabling problem and define the two relations $co(S)$ and $reco(S)$ of type $[\mathfrak{S} \leftrightarrow \mathfrak{S}]$ by $co(S) = S(B \cap \overline{\mathsf{I}})S^{\mathsf{T}}$ and $reco(S) = S\,\overline{B}\,S^{\mathsf{T}}$. Then it holds for all $s, s' \in \mathfrak{S}$ that $co(S)_{s,s'}$ if and only if s and s' are combinable and $reco(S)_{s,s'}$ if and only if s and s' are restricted combinable.*

Proof. Given arbitrary elements $s, s' \in \mathfrak{S}$, it holds that

$$\begin{aligned} s \text{ and } s' \text{ are combinable} &\iff \exists\, g, g' : S_{s,g} \wedge S_{s',g'} \wedge g \neq g' \wedge B_{g,g'} \\ &\iff \exists\, g, g' : S_{s,g} \wedge S_{s',g'} \wedge (\overline{\mathsf{I}} \cap B)_{g,g'} \\ &\iff (S(B \cap \overline{\mathsf{I}})S^{\mathsf{T}})_{s,s'} \end{aligned}$$

and in a similar way the second claim is verified. □

Based on the above relational functions *co* and *reco*, we are now in the position formally to define our notion of isomorphism.

Definition 4.1. Two solutions S and S' of the university timetabling problem are called isomorphic if $co(S) = co(S')$ and $reco(S) = reco(S')$. In this case we write $S \cong S'$.

Recall that a relation P for which domain and range coincide is a permutation relation if and only if P as well as its transpose P^{T} are mappings in the relation-algebraic sense, i.e., $PP^{\mathsf{T}} = P^{\mathsf{T}}P = \mathsf{I}$. As we will see later, we can use block-preserving permutation relations to create isomorphic solutions from a given solution of the university timetabling problem. This specific kind of permutation relations is introduced as follows.

Definition 4.2. Given the relation B as in Theorem 4.1, we call a permutation relation $P : \mathfrak{G} \leftrightarrow \mathfrak{G}$ block-preserving if $B \subseteq PBP^{\mathsf{T}}$.

In words, the inclusion $B \subseteq PBP^{\mathsf{T}}$ means that if two groups belong to the same block, then this holds for their images under the permutation relation, too. The following theorem clarifies the relationship between isomorphism of solutions and block-preserving permutation relations. Its first part is an immediate consequence of the definitions, the more complicated proof of the second part is presented in Appendix A of the paper.

Theorem 4.2. (a) *If the relation S is a solution of the university timetabling problem and P is a block-preserving permutation relation, then SP is also a solution of this problem.*

(b) *For two solutions S and S' of the university timetabling problem we have $S \cong S'$ if and only if there exists a block-preserving permutation relation P such that $S' = SP$.*

To determine the set of all solutions which are isomorphic to a given solution, we rather follow the technique of Section 3 for computing solutions of the university timetabling problem. Hence, we start with the following theorem that states a relation-algebraic expression which depends on a vector $v = vec(P)$ and evaluates to the universal relation L of type $[\mathbf{1} \leftrightarrow \mathbf{1}]$ if and only if v is the corresponding vector of a block-preserving permutation relation P. In the theorem, by $\pi : \mathfrak{G} \times \mathfrak{G} \leftrightarrow \mathfrak{G}$ and $\rho : \mathfrak{G} \times \mathfrak{G} \leftrightarrow \mathfrak{G}$ we denote the projection relations of $\mathfrak{G} \times \mathfrak{G}$.

Theorem 4.3. *Let the relation B be as in Theorem 4.1. Furthermore, assume $P : \mathfrak{G} \leftrightarrow \mathfrak{G}$ and a vector $v : \mathfrak{G} \times \mathfrak{G} \leftrightarrow \mathbf{1}$ such that $v = vec(P)$. Then P is a block-preserving permutation relation if and only if*

$$\overline{\mathsf{L}(\mathsf{L}\,\overline{\pi^{\mathsf{T}} v} \cup \mathsf{L}\,\overline{\rho^{\mathsf{T}} v} \cup (v \cap ((\mathsf{I} \parallel \overline{\mathsf{I}}) \cup (\overline{\mathsf{I}} \parallel \mathsf{I}) \cup (B \parallel \overline{B}))v))} = \mathsf{L}.$$

Proof. Like in the proof of Theorem 3.1, we can show the following two equivalences by combining the assumption $v = vec(P)$ with the two properties (2) and (7) of Section 2:

$$\begin{aligned} P \text{ injective} &\iff \overline{\mathsf{L}((\overline{\mathsf{I}} \parallel \mathsf{I})v \cap v)} = \mathsf{L}, \\ P \text{ surjective} &\iff \overline{\mathsf{L}\,\overline{\rho^{\mathsf{T}} v}} = \mathsf{L}. \end{aligned}$$

Using additionally the relation-algebraic equations for specifying univalence and totality of relations that have been given in the proof of Theorem 3.2 for the relation P as well as its corresponding vector $v = vec(P)$, we obtain that P is a permutation relation if and only if

$$\overline{\mathsf{L}((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v)} \cap \overline{\mathsf{L}\,\overline{\pi^{\mathsf{T}} v}} \cap \overline{\mathsf{L}((\overline{\mathsf{I}} \parallel \mathsf{I})v \cap v)} \cap \overline{\mathsf{L}\,\overline{\rho^{\mathsf{T}} v}} = \mathsf{L}.$$

Supposing this equation to hold, P is a mapping and we are able to transform the condition $B \subseteq PBP^{\mathsf{T}}$ as follows:

$$\begin{aligned} B \subseteq PBP^{\mathsf{T}} &\iff BP \subseteq PB && P \text{ mapping} \\ &\iff B^{\mathsf{T}}P \subseteq PB && B \text{ symmetric} \\ &\iff B\,\overline{PB} \subseteq \overline{P} && \text{Schröder rule} \\ &\iff BP\,\overline{B} \subseteq \overline{P} && [15, \text{Prop. } 4.2.4] \\ &\iff vec(BP\,\overline{B}) \subseteq vec(\overline{P}) && vec \text{ isomorphism} \\ &\iff (B \parallel \overline{B}^{\mathsf{T}})vec(P) \subseteq \overline{vec(P)} && vec \text{ isomorphism } (7) \\ &\iff (B \parallel \overline{B}^{\mathsf{T}})v \subseteq \overline{v} && v = vec(P) \\ &\iff \overline{\mathsf{L}(v \cap (B \parallel \overline{B})v)} = \mathsf{L} && \text{due to } (2) \end{aligned}$$

If we intersect the left-hand side of the last equation of this derivation with the left-hand side of the above equation, we get that P is a block-preserving permutation relation if and only if

$$\overline{\mathsf{L}((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v)} \cap \overline{\mathsf{L}\,\overline{\pi^{\mathsf{T}} v}} \cap \overline{\mathsf{L}((\overline{\mathsf{I}} \parallel \mathsf{I})v \cap v)} \cap \overline{\mathsf{L}\,\overline{\rho^{\mathsf{T}} v}} \cap \overline{\mathsf{L}(v \cap (B \parallel \overline{B})v)} = \mathsf{L}.$$

The last steps of the proof are rather the same as in the case of Theorem 3.2. We use a de Morgan low, introduce two universal relations for type adaption and apply commutativity of join and a distributivity law. □

Like in Section 3, from Theorem 4.3 we immediately obtain the following function Ψ on relations that is defined using the variable X, constant relations, complements, joins, meets and left-compositions only:

$$\Psi(X) = \overline{\mathsf{L}(\mathsf{L}\,\overline{\pi^{\mathsf{T}} X} \cup \mathsf{L}\,\overline{\rho^{\mathsf{T}} X} \cup (X \cap ((\mathsf{I} \parallel \overline{\mathsf{I}}) \cup (\overline{\mathsf{I}} \parallel \mathsf{I}) \cup (B \parallel \overline{B}))X))}.$$

As a consequence, the application of the vector predicate Ψ to the membership-relation $\mathsf{M} : \mathfrak{G} \times \mathfrak{G} \leftrightarrow [\mathfrak{G} \leftrightarrow \mathfrak{G}]$ and a transposition of the result yield a vector

$$b = \Psi(\mathsf{M})^{\mathsf{T}} \tag{12}$$

of type $[[\mathfrak{G} \leftrightarrow \mathfrak{G}] \leftrightarrow \mathbf{1}]$ that specifies exactly those columns of M which are corresponding vectors of block-preserving permutation relations. According to the technique we have presented in Section 2, hence, a column-wise representation of the set $\mathfrak{P}$ of all block-preserving permutation relations (as a subset of the type $[\mathfrak{G} \leftrightarrow \mathfrak{G}]$ of the relations on the set $\mathfrak{G}$) is given by the relation

$$E = \mathsf{M}\, inj(b)^{\mathsf{T}} \tag{13}$$

of type $[\mathfrak{G}\times\mathfrak{G} \leftrightarrow \mathfrak{P}]$. To be more precise, the function $P \mapsto vec(P)$ constitutes a one-to-one correspondence between $\mathfrak{P}$ and the set of all columns of E (where each column is considered as a vector of type $[\mathfrak{G}\times\mathfrak{G} \leftrightarrow \mathbf{1}]$). In the remainder of the section, we show how the relation E of (13) can be used to compute the set of all solutions isomorphic to a given solution S. The decisive property is presented in the next theorem. It states a relation-algebraic expression for the column-wise representation of all solutions isomorphic to S, where, however, in contrast to the notion introduced in Section 2, multiple occurrences of columns are allowed. In the proof, we use the notation $R^{(y)}$ for the y-column of R as introduced in Section 2.

Theorem 4.4. *Assume $S : \mathfrak{G} \leftrightarrow \mathfrak{G}$ to be a solution of the university timetabling problem, E as the relation introduced in (13) and $I_S : \mathfrak{G}\times\mathfrak{G} \leftrightarrow \mathfrak{P}$ to be defined as $I_S = (S \parallel \mathsf{I})E$. Then every block-preserving permutation relation $X \in \mathfrak{P}$ leads to a solution $rel(I_S^{(X)})$ of the university timetabling problem such that $rel(I_S^{(X)}) \cong S$, and for every further solution S' with $S' \cong S$ there exists a block-preserving permutation relation $Y \in \mathfrak{P}$ such that $vec(S') = I_S^{(Y)}$.*

Proof. To prove the first statement, we assume $X \in \mathfrak{P}$. Then, the one-to-one correspondence between the set $\mathfrak{P}$ and the set of all columns of E shows the existence of a block-preserving permutation relation $P : \mathfrak{G} \leftrightarrow \mathfrak{G}$ fulfilling $E^{(X)} = vec(P)$. This leads to the equation

$$I_S^{(X)} = ((S \parallel \mathsf{I})E)^{(X)} = (S \parallel \mathsf{I})E^{(X)} = (S \parallel \mathsf{I})vec(P) = vec(S\,P)$$

because of an obvious property of column selection in the case of a composition of relations and property (7) of Section 2. The derived equation in turn shows

$$rel(I_S^{(X)}) = rel(vec(SP)) = SP$$

and, finally, Theorem 4.2(a) leads to the desired result $rel(I_S^{(X)}) \cong S$,

For a proof of the second claim, we start with a further solution S' such that $S' \cong S$. Then Theorem 4.2(b) yields a block-preserving permutation relation $P : \mathfrak{G} \leftrightarrow \mathfrak{G}$ with $S' = SP$. Next, we apply again property (7) and get

$$vec(S') = vec(SP) = (S \parallel \mathsf{I})vec(P).$$

Since the relation $E : \mathfrak{G}\times\mathfrak{G} \leftrightarrow \mathfrak{P}$ column-wisely represents the set $\mathfrak{P}$ of all block-preserving permutation relations, there exists again $Y \in \mathfrak{P}$ such that for the Y-column $E^{(Y)}$ we obtain $vec(P) = E^{(Y)}$. Combining this with the above result and the definition of I_S yields

$$vec(S') = (S \parallel \mathsf{I})vec(P) = (S \parallel \mathsf{I})E^{(Y)} = ((S \parallel \mathsf{I})E)^{(Y)} = I_S^{(Y)}$$

and we are done. □

Now, we use Theorem 4.4 and describe a procedure for the computation of the set of all solutions of the university timetabling problem up to isomorphism. It easily can be implemented in RelView. In a first step, we determine the vector $t : [\mathfrak{G} \leftrightarrow \mathfrak{G}] \leftrightarrow \mathbf{1}$ of (11) that specifies those columns of the membership-relation $\mathsf{M} : \mathfrak{G}\times\mathfrak{G} \leftrightarrow [\mathfrak{G} \leftrightarrow \mathfrak{G}]$ which correspond to solutions of the timetabling problem and the relation $E : \mathfrak{G}\times\mathfrak{G} \leftrightarrow \mathfrak{P}$ of (13) that column-wisely enumerates the block-preserving permutation relations. Selecting a point p contained in t, we then compute a single solution S as described in Section 3 and the column-wise representation I_S of all solutions isomorphic to S. With

$$t' = t \cap syq(\mathsf{M}, I_S)\mathsf{L} \tag{14}$$

we obtain a vector of type $[[\mathfrak{G} \leftrightarrow \mathfrak{G}] \leftrightarrow \mathbf{1}]$ that specifies all columns of M which correspond to solutions isomorphic to S. This follows from the equivalence

$$\begin{aligned} (t \cap syq(\mathsf{M}, I_S)\mathsf{L})_X &\iff t_X \wedge \exists\, Y : syq(\mathsf{M}, I_S)_{X,Y} & \\ &\iff t_X \wedge \exists\, Y : \mathsf{M}^{(X)} = I_S^{(Y)} & \text{see Section 2} \\ &\iff rel(\mathsf{M}^{(X)}) \cong S & \text{Theorem 4.4} \end{aligned}$$

for all relations $X : \mathfrak{S} \leftrightarrow \mathfrak{G}$, where Y ranges over the set $\mathfrak{P}$ of all block-preserving permutation relations. By modifying t to $(t \cap \overline{t'}) \cup p$ with t' from (14), we can remove all solutions isomorphic to S from the solution vector t, except S itself. Successive applications of this technique leads to a vector of type $[[\mathfrak{S} \leftrightarrow \mathfrak{G}] \leftrightarrow \mathbf{1}]$ that, finally, specifies by its 1-entries exactly one element of each set of isomorphic solutions.

Experience has shown that in most cases the number of solutions can be reduced considerable if we restrict us to non-isomorphic ones. In particular, there exist exactly 1296 block-preserving permutation relations for the original timetabling problem of our university administration with 9 groups and 3 blocks, so that for each of its solutions there are up to 1296 isomorphic solutions. Regarding Example 3.1, where we deal with 2 blocks and 6 groups only, there are exactly 72 block-preserving permutation relations, and the 144 solutions of the timetabling problem of the example can be reduced to only two solutions which are not isomorphic.

5. A more simple approach for timetabling

Applying the RelView-implementation of the algorithm of Section 3 to the input data delivered by the university administration, we obtained the solution vector t of (11) to be empty. Since this meant that there exists no solution, in accordance with the university administration we changed the three categories of possible combinations slightly and applied the RelView-program to the new relations J and N. Again we got $t = \mathsf{O}$. Repeating this process several times, we finally found a non-empty t. But thus we had changed the categories in such a way that a further perpetuation of the original trisection of the combinations into "very frequently", "less common" and "hardly ever selected" seemed inappropriate since "less common" was almost empty. So, we decided to drop the category "less common" and to work with the remaining two categories only.

Because of the cumbersome procedure and the fact that two categories seem to suffice, we also checked whether the group/block division technique still is reasonable and developed, for the purpose of testing and in collaboration with our university administration, an alternative and more simple model for timetabling. In the new model, the development of which orientates on the approach of [13,14] sketched in the introduction, there are four disjoint time slots of equal size, denoted as t_1, t_2, t_3 and t_4, such that none of the subjects requires more than two of them. All so-called small subjects entirely can take place in each of these "base time slots". To treat the remaining large subjects, too, we introduced two further time slots t_5 and t_6, where t_5 consists of the hours of t_1 and t_2 and t_6 consists of the hours of t_3 and t_4. Hence, the large subjects can take place in these additional time slots. Of course, this model led to time conflicts between certain time slots. A timetable that enables a three years duration of study for the very frequently combinations, now is given by a function from the set of subjects to the set of time slots such that the following two properties hold:

(a) For all subjects s and time slots t, if s is mapped to t then t is available for s.
(b) There are no time conflicts between the courses of two different subjects if the latter constitute a combination of the first category.

To formalize and generalize also this informal description to an abstract university timetabling problem, we again assume $\mathfrak{S}$ to denote a finite set of subjects, but now, instead of $\mathfrak{G}$ and $\mathfrak{B}$ as used in the original approach, $\mathfrak{T}$ to denote a finite set of time slots. For modeling the partitioning of the pairs of subjects into the two categories, we assume a relation $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$ to be at hand such that

$$F_{s,s'} \iff s, s' \text{ is a combination of the first category} \tag{15}$$

for all subjects $s, s' \in \mathfrak{S}$. Then F is symmetric and irreflexive, where the latter property follows from the fact that combinations have to consist of two different subjects. It should be remarked that the relation F suffices for completely describing the two categories, since the symmetric and irreflexive relation $\overline{F} \cap \overline{\mathsf{I}}$ exactly specifies the pairs of different subjects which are hardly ever selected. Besides F, we assume an availability relation $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$ that specifies availability, i.e., is defined component-wise by

$$A_{s,t} \iff s \text{ can take place in } t \tag{16}$$

for all subjects $s \in \mathfrak{S}$ and time slots $t \in \mathfrak{T}$. And, finally, we assume a onflict relation $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$ such that

$$C_{t,t'} \iff t \text{ and } t' \text{ are in time conflict} \tag{17}$$

for all time slots $t, t' \in \mathfrak{T}$, where time slots are in time conflict if and only if they contain common hours, Note, that because of this interpretation C is a reflexive and symmetric relation. Considering the relations of (15) to (17) as input of the revised university timetabling problem, a solution relation-algebraically can be defined as follows.

Definition 5.1. Given the three input relations $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$, $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$ and $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{T}$ is a solution of the revised university timetabling problem if $S \subseteq A$, $FSC \subseteq \overline{S}$, $S^{\mathsf{T}}S \subseteq \mathsf{I}$ and $\mathsf{L} \subseteq S\mathsf{L}$.

That $S \subseteq A$ formalizes the above property (a) is trivial. In case of $FSC \subseteq \overline{S}$ and property (b) this is shown as in the case of the original model of Section 3. Finally, $S^{\mathsf{T}}S \subseteq \mathsf{I}$ and $\mathsf{L} \subseteq S\mathsf{L}$ specify again S to be a mapping. If only the first three inclusions of Definition 5.1 hold, the univalent relation S is called a *partial solution* of the revised university timetabling problem.

Following exactly the pattern of Section 3, in the remainder of this section we develop a relation-algebraic algorithm for solving the revised university timetabling problem. Here is the analogon of Theorem 3.1. In its last inclusion $\pi : \mathfrak{S}\times\mathfrak{T} \leftrightarrow \mathfrak{S}$ denotes the first projection relation of $\mathfrak{S}\times\mathfrak{T}$.

Theorem 5.1. *Assume the three input relations $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$, $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$ snd $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{T}$ and a vector $v : \mathfrak{S}\times\mathfrak{T} \leftrightarrow \mathbf{1}$ such that $v = vec(S)$. Then S is a solution of the revised university timetabling problem if and only if $v \subseteq vec(A)$, $(F \parallel C)v \subseteq \overline{v}$, $(\mathsf{I} \parallel \overline{\mathsf{I}})v \subseteq \overline{v}$ and $\mathsf{L} \subseteq \pi^{\mathsf{T}}v$.*

Proof. The claim follows from the fact that the nth inclusion of Definition 5.1 is equivalent to the nth inclusion of the theorem $(1 \le n \le 4)$. The first case is trivial, the second one shown by

$$\begin{array}{rlll} FSC \subseteq \overline{S} & \Longleftrightarrow & vec(FSC) \subseteq vec(\overline{S}) & vec \text{ isomorphism} \\ & \Longleftrightarrow & vec(FSC) \subseteq \overline{vec(S)} & vec \text{ isomorphism} \\ & \Longleftrightarrow & (F \parallel C^{\mathsf{T}})vec(S) \subseteq \overline{vec(S)} & \text{due to (7)} \\ & \Longleftrightarrow & (F \parallel C^{\mathsf{T}})v \subseteq \overline{v} & v = vec(S) \\ & \Longleftrightarrow & (F \parallel C)v \subseteq \overline{v} & C \text{ is symmetric} \end{array}$$

and for the remaining cases see the proof of Theorem 3.1. □

Due to this theorem, we are again in a position to present a relation-algebraic expression that depends on a vector $v = vec(S)$ and evaluates to the universal relation L of type $[\mathbf{1} \leftrightarrow \mathbf{1}]$ if and only if v represents a solution S of the revised university timetabling problem. The corresponding next theorem is the analogon of Theorem 3.2.

Theorem 5.2. *Let again the relations F, A, C, S, v and π be as in Theorem 5.1. Then S is a solution of the revised university timetabling problem if and only if*

$$\overline{\mathsf{L}((v \cap \overline{vec(A)}) \cup ((F \parallel C)v \cap v) \cup ((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v) \cup \mathsf{L}\,\overline{\pi^{\mathsf{T}}v})} = \mathsf{L}.$$

Proof. Property (2) of Section 2 implies the equivalences

$$\begin{array}{rcl} v \subseteq vec(A) & \Longleftrightarrow & \overline{\mathsf{L}(v \cap \overline{vec(A)})} = \mathsf{L}, \\ (F \parallel C)v \subseteq \overline{v} & \Longleftrightarrow & \overline{\mathsf{L}((F \parallel C)v \cap v)} = \mathsf{L} \end{array}$$

and from the proof of Theorem 3.2 we know already

$$\begin{array}{rcl} (\mathsf{I} \parallel \overline{\mathsf{I}})v \subseteq \overline{v} & \Longleftrightarrow & \overline{\mathsf{L}((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v)} = \mathsf{L}, \\ \mathsf{L} \subseteq \pi^{\mathsf{T}}v & \Longleftrightarrow & \overline{\mathsf{L}\,\overline{\pi^{\mathsf{T}}v}} = \mathsf{L}. \end{array}$$

Combining these four equivalences with Theorem 5.1, we get that the relation S is a solution of the revised university timetabling problem if and only if it holds

$$\overline{\mathsf{L}(v \cap \overline{vec(A)})} \cap \overline{\mathsf{L}((F \parallel C)v \cap v)} \cap \overline{\mathsf{L}((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v)} \cap \overline{\mathsf{L}\,\overline{\pi^{\mathsf{T}}v}} = \mathsf{L}.$$

The remaining steps are as in the proof of Theorem 3.2. □

Analogous to the approach of Section 3. the left-hand side of the equation of Theorem 5.2 leads to a vector predicate on relations, viz.

$$\Phi(X) = \overline{\mathsf{L}((X \cap \overline{vec(A)}\,\mathsf{L}) \cup ((F \parallel C)X \cap X) \cup ((\mathsf{I} \parallel \overline{\mathsf{I}})X \cap X) \cup \mathsf{L}\,\overline{\pi^{\mathsf{T}}X})},$$

which, in turn, with the specific argument $\mathsf{M} : \mathfrak{S}\times\mathfrak{T} \leftrightarrow [\mathfrak{S} \leftrightarrow \mathfrak{T}]$, yields a vector

$$t = \Phi(\mathsf{M})^{\mathsf{T}} \tag{18}$$

of type $[[\mathfrak{S}\leftrightarrow\mathfrak{T}]\leftrightarrow\mathbf{1}]$ such that for all relations $X:\mathfrak{S}\leftrightarrow\mathfrak{T}$ the entry t_X is 1 if and only if the X-column $\mathsf{M}^{(X)}:\mathfrak{S}\times\mathfrak{T}\leftrightarrow\mathbf{1}$ of the membership-relation M corresponds to a solution of the revised university timetabling problem. Also the further steps to obtain from t one (or even all) single solution(s) are as in Section 3.

6. An alternative method for computing solutions

In this section, we use the equivalences shown in the proof of the main result of Section 5 and sketch yet another relation-algebraic procedure for solving the revised university timetabling problem of the last section. As a preparatory step, we prove the following fact about vectors.

Lemma 6.1. *For all vectors v and w we have $v\subseteq w$ if and only if $vv^{\mathsf{T}}\subseteq ww^{\mathsf{T}}$.*

Proof. The direction "$\Rightarrow$" is trivial. The same holds for direction "$\Leftarrow$" if $v=\mathsf{O}$. To prove "$\Leftarrow$" in case of $v\neq\mathsf{O}$, we start with $\mathsf{L}v=\mathsf{L}v\mathsf{L}=\mathsf{L}$, which follows from the vector property and the Tarski rule in combination with $v\neq\mathsf{O}$. Hence, we have $v^{\mathsf{T}}\mathsf{L}=\mathsf{L}$ (surjectivity of v). Now, the result follows from $v=v\mathsf{L}=vv^{\mathsf{T}}\mathsf{L}\subseteq ww^{\mathsf{T}}\mathsf{L}\subseteq w\mathsf{L}=w$ using the vector properties of v and w, the surjectivity of v and the assumption $vv^{\mathsf{T}}\subseteq ww^{\mathsf{T}}$. $\square$

Given an undirected graph, a set V of vertices is called stable if no two vertices from it are adjacent. Supposing G as the graph's symmetric adjacency relation, this means that for all $x\in V$ and $y\in V$ it follows $\overline{G}_{x,y}$. If the set V is represented by a vector v, then a little calculation shows that V is stable if and only if $Gv\subseteq\overline{v}$. As already mentioned in the introduction, our alternative method of solution bases on stable sets. The following theorem relation-algebraically describes the construction of the graph's adjacency relation from the input of the revised university timetabling problem.

Theorem 6.1. *Let again the relations F, A, C, S and v be as in Theorem 5.1. Then S is a partial solution of the revised university timetabling problem if and only if $((\overline{vec(A)vec(A)^{\mathsf{T}}})\cup(F\parallel C)\cup(\mathsf{I}\parallel\overline{\mathsf{I}}))v\subseteq\overline{v}$.*

Proof. We start with the first demand on S and transform it as follows:

$$
\begin{array}{rll}
S\subseteq A \iff & v\subseteq vec(A) & \text{proof of Theorem 5.1}\\
\iff & vv^{\mathsf{T}}\subseteq vec(A)vec(A)^{\mathsf{T}} & \text{Lemma 6.1}\\
\iff & \overline{vec(A)vec(A)^{\mathsf{T}}}\,v\subseteq\overline{v} & \text{Schröder rule}
\end{array}
$$

Due to the proof of Theorem 5.1, we have the equivalence

$$FSC\subseteq\overline{S}\iff(F\parallel C)v\subseteq\overline{v}$$

for the second demand on S and the equivalence

$$S^{\mathsf{T}}S\subseteq\mathsf{I}\iff(\mathsf{I}\parallel\overline{\mathsf{I}})v\subseteq\overline{v}$$

for its third demand. By simple laws of lattice theory, the conjunction of the three just calculated inclusions between vectors is equivalent to the inclusion

$$\overline{vec(A)vec(A)^{\mathsf{T}}}\,v\cup(F\parallel C)v\cup(\mathsf{I}\parallel\overline{\mathsf{I}})v\subseteq\overline{v}$$

and an application of $\cup$-distributivity, finally, shows the claim. $\square$

If we use the abbreviation $a=vec(A)$ and define by

$$G=\overline{aa^{\mathsf{T}}}\cup(F\parallel C)\cup(\mathsf{I}\parallel\overline{\mathsf{I}})\tag{19}$$

a relation of type $[\mathfrak{S}\times\mathfrak{T}\leftrightarrow\mathfrak{S}\times\mathfrak{T}]$, then the above remark and Theorem 6.1 say that the vector $v:\mathfrak{S}\times\mathfrak{T}\leftrightarrow\mathbf{1}$ corresponds to a partial solution of the revised university timetabling problem if and only if v represents a stable vertex set of the graph g with the vertex set $\mathfrak{S}\times\mathfrak{T}$ and the adjacency relation G of (19). Since the inclusion $Gv\subseteq\overline{v}$ is equivalent to the equation $\overline{\mathsf{L}(Gv\cap v)}=\mathsf{L}$ and the definition

$$\Psi(X)=\overline{\mathsf{L}(GX\cap X)}$$

obviously yields again a vector predicate, the vector

$$m = gre(\mathsf{C}, \Psi(\mathsf{M})^{\mathsf{T}}) \tag{20}$$

of type $[[\mathfrak{S} \leftrightarrow \mathfrak{T}] \leftrightarrow \mathbf{1}]$ represents the set $\mathcal{S}_g$ of all maximum stable vertex sets of g. In definition (20), we use a membership-relation $\mathsf{M} : \mathfrak{S}\times\mathfrak{T} \leftrightarrow [\mathfrak{S} \leftrightarrow \mathfrak{T}]$, a *size-comparison relation*[1] $\mathsf{C} : [\mathfrak{S} \leftrightarrow \mathfrak{T}] \leftrightarrow [\mathfrak{S} \leftrightarrow \mathfrak{T}]$ such that $\mathsf{C}_{X,Y}$ if and only if $|X| \leq |Y|$ for all sets $X, Y \in 2^X$, and a function $gre(Q, R) = R \cap \overline{\overline{Q}^{\mathsf{T}} R}$ that column-wisely computes greatest elements w.r.t. the quasi order Q (for the latter, see, e.g., [4] for details).

Using a similar procedure as in the case of the vector t of (18), the above vector m allows to decide whether the revised university timetabling problem is solvable. If $p \subseteq m$ is a point, then $\mathsf{M}p : \mathfrak{S}\times\mathfrak{T} \leftrightarrow \mathbf{1}$ corresponds to a partial solution of the problem. As a consequence, if the number of 1-entries of $\mathsf{M}p$ equals the cardinality of $\mathfrak{S}$, then this vector even corresponds to a (total) solution. Otherwise there are no (total) solutions but only strictly partial ones.

The relation G of (19) is symmetric and its second part $F \parallel C$ and third part $\mathsf{I} \parallel \overline{\mathsf{I}}$ are irreflexive. But irreflexivity of G does not hold in general. A little reflection shows that the 1-entries in the diagonal of its first part $\overline{aa^{\mathsf{T}}}$ exactly correspond to the pairs $\langle s, t\rangle$ for which $\overline{A}_{s,t}$ holds, i.e., for which t is not available for s. This fact allows to reduce the size of the problem. Instead G the (often considerable) smaller relation $G' = inj(\overline{d})\, G\, inj(\overline{d})^{\mathsf{T}}$ may be used as adjacency relation, where the vector $d = (\overline{aa^{\mathsf{T}}} \cap \mathsf{I})\mathsf{L} = \overline{a} = vec(\overline{A})$ represents the set $\langle s, t\rangle$ of pairs with $\overline{A}_{s,t}$. The correctness of the reduction follows from the fact that each vector v' with $G'v' \subset \overline{v'}$ exactly corresponds to a vector v with $Gv \subset \overline{v}$ via the two functions $v' \mapsto inj(a)^{\mathsf{T}} v$ and $v \mapsto inj(a)v'$.

7. Implementation and results

Relation algebra has a fixed and surprisingly small set of constants and operations which – in the case of finite carrier sets – can be implemented very efficiently. At the University of Kiel we have developed a visual computer system for the visualization and manipulation of relations and for relation-algebraic prototyping and programming, called RelView. The tool is written in the programming language C, uses reduced ordered BDDs for implementing relations and makes full use of the X-windows graphical user interface. Details and applications can be found, for instance, in [1–4].

The main purpose of the computer system RelView is the evaluation of relation-algebraic expressions. These expressions are constructed from the relations of the tool's workspace using pre-defined operations and tests, user-defined relational functions, and user-defined relational programs. A relational program is much like a function procedure in Pascal or Modula 2, except that it only uses relations as data type. It starts with a head line containing the program name and the list of the formal parameters, which stand for relations. Then the declaration of the local relational domains, functions and variables follows. Domain declarations can be used to introduce projection relations. The third part of a program is the body, a while-program over relations. As a program computes a value, finally, its last part consists of a return-clause, which is a relation-algebraic expression whose value after the execution of the body is the result. For example, the RelView-version of the vector predicate Φ used in Section 5 for solving the revised university timetabling problem looks as follows:

```
Phi(A,F,C,X)
  DECL Prod = PROD(F,C);
       pi, Q1, Q2, Q3, Q4
  BEG  pi = p-1(Prod);
       Q1 = X & -(vec(A) * L1n(X));
       Q2 = par(F,C) * X & X;
       Q3 = par(I(F),-I(C)) * X & X;
       Q4 = L(pi) * -(pi^ * X)
       RETURN -(Ln1(pi)^ * (Q1 | Q2 | Q3 | Q4))
  END.
```

In this relational program, the first declaration introduces `Prod` as a name for the direct product $\mathfrak{S}\times\mathfrak{T}$. Using `Prod`, the first projection relation is then computed and stored as `pi` by the first assignment of the body. The remaining part of the

[1] As in the case of membership-relations, a simple Boolean matrix implementation of size-comparison relations is unusable in practice, but the same does not hold if BDDs are used. In [2] a BDD-implementation of $\mathsf{C} : 2^X \leftrightarrow 2^X$ is presented, where the number of vertices is quadratic in the size of X. It also is part of RelView.

program consists of a direct translation of the expression defining Φ into RelView-notation, where `^`, `-`, `|`, `&` and `*` denote transposition, complement, union, meet and composition, the pre-defined operations `Ln1`, `L1n` and `L` compute for a relation $R : X \leftrightarrow Y$ the universal relations of type $[X \leftrightarrow \mathbf{1}]$, $[\mathbf{1} \leftrightarrow Y]$ and $[X \leftrightarrow Y]$, respectively, the pre-defined operation `I` yields for $R : X \leftrightarrow X$ the identity relation $\mathsf{I} : X \leftrightarrow X$ and the user-defined relational programs `vec` and `par` implement the functions *vec* and $\|$, respectively.

We have applied the RelView-program `Phi` to the university administration's original problem with 34 subjects and the six time slots t_1 to t_6 as described in Section 5. Since this led (using matrix terminology) to a membership-relation of size 204×2^{204} in the specification (18) of the solution vector t, the RelView tool has not been able to yield a result within an adequate time – despite the efficient BDD-implementation of relations it uses. In this situation, two facts helped us to reduce the problem size considerable and to obtain, finally, results within a few seconds only.

First, we noticed that there was only one large subject (chemistry, abbreviated as c) that required two time slots. Hence, the model with the six time slots was not appropriate in this case. Instead, chemistry was subdivided into two subjects c_1 and c_2, so that each of them had to be mapped to one of the four base time slots t_1, t_2, t_3 or t_4. This led to a modified input F', A', C' for the revised university timetable problem.[2]

(a) The type of the relation F' became $[\mathfrak{S}' \leftrightarrow \mathfrak{S}']$, where the set $\mathfrak{S}'$ is defined as $(\mathfrak{S} \setminus \{c\}) \cup \{c_1, c_2\}$. For all $s, s' \in \mathfrak{S} \setminus \{c\}$ we defined $F'_{s,s'}$ if and only if $F_{s,s'}$ and, in view of the "new" subjects, $F'_{c_i,s'}$ if and only if $F_{c,s'}$, respectively F'_{s',c_i} if and only if $F_{s',c}$. To guarantee, that c_1 and c_2 are assigned to different base time slots, we finally defined c_1, c_2 as a combination of the first category, which meant F'_{c_1,c_2} and F'_{c_2,c_1}.

(b) The relation A' could be defined as universal relation

$$A' = \mathsf{L} \tag{21}$$

of type $[\mathfrak{S}' \leftrightarrow \mathfrak{T}']$, where $\mathfrak{T}' = \{t_1, \ldots, t_4\}$, because now every subject could take place in every base time slot.

(c) Since the splitting of the subject chemistry abolished all conflicts between base time slots, finally, C' could be the identity relation of type $[\mathfrak{T}' \leftrightarrow \mathfrak{T}']$, i.e., we define

$$C' = \mathsf{I}. \tag{22}$$

By modifying the input relations in this way, the RelView-program `Phi` could be used to compute all solutions in reasonable time, since the size of the membership-relation has reduced to 140×2^{140}.

Besides the splitting of chemistry, we could use another property of the given problem to reduce the problem size even more. The four Romanian languages Spanish, Portuguese, Italian and French (abbreviated as s, p, i, f) formed an important clique in the graph with adjacency relation F. By demand of the curricula, each of these subjects must be combinable with the other three. Hence, the four languages of the set $\mathfrak{R} = \{s, p, i, f\}$ had to be assigned to four different base time slots. Predefining a base time slot for each of these subjects via an injective mapping $R : \mathfrak{R} \leftrightarrow \mathfrak{T}'$ that assigns to each Romanian language exactly one base time slot, we could reduce the set of subjects to $\mathfrak{S}'' = \mathfrak{S}' \setminus \mathfrak{R}$ by omitting the four Romanian languages. To consider the dependencies between the Romanian languages and other subjects, the input had to be modified again.

(a') The modification F'' of F' became the restriction of the relation F' to the set $\mathfrak{S}''$, i.e., with $v : \mathfrak{S}' \leftrightarrow \mathbf{1}$ as the vector representation of $\mathfrak{R}$ as subset of $\mathfrak{S}'$, we got $F'' : \mathfrak{S}'' \leftrightarrow \mathfrak{S}''$ by

$$F'' = inj(\,\overline{v}\,)\, F'\, inj(\,\overline{v}\,)^{\mathsf{T}}. \tag{23}$$

(b') In the refined model, obviously a subject $s \in \mathfrak{S}''$ could take place in a base time slot $t \in \mathfrak{T}'$ if and only if there is no $r \in \mathfrak{R}$ such that $F'_{s,r}$ and $R_{r,t}$. Based on this observation, a little reflection brought the new version

$$A'' = \overline{inj(\,\overline{v}\,)\, F'\, inj(v)^{\mathsf{T}}\, R} \tag{24}$$

of type $[\mathfrak{S}'' \leftrightarrow \mathfrak{T}']$ of the availability relation, where v is the vector introduced in (a').

(c') The removal of the four Romanian languages from $\mathfrak{S}'$ caused no conflicts between base time slots. As a consequence, the conflict relation furthermore could be the identity relation of type $[\mathfrak{T}' \leftrightarrow \mathfrak{T}']$, i.e.,

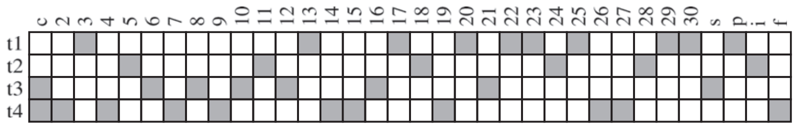
$$C'' = \mathsf{I}. \tag{25}$$

At this place it should be mentioned that, as in the cases (b) to (c') via (21) to (25), also the relation F' of (a) relation-algebraically may be specified by an expression. However, since F' is defined via a case distinction, this requires the use of disjoint unions and their injection relations, which is beyond the scope of this paper.

Since the size of the set $\mathfrak{S}''$ is 31, now the size of the membership-relation used in (18) is 124×2^{124}, which is a moderate size to solve this problem and compute all solutions within a few seconds. But this is in the strict sense the end of the story.

[2] Obviously, the modification also can be applied in case of more than one large subject.

To narrate the entire story, at the beginning – as in the case of the original approach – the two categories of combinations of subjects provided by the university administration led to an empty solution vector t. With the help of an additional RelView-program (for its development, see [2]), we then determined all maximum cliques of the graph with adjacency relation F'' since large cliques (especially cliques of more than four subjects) caused the impossibility to find solutions. Step-by-step 1-entrys of F'' had been changed to 0 to destroy as much as possible maximum cliques, until we obtained an input that led to a non-empty solution vector. The knowledge of the largest cliques was important for this process to modify the relation defining the categories in a goal-oriented way. We started the process with 133 combinations in the first category and reduced them to 119 until being successful. The latest version of the relation F'' led to 32 solutions of the revised university timetabling problem. The one that was chosen by the administration of our university is shown in the following picture. This 34×4 Boolean RelView-matrix has been obtained from the computed 31×4 matrix by going back to c instead of c_1 and c_2 (i.e., by replacing the c_1- and c_2-row by their union), then adding the implicit assignments of the four Romanian languages as four rows at the bottom, and, finally, by transposing the resulting 4×34 matrix to save space.



The chosen solution enables to study 418 of the $\frac{34 \cdot 33}{2} = 561$ possible combinations of subjects without any overlapping. It has been discussed in commissions of the two faculties concerned with the undergraduate education of secondary school teachers. Whereas the Faculty of Philosophy has decided to introduce the new model and the computed timetable, the Faculty of Mathematics and Natural Sciences refused this and developed its own timetable by modifying the hitherto timetable. An ultimate decision about the introduction in both faculties and the final form of the timetable still is missing.

8. Concluding remarks

In this paper, we have combined relation algebra and the RelView tool to specify and solve timetabling problems which should enable the undergraduate education of secondary school teachers at the University of Kiel within three years in the normal case. Only for combinations of subjects which are hardly ever selected a longer duration of study should be necessary. During the entire project the concise and very formal language of relation algebra and the plentifulness of relation-algebraic laws has been very helpful. Also RelView proved to be an ideal tool for the tasks to be solved. Systematic experiments helped us to get insight into the specific character of the problem and to develop the relation-algebraic formalizations. Particularly with regard to these activities the concise form of RelView-programs and the tool's visualization facilities have been of avail.

Decisive for solving the posed problems has been the notion of a vector predicate since, when applied to a "proper" relation R instead of a vector, such a function allows to test a certain property for all columns of R in parallel and to filter out exactly those one is interested in. Implicitely, vector predicates have been used since many years. But, to our knowledge, except [10,11] all former applications dealt with the test and column-wise computation of certain subsets of a base set (like the carrier set of a partial order in the case of maximal elements) or its power set (like that of the vertex set of a graph in the case of stable sets or cliques) only. The novelty of [10,11] and this paper is the combination of vector predicates with the functions *vec* and *rel* and property (7) to test and column-wise compute subsets of sets of relations.

Meanwhile, we have applied the method to other problems, too, e.g., in the context of Petri nets or evolutionary algorithms (see [11]). In doing so, also the limits of the method became apparent, for example, the non-applicability of property (7) in the case of a set of relations which have to be transitive. Presently, we work on the overcoming of these restrictions. Besides the "direct" development of relation-algebraic expressions that specify such sets $\mathfrak{S}$ without using the property, we also concentrate on the development of specifications for good approximations of $\mathfrak{S}$ using (7). From the latter, we hope that $\mathfrak{S}$ can be obtained by inspecting only a moderate number of relations.

Acknowledgement

We thank F. Meyer from the administration of the University of Kiel for his support and a lot of stimulating discussions.

Appendix A

In this appendix, we prove Theorem 4.2(b). Since this part of the theorem consists of an equivalence, we reformulate it as two separate theorems and prove these one after another. Here is the first one. Note, that because of Theorem 4.2(a) a right composition SP of a solution S with a block-preserving permutation relation P yields again a solution.

Theorem 1. *If the relation S is a solution of the university timetabling problem and P is a block-preserving permutation relation, then S and SP are isomorphic solutions.*

Proof. Since P is a block-preserving permutation relation if and only if it is a relational isomorphism wrt, the input relation B of the university timetabling problem in the sense of [15], we have $B = PBP^\top$. This yields

$$\begin{aligned} co(SP) &= SP(B \cap \bar{\mathsf{I}})(SP)^\top \\ &= SP(B \cap \bar{\mathsf{I}})P^\top S^\top \\ &= S(PBP^\top \cap P\bar{\mathsf{I}}P^\top)S^\top && P \text{ univalent} \\ &= S(PBP^\top \cap \overline{P\mathsf{I}P^\top})S^\top && P \text{ mapping} \\ &= S(PBP^\top \cap \bar{\mathsf{I}})S^\top && \text{since } PP^\top = \mathsf{I} \\ &= S(B \cap \bar{\mathsf{I}})S^\top && \text{see above} \\ &= co(S). \end{aligned}$$

Using $B = PBP^\top$ again, we get furthermore that

$$\begin{aligned} reco(SP) &= SP\,\overline{B}\,(SP)^\top \\ &= SP\,\overline{B}\,P^\top S^\top \\ &= S\,\overline{PBP^\top}\,S^\top && P \text{ mapping} \\ &= S\,\overline{B}\,S^\top && \text{see above} \\ &= reco(S). \end{aligned}$$

Both calculations show that S and SP are isomorphic solutions. □

The input relation B of the university timetabling problem is reflexive. A consequence is $\overline{(B \cap \bar{\mathsf{I}}) \cup \overline{B}} = \overline{\bar{\mathsf{I}} \cup \overline{B}} = \mathsf{I}$, which in turn yields for all solutions S of the university timetabling problem

$$\begin{aligned} SS^\top &= S\,\overline{(B \cap \bar{\mathsf{I}}) \cup \overline{B}}\,S^\top \\ &= \overline{S((B \cap \bar{\mathsf{I}}) \cup \overline{B})S^\top} && S \text{ univalent} \\ &= \overline{S(B \cap \bar{\mathsf{I}})S^\top \cup S\overline{B}S^\top} \\ &= \overline{co(S) \cup reco(S)}. \end{aligned}$$

After these preparations, we are able to show the remaining direction of Theorem 4.2(b) by element-wise reasoning. Doing so, we assume that the containment of the groups in the blocks is given by a mapping (in the usual mathematical sense) $bc : \mathfrak{G} \to \mathfrak{B}$ such that $bc(g) \in \mathfrak{B}$ is the unique block the group $g \in \mathfrak{B}$ belongs to.

Theorem 2. *If the relations S and S' are isomorphic solutions of the university timetabling problem, then there exists a block-preserving permutation relation P such that $S' = SP$.*

Proof. The existence of the relation P is shown by a series of single steps. First, we consider the subset $\mathfrak{G}_S$ of $\mathfrak{G}$ that is represented by the vector $S^\top\mathsf{L}$, i.e. the set

$$\mathfrak{G}_S = \{g \in \mathfrak{G} \mid (S^\top\mathsf{L})_g\} = \{g \in \mathfrak{G} \mid \exists s \in \mathfrak{S} : S_{s,g}\}$$

and also the subset $\mathfrak{G}_{S'}$ of $\mathfrak{G}$ that is represented by the vector $S'^\top\mathsf{L}$. Then, we have that for all $g \in \mathfrak{G}_S$ there exists $g' \in \mathfrak{G}_{S'}$ such that for all $s \in \mathfrak{S}$ the relationships $S_{s,g}$ and $S'_{s,g'}$ are equivalent.

Proof. Assume $g \in \mathfrak{G}_S$. Then there exists $\tilde{s} \in \mathfrak{S}$ such that $S_{\tilde{s},g}$. For S' is a mapping, there exists exactly one $g' \in \mathfrak{G}$ with $S'_{\tilde{s},g'}$. Suppose now an arbitrary $s \in \mathfrak{S}$. To verify the equivalence of $S_{s,g}$ and $S'_{s,g'}$, we assume $S_{s,g}$. Then this yields $(SS^\top)_{\tilde{s},s}$. Since S and S' are isomorphic, we obtain from the preparatory calculation that

$$SS^\top = \overline{co(S) \cup reco(S)} = \overline{co(S') \cup reco(S')} = S'S'^\top$$

and it follows $(S'S'^\top)_{\tilde{s},s}$. Finally, the univalence of S' yields $S'_{s,g'}$. The other implication of the equivalence can be shown analogously.

Due to the fact that for each $g \in \mathfrak{G}_S$ there exists exactly one $g' \in \mathfrak{G}_{S'}$ with the property stated above, we can define a bijective mapping (again in the usual mathematical sense) as follows:

$$\psi : \mathfrak{G}_S \to \mathfrak{G}_{S'}, \quad \psi(g) = g'$$

The mapping ψ preserves block containment, i.e., for all $g, g' \in \mathfrak{G}_S$ we have that from $bc(g) = bc(g')$ it follows $bc(\psi(g)) = bc(\psi(g'))$.

Proof. For $g = g'$ the statement obviously holds. Now, let g and g' be different groups and contained in the same block. Then it holds $(B \cap \bar{\mathrm{I}})_{g,g'}$. For the pair $g, g' \in \mathfrak{G}_S$ there exists a pair $s, s' \in \mathfrak{S}$ with $S_{s,g}$ and $S_{s',g'}$ and it follows $co(S)_{s,s'}$ from $(B \cap \bar{\mathrm{I}})_{g,g'}$. Because S and S' are isomorphic, we have also $co(S')_{s,s'}$ and from $S'_{s,\psi(g)}$ and $S'_{s',\psi(g')}$ we get the desired result.

As an immediate consequence of the just proven property, we can define another mapping $\alpha : bc(\mathfrak{G}_S) \to bc(\mathfrak{G}_{S'})$ by $\alpha(bc(g)) = bc(\psi(g))$. The mapping α is bijektive and, therefore, we obtain a bijective mapping $\beta : \mathfrak{B} \leftrightarrow \mathfrak{B}$ with $\beta_{|bc(\mathfrak{G}_S)} = \alpha$. Let now $b \in \mathfrak{B}$ be any block. Then it holds

$$|\mathfrak{G}_S \cap bc^{-1}(b)| = |\mathfrak{G}_{S'} \cap bc^{-1}(\beta(b))|.$$

Proof. Assume $g \in \mathfrak{G}_S \cap bc^{-1}(b)$. Then, we obtain $\psi(g) \in \mathfrak{G}_{S'}$, and from the property $bc(g) = b$ it follows $bc(\psi(g)) = \alpha(bc(g)) = \alpha(b) = \beta(b)$ and, therefore, $\psi(g) \in bc^{-1}(\beta(b))$. We can conclude that $\psi(g) \in \mathfrak{G}_{S'} \cap bc^{-1}(\beta(b))$ and, therefore, get $\psi(\mathfrak{G}_S \cap bc^{-1}(b)) \subseteq \mathfrak{G}_{S'} \cap bc^{-1}(\beta(b))$. Combining this with the bijectivity of ψ, we arrive at

$$|\mathfrak{G}_S \cap bc^{-1}(b)| = |\psi(\mathfrak{G}_S \cap bc^{-1}(b))| \le |\mathfrak{G}_{S'} \cap bc^{-1}(\beta(b))|.$$

By exchanging S and S' and using the inverse mappings of ψ and α, we obtain the reverse estimation, too, that completes the proof.

Another immediately consequence is for all $b \in \mathfrak{B}$ the equality

$$|(\mathfrak{G} \setminus \mathfrak{G}_S) \cap bc^{-1}(b)| = |(\mathfrak{G} \setminus \mathfrak{G}_{S'}) \cap bc^{-1}(\beta(b))|.$$

So, for each $b \in \mathfrak{B}$ there exists a bijective mapping

$$\psi^{(b)} : (\mathfrak{G} \setminus \mathfrak{G}_S) \cap bc^{-1}(b) \to (\mathfrak{G} \setminus \mathfrak{G}_{S'}) \cap bc^{-1}(\beta(b))$$

and this allows to define the permutation relation $P : \mathfrak{G} \leftrightarrow \mathfrak{G}$ we are looking for as follows: For all $g, g' \in \mathfrak{G}_S$ we define $P_{g,g'}$ if and only if $\psi(g) = g'$ and for all $b \in \mathfrak{B}$ and $g, g' \in (\mathfrak{G} \setminus \mathfrak{G}_S) \cap bc^{-1}(b)$ we define $P_{g,g'}$ if and only if $\psi^{(b)}(g) = g'$. The permutation relation P is block-preserving by construction. It remains to verify $S' = SP$.

Proof. Assume $s \in \mathfrak{S}$ and $g' \in \mathfrak{G}$ such that $(SP)_{f,g'}$. Then there exists $g \in \mathfrak{G}$ with $S_{s,g}$ and $P_{g,g'}$. From $g \in \mathfrak{G}_S$ it follows $\psi(g) = g'$ and, therefore, $S'_{s,g'}$ due to the definition of ψ. Hence, we have $SP \subseteq S'$. Now, from S' and SP being mappings it even follows $SP = S'$ (cf. [15]), and we are done. $\square$

References

[1] R. Behnke, R. Berghammer, E. Meyer, P. Schneider, RelView – a system for calculation with relations and relational programming, in: E. Astesiano (Ed.), Proceedings of the 1st Conference on Fundamental Approaches to Software Engineering, LNCS, vol. 1382, Springer, 1998, pp. 318–321.

[2] R. Berghammer, B. Leoniuk, U. Milanese, Implementation of relation algebra using binary decision diagrams, in: H. de Swart (Ed.), Proceedings of the 6th International Workshop on Relational Methods in Computer Science, LNCS, vol. 2561, Springer, 2002, pp. 241–257.

[3] R. Berghammer, F. Neumann, RelView – an OBDD-based computer algebra system for relations, in: V.G. Gansha, E.W. Mayr, E. Vorozhtsov (Eds.), Proceedings of the 8th International Workshop on Computer Algebra in Scientific Computing, LNCS, vol. 3718, Springer, 2005, pp. 40–51.

[4] R. Berghammer, Relation-algebraic computation of fixed points with applications, J. Logic Algebraic Programming 66 (2006) 112–126.

[5] R. Berghammer, Ordnungen, Verbände und Relationen mit Anwendungen, Vieweg-Teubner, 2008.

[6] C. Brink, W. Kahl, G. Schmidt (Eds.), Relational Methods in Computer Science. Advances in Computing Science, Springer, 1997.

[7] H. de Swart, E. Orlowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments, LNCS, vol. 2929, Springer, 2003.

[8] H. de Swart, E. Orlowska, G. Schmidt, M. Roubens (Eds.), Theory and Applications of Relational Structures as Knowledge Instruments II, LNAI, vol. 4342, Springer, 2006.

[9] J.L. Gross, J. Yellen (Eds.), Handbook of Graph Theory, CRC Press, 2003.

[10] B. Kehden, Evaluating sets of search points using relational algebra, in: R.A. Schmidt (Ed.), Proceedings of the 9th International Workshop on Relational Methods in Computer Science, LNCS, vol. 4136, Springer, 2006, pp. 266–280.

[11] B. Kehden, Vectors and Vector Predicates and their Use in the Development of Relational Algorithms, Dissertation, Christian-Albrechts-Universität Kiel, 2008 (in German).

[12] A. Schaerf, A survey of automated timetabling, Artif. Intell. Rev. 13 (1999) 87–127.

[13] G. Schmidt, T. Ströhlein, Some aspects in the construction of timetables, in: J.L. Rosenfeld (Ed.), Proceedings of the IFIP Congress 1974, North Holland, 1974, pp. 516–520.

[14] G. Schmidt, T. Ströhlein, A Boolean matrix iteration in timetable construction, Linear Algebra Appl. 15 (1976) 27–51.

[15] G. Schmidt, T. Ströhlein, Relationen und Graphem, Springer, 1989. (Available also in English: Relations and Graphs. Discrete Mathematics for Computer Scientists, EATCS Monographs on Theoretical Computer Science, Springer, 1993).

[16] A. Tarski, On the calculus of relations, J. Symbolic Logic 6 (1941) 73–89.