M. Abraham
D. Gabbay
U. Schild

Analysis of the Talmudic Argumentum A Fortiori Inference Rule (Kal Vachomer) using Matrix Abduction

Abstract. We motivate and introduce a new method of abduction, Matrix Abduction, and apply it to modelling the use of non-deductive inferences in the Talmud such as Analogy and the rule of Argumentum A Fortiori. Given a matrix $\mathbb{A}$ with entries in $\{0, 1\}$, we allow for one or more blank squares in the matrix, say $a_{i,j} =?$. The method allows us to decide whether to declare $a_{i,j} = 0$ or $a_{i,j} = 1$ or $a_{i,j} =?$ undecided. This algorithmic method is then applied to modelling several legal and practical reasoning situations including the Talmudic rule of Kal-Vachomer. We add an Appendix showing that this new rule of Matrix Abduction, arising from the Talmud, can also be applied to the analysis of paradoxes in voting and judgement aggregation. In fact we have here a general method for executing non-deductive inferences.

Keywords: Matrix Abduction, Talmudic logic, Argumentum A Fortiori, Qal-Vachomer, argumentation.

Contents

Special Issue: New Ideas in Applied Logic
Edited by **Dov M. Gabbay and Jacek Malinowski**

Studia Logica (2009) 92: 281–364
DOI: 10.1007/s11225-009-9202-5

1. Introduction and Motivation

This section explains and motivates the intuitive use of the new method of Matrix Abduction to analyse the non-deductive rules of Analogy and Argumentum A Fortiori. This rule is a form of Induction rule when used in an Artificial Intelligence context and is a recognised Jurisprudence rule in Jewish, Islamic and Indian legal reasoning. In the Jewish Talmud it is known as the Binyan Abh and the Kal-Vachomer rules. In Islamic jurisprudence it is known as Qiyas (analogy) and in Sanskrit logic (Nyaya) it is known as Kaimatya Nyaya (or Kaimutika Nyaya, the *even more so*) rule.

1.1. Matrix Abduction in AI

Let us begin by trying to buy, over the Internet, two items:

1. An LCD computer screen
2. A digital camera

We start with the LCD screen. We want something good, within a price range we can afford and we would especially like it to have stereophonic speakers. So the usual thing to do in such cases is to go to a price comparison website. In our case.[1] We went to `www.wallashops.co.il` and got comparison tables for four candidates.

Screen 1. Xerox XM7 24A

Screen 2. Viewsonic FHD VX 2640w

Screen 3. Nec 2470 WVX

Screen 4. Nec 24 WMCX

The specifications of interest we got are shown in Figure 1 below.

It seems that for screen 3 there is no information about the stereophonic feature. It was not possible to get the information from other sites. Can we abduce the information from the table itself? How do we do that?

Let us check another example, where a similar problem arises. We look for cameras at the same site.

Camera 1. Canon A590 8MP + 4GB

Camera 2. Olympus FE20 (thin) 8MP + 4GB

Camera 3. Olympus FE60 + 2GB

Camera 4. Olympus 8MP in Hebrew + 1GB.

	P price over £450	C self collection	I screen bigger than 24inch	R reaction time below 4ms	D dot size less than 0.275	S stereo-phonic
Screen 1	0	1	0	1	0	1
Screen 2	0	0	1	1	0	1
Screen 3	0	0	0	0	1	?
Screen 4	1	1	0	0	1	1

1 = yes; 0 = no; ? = no data given

Figure 1.

	P price over £100	M over 12 monthly payments	D quick delivery	B more than one battery	W weighs more than 150g	F flash has more than 3 states	E can edit image afterwards
Camera 1	1	1	0	1	1	0	?
Camera 2	0	0	0	0	0	1	1
Camera 3	0	0	1	0	0	1	1
Camera 4	1	0	1	0	0	0	1

Figure 2.

Figure 2 gives the specifications for comparison.

Again, there is no information whether Camera 1 can edit an image taken into the camera memory. Our question is, can we assume that this type of camera, as compared with the others, will have this feature? Can we use the matrix to get the answer? See Example 3.25 for a solution.

We can now formulate the general problem:

Definition 1.1 (Matrix abduction problem). Let $\mathbb{A} = [a_{i,j}]$ be a $0-1$ matrix, where $a_{i,j} \in \{0, 1, ?\}$ $i = 1, \ldots, m$ (m rows) $j = 1, \ldots, n$ (n columns) such that the following holds:

a. $m \leq n$ (there are more columns than rows[2]

b. exactly one a_{i_0,j_0} is undecided all the others are in $\{0, 1\}$.

The abduction problem is to devise some algorithm which can decide whether $a_{i_0,j_0} =?$ should be 1 or $a_{i_0,j_0} =?$ should be 0 or $a_{i_0,j_0} =?$ must remain undecided.

We cannot solve this problem without further assumptions on the meaning of the entries. Put in different words, if we give an algorithm $\mathcal{A}$ to

[1]This is a real example we are describing, of what we did on 1.2.09.

[2]This condition does not matter in the formal abstract case, since we can rotate the matrix. However, in some applications the rows and columns may have special meaning. The formal machinery works even if $n < m$.

be applied to $\mathbb{A}$, we need to specify its range of applicability. To specify whether $\mathcal{A}$ can be meaningfully applied to $\mathbb{A}$, we need to know how $\mathbb{A}$ was constructed. In other words, we need some assumptions about the meaning of the rows and columns of $\mathbb{A}$.

The examples above suggest that we can look at the roles of $\mathbb{A}$ as representative agents or causes, which can generate various features. The columns of the matrix represent the features. So objects like cameras or LCD screens can "generate" the properties listed in the columns. There are other examples, for instance, hurricanes can generate a lot of damage through various features. If we go to the web, we can find a list of names for hurricanes and a list of the main kinds of features they generated. We can construct, for example, Figure 3.

	rip tide	winds	storm surge	flooding	tornado
Katrina					
Andrew					
Ivan					
Hugo					
Camille					
$\vdots$					

Figure 3.

The $a_{i,j}$ slots usually contain numerical data or even qualitative data. For example, the wind column may contain the maximum speed in miles per hour of each hurricane.

To turn the data into $0-1$ data we need to decide on a cut-off point. Say for winds we choose 150 miles per hour. We have two choices for the wind column. Do we take 1 to mean over 150 miles per hour or do we take it to be 1 = under 150 miles per hour? The reader might think it is a matter of notation but it is not! We need to assume that all the column features pull in the same direction. In the hurricane case the direction we can take is the capacity for damage. In the LCD screen and camera case it is performance. So to put 1 as opposed to 0 in a box indicates more strength to the feature in the general agreed shared direction. So the representation of the columns must be compatible with the chosen direction. So if 1 in the winds means over 150 miles (in direction of increased damage), then 1 in the tornadoes column must go in the same direction. To give an example of a matrix where there is no direction to the columns, take a simple graph, see Figure 4.

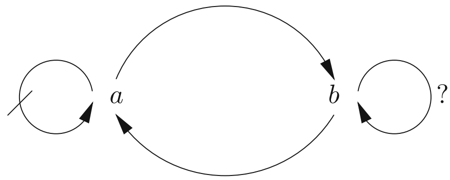


Figure 4.

This is a binary relation R

$$(a,b) \in R, (b,a) \in R, (a,a) \notin R.$$

We ask is $(b,b) \in R$? Form the characteristic matrix in Figure 5

	a	b
a	0	1
b	1	?

Figure 5.

There is no direction or meaning here. We might apply our algorithm formally and get an answer but it means nothing.

Let us give another example. Imagine a society of agents and various context in which the agents might wish to apply actions. Say action **a** might be to shoot to kill and the context B might be a burglar in the middle of the night. The matrix $\mathbb{A}$ might give $0-1$ values to indicate the accepted norms.

So we may get Figure 6

	A	B	C	D
a	0	1	1	0
b	1	?	0	0
c	1	1	0	1

Figure 6.

$\mathbf{a}, \mathbf{b}, \mathbf{c}$ are actions and A, B, C, D are situations.

We know that action **b** for example is allowed in situation A but we don't know about situation B.

The reader may ask how can such a problem arise? If actions are described by pre-conditions and post-conditions and situations (states) are also described properly then we can check whether the preconditions hold and whether there are any restrictions on the execution of the action. The problem is that the above presentation is already a formal model.

If we construct a table and use common sense, there may be clear answers in some places and question marks in others. Figure 7 is such a case:

	burglar unarmed	burglar armed	burglar could be armed	burglar could be armed but several murders same week
shoot the burglar	0	1	?	1
beat up the burglar	?	1	1	1

Figure 7.

This example also has a "direction". We allow severe actions for severe situations. So we may decide on the basis of the matrix that if the burglar may be armed then better shoot him and maybe then decide that if he is definitely not armed then beat him up, sensing the general "severe" spirit of the case.

Another example could be from monadic predicate logic over a finite domain. The domain can be $\{d_1, \ldots, d_n\}$ and the monadic predicates $A_1(x), \ldots, A_m(x)$. Assume we know all values of $A_i(d_j) = e_{i,j}$ except one, say $A_k(d_r) =?$. We get the matrix of Figure 8

	d_1	d_2	$\ldots$	d_r	$\ldots$	d_n
A_1	$e_{1,1}$	$e_{1,2}$				
A_1	$e_{2,1}$	$e_{2,2}$				
$\vdots$						
A_k	$e_{k,1}$	$e_{k,2}$		?		
$\vdots$						
A_m	$e_{m,1}$	$e_{m,2}$				

Figure 8.

Our method allows us, under certain assumptions, to get a value for $A_k(d_r) = e_{k,r}$. More on this example in Appendix D.

The above discussion gives us an idea of when a matrix $\mathbb{A}$ is within the range of applicability of our matrix abduction algorithm. However the explanation is not formal but in terms of examples showing how the matrix is constructed we have not given a formal definition which says when an arbitrary $0-1$ matrix is within the range of applicability.

We shall do this in Section 3 where we develop some matrix machinery. Definition 3.7 describes when a matrix has a 'direction'. Meanwhile consider the two matrices in Figure 9. The first is not within range, it has no direction, the second is within range, as we shall see later.

(1)

	A	B	C
a	1	0	0
b	0	1	0
c	0	0	1

(2)

	A	B	C
a	1	1	1
b	0	1	1
c	0	0	1

Figure 9.

1.2. The Talmudic Kal-Vachomer[3]

Here we give a small example. An extensive model will be given later in the paper.

A bull can do damage in two ways.[4] It can trample something with its feet or it can use its horns. Also the location of the arena of the damage can either be in a public place or a private place (e.g. a public road or a private garden). The amount of compensation paid by the owner of the bull depends on these features.

Figure 10 describes the situation.[5] The entries indicate proportion of the damage to be paid, as indicated in the Bible.

---

[3] Also written as Qal-Vahomer, or Qal-Vachomer.

[4] Actually a bull can damage in three ways. He can eat (Tooth), trample (Foot) and gore (Horn). The Horn is intentional damage. The Tooth and Foot are not intentional, but the Tooth gives benefit to the perpertrator and the Foot gives no benefit.

[5] If we insist on $\{0,1\}$ values in the matrix we can read the entries as taking either the value 0 or taking the value of $\frac{1}{2}$ or 1 in which case the (Horn, public place) square will be 1. Subsequent considerations for $x=?$ in Section 1.3 always use $x \geq \frac{1}{2}$ or $x=0$ as options anyway.

	public place	private place
Foot action	0	1
Horn action	$\frac{1}{2}$	$x = ?$

Figure 10.

The Talmudic law specifies that foot damage by a bull at a public place needs to pay 0 compensation. Horn damage at a public place pays $\frac{1}{2}$ the cost of damage as compensation.

In a private area foot damage must be paid in full. What can we say now about payment for horn action in a private place? This is not specified explicitly in the Biblical written law and the Talmud is trying to abduce it from the above matrix using Kal-Vachomer.

The next section shows how the Talmud does it.

1.3. Preview of the model

We use the bull example of Section 1.2 to show how the model works.

First the intuitive argument:
We see from the public arena, that horn damage is considered more serious than foot damage. You need to walk but certainly you don't need to use your horn in a public road! If this is the case, then if in a private place foot damage has to pay in full, then certainly horn damage has to pay in full.

We can also look at Figure 10 from the row point of view. We see from row 1 that damage in private area is considered more seriously than damage in public area. You are allowed routinely to walk and move in public areas but not in a private area. So if horn action in public area pays $\frac{1}{2}$ then certainly it has to pay at least half if done in a private area!

We now give you a glimpse of the maths of the model. Consider the matrix of Figure 10 and consider columns as vectors and consider two cases:

Case 1 we put for ? the value $x \geq \frac{1}{2}$.
Case 2 we put for ? the value $x = 0$.

For $x \geq \frac{1}{2}$ we have

$$\begin{array}{ccc} \text{Public} & & \text{Private} \\ \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix} & \leq & \begin{pmatrix} 1 \\ x \end{pmatrix} \end{array}$$

but for $x = 0$ we get that the two columns are not comparable.

We get two types of orderings, described in Figures 11 and 12. a, b are two abstract points of ordering which in our case can represent the public column and the private column. The two abstract orderings we get are

$$a \quad \to \quad b$$

Figure 11.

$$\bullet\ b \qquad\qquad\qquad \bullet\ a$$

Figure 12.

So one is a linear ordering $a \leq b$ and the other is no ordering of a, b. We ask which one is "nicer". The intuitive answer is that Figure 11 is nicer. So $x \geq \frac{1}{2}$ is our answer to the question in this case.

Comparing rows we get for $x = 1$:

$$\text{Horn } (\tfrac{1}{2}, 1) \geq \text{Foot}(0, 1),$$

i.e. the abstract ordering of Figure 11 again, where now a, b represent foot and horn rows.

For $x \leq \frac{1}{2}$ we get two incomparable rows, i.e. Figure 12 again. So again, if Figure 11 is considered "nicer", we must take $x = 1$ as our answer.

The actual case is decided in Jewish law as $x = \frac{1}{2}$.

1.4. Qiyas and Kaimutika Nyaya

We saw that the Talmudic rule of Kal-Vachomer is used in Jewish Jurisprudence to derive further conclusions and laws from the explicit existing laws in the Bible. A similar rule in Islamic Jurisprudence is in the Qiyas, see [19] and [17].

Literally Qiyas means measuring or ascertaining the length, weight or quality of something. It is used to extend a Shariah ruling from an original case to a new case. This is done by identifying a common cause between the original case and the new case. See Examples 1.2 and 1.3.

Example 1.2 (Example of Qiyas (from Wikipedia)). For example, Qiyas is applied to the injunction against drinking wine `/wiki/Wine` to create an injunction against cocaine `/wiki/Cocaine` use.

1. Identification of a clear, known thing or action that might bear a resemblance to the modern situation, such as the wine drinking.
2. Identification of the ruling on the known thing. Wine drinking is prohibited.
3. Identification of the reason behind the known ruling . For example, wine drinking is prohibited because it intoxicates. Intoxication is bad because it removes Muslims `/wiki/Muslim` from mindfulness of God.
4. The reason behind the known ruling is applied to the unknown thing. For instance cocaine use intoxicates the user, removing the user from mindfulness of God. It is therefore prohibited.

Example 1.3. This example is from `www.islamtoday.com`.

What is the ruling on giving one's parents a good smack?

We will not find any text in our scriptures that directly addresses this question. However, we are in no doubt that it is absolutely prohibited and sinful to do so.

We find in the Qur'ân that it is sinful to even mutter "ugh" or "uff" to our parents in exasperation when they ask us to do something for them.

Allah says: "And your Lord has commanded that you shall not worship any but Him, and that you show kindness to your parents. If either or both of them reach old age with you, say not to them so much as "ugh" nor chide them, but speak to them a generous word." [*Sûrah al-Isrâ*: 23]

We are prohibited to say "ugh" to our parents, because it is abusive behaviour. At the very least, it hurts their feelings. We can have no doubt that shoving them or smacking them is even more abusive and hurtful. Since the reason for prohibition is even more evident here, we can be certain that smacking our parents is unlawful and very sinful.

A similar rule exists in Indian Logic. We quote an example:

Example 1.4. **(Kaimutika Nyaya (from sadagopan.org)** `http://www.biblio.org/sadagopan/ahobilavalli/sus_v2p2.pdf`**)**

It has been said also in SANdilya-smriti: "There may be doubts concerning the redemption of those who serve AchArya, but there is absolutely no doubt about the redemption of those who delight in the service of His devotees" (1-95). So, in the case of those who depend solely on the AchArya, there is no doubt at all concerning the fruition of prapatti, by the principle of "kaimutika nyAya".

(Will not the Lord, who saves those who take refuge in His devotees, save those who take refuge in their AchAryas? Will not a benefit, which is got by one who is not qualified, be obtained by one who is qualified?).

It is thus established that sarveSvara, the Lord of all, will not grant us the supreme goal of existence, unless prapatti is performed in any of these two forms, and by some one or other. Thus the Lord has done another favour by revealing these important messages inbuilt in these mantras, said SwAmi Desikan, in this sub-section

It is now time to define the mathematical model.

2. Motivating the matrix model

Before we give the algorithm, let us say how it is to be used.

The algorithm works as follows. We are given a matrix $\mathbb{A}$ with one place with $x = ?$ and all the rest are entries from $\{0, 1\}$. We need to decide which is better $x = 0$ or $x = 1$ or declare the case as formally undecided.

Let $\mathbb{A}^1$ be the matrix with $x = 1$ and $\mathbb{A}^0$ be the matrix with $x = 0$.

Step 1
Let Π_1 be the partial order of the columns of $\mathbb{A}^1$, taken as vectors and compared coordinate wise. Let Π_0 be the same for $\mathbb{A}^0$.

Step 2
Decide, if you can, which is "nicer". (Formal definitions will be given later.)

Step 3
If Π_1 is definitely nicer than Π_0 then say $x = 1$ is the output. If Π_0 is definitely nicer than Π_1 then let output be $x = 0$.

If neither can be shown to be nicer then say that x is undecided.

Thus we need an algorithm on two partial orders X and Y to say either "X is better than Y" or "Y is better than X" or "X, Y are not comparable".

This algorithm must be compatible with the meaning of the rows and columns of the matrix as discussed in Section 1.1 and may use the matrix for help.

The next section will give precise mathematical definitions but before than we need to give some methodological remarks.

The Kal-Vachomer rule (and the algorithms supporting it) are nonmonotonic rules of induction. This means they are not absolute deductive rules but defeasible common sense rules. So we may use these rules to obtain a conclusion A, but further information and further arguments may force us to doubt A or even come to accept $\sim A$.

Let us compare this rule with ordinary Abduction.

Imagine we have hard facts, accepted statements of the form $\Delta = \{A_1, \ldots, A_m\}$, we are looking for further information. We consider Δ and using common sense, experience, our knowledge of the way the world works, our creative imagination, our religion and whatever else we bring to bear on the case, we put forward that H should be added to Δ. The decision to add H is defeasible. We may find out later, through more facts, etc., that H was the wrong addition.

On the other hand, compare this with a proof in school geometry. If $A_1, \ldots, A_m$ are assumptions about geometric figure and we *prove* H, then H follows absolutely! It is not defeasible no matter how much more information we get.

Matrix abduction is a defeasible rule. We looked for the missing information about the camera and could not get it. We may use matrix abduction to conjecture that the camera did have a stereophonic feature and decided to order it from the dealer. It is quite possible that when we get the camera we find out that it does not have this feature. This does not mean our matrix abduction rule was wrongly applied. The rule was correctly applied but was defeated by further data.

Another typical case of abduction can be described as follows. Given a theory Δ, say $\Delta = \{A, A \wedge B \to C, X\}$ and a result G, say that $G = C$ such that we know that $\Delta \vdash G$ must hold. However without knowing what X is, G does not follow from Δ. We therefore want to *abduce* a hypothesis H such that

1. $H \nvdash G$
2. $\Delta + H \vdash G$.

There may be several such candidates but we decide that a certain algorithm is the one we use. Once we decide on that, we can calculate H and let $X = H$. In the above case we can use a goal directed algorithm for $G = C$ which goes as follows:

1. What can give us C? We can get C from the data $A \wedge B \to C$ if we have two items A and B.
2. We do have A, but B is not available, however we have X which we don't know what it is.
3. So abduce $X = B$.

Thus $H = B$

Note that the weaker assumption $X = (A \to B)$ would also do the job but this is not what our algorithm does.

So for the case of the matrix $\mathbb{A}$ the inductive/abductive step is to use our specific algorithm. Once we decide on that, the answer becomes mathematically determined. However the whole process gives us defeasible conclusions, not absolute conclusions.

3. Superiority relation on partial orders

This section develops the mathematical machinery for our Matrix Abduction algorithm.

DEFINITION 3.1 (Graphs).

1. A partial order is a set S with a binary relation $<$ which is transitive and irreflexive. We write $\tau = (S, <)$ for a partial order. We write $\leq$ for the reflexive closure of $<$.
2. Let $x \prec y$ be the relation

$$x \prec y \text{ iff } x < y \wedge \sim \exists z (x < z < y)$$

   We represent graphically "$x \prec y$" by "$x \leftarrow y$". So for example in Figure 13, we have $x_1 \prec x_2, x_2 \prec x_3, y_1 \prec x_3$, etc., etc.
   $x \prec y$ means that x is an immediate predecessor of y (or equivalently y is an immediate successor of x).
   We have that $<$ is the transitive closure of $\prec$. From now on we look at $(S, \prec)$.
3. A path in $(S, \prec)$ of length n, is a chain of the form $x_1 \prec x_2 \prec \ldots \prec x_n$. A path is maximal if there is no longer path containing it.
4. Let $T \subseteq S$. Let T^*, the projection of T, be the set

$$T^* = \{y \mid y \leq t, t \in T\}.$$

5. Let T be a maximal path in $(S, \prec)$. T is said to be maximal thin path if there is no other maximal path T_1 such that T_1^* has less elements than T^*.
6. Figure 13 is a good graph to serve as an example for our concepts. There are two maximal paths.

$$T_1 : x_1 \prec x_2 \prec x_3 \prec z \prec x_4 \prec v$$

and

$$T_2 : x_1 \prec x_2 \prec x_3 \prec z \prec x_4 \prec x_5.$$

The path ending up in v is thinner, because $y_3 \in T_2^*$ but $y_3 \notin T_1^*$.

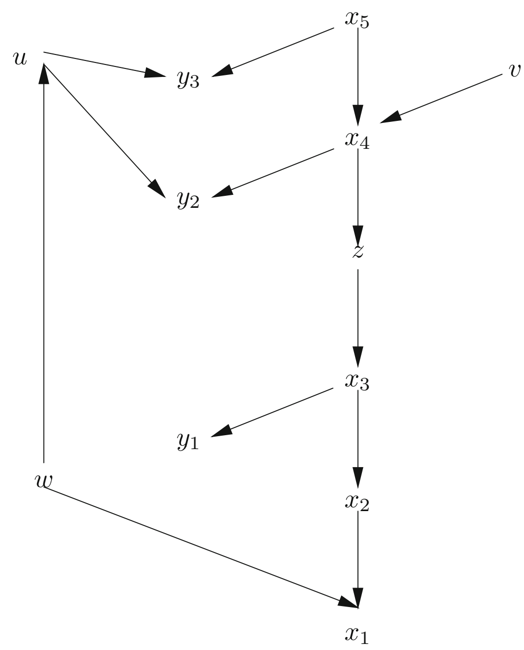


Figure 13.

7. We now divide $(S, \prec)$ into levels.
Level 1
All minimal points, i.e. all points x such that $\sim \exists y(y < x)$.
Level $n+1$
Let $P_r(y)$ be the set of all predecessors of y. Then y is of level $n+1$ if all predecessors of y are of level $\leq n$, with at least one such prececessor being of level n.
For example, in Figure 13, the following are the levels of the nodes:
Level 1 x_1, y_1, y_2, y_3
Level 2 x_2, u
Level 3 x_3, w
Level 4 z
Level 5 x_4
Level 6 x_5, v.

8. A point $z \in S$ is a critical point if the following holds:
   (a) z has at least two predecessors
   (b) there exists y such that $\sim (z < y)$ and all predecessors of z are less than y.

Definition 3.2 (Abduction matrices). An abduction matrix $\mathbb{A}$ has the form

$$\mathbb{A} = [a_{i,j}], 1 \leq i \leq m, 1 \leq j \leq n$$

where i runs over rows and j over columns we require

1. $m \leq n$
2. $a_{i,j} \in \{0, 1, ?\}$
3. exactly one $a_{i,j}$ has value ?

Definition 3.3.

1. A matrix $\mathbb{A}$ is definite if $a_{i,j} \neq ?$ for all i, j, that is we always have values in $\{0, 1\}$ for the entries.
2. Given a definite matrix $\mathbb{A}$, its columns are $0 - 1$ vectors of length m, i.e.

$$V_j = (a_{1,j}, a_{2,j}, \ldots, a_{m,j}).$$

Define an ordering on two vectors V, V', by comparing coordinates.

$$V \leq V' \text{ iff for all } i, v_i \leq v'_i,$$

where

$$V = (v_1, \ldots, v_m), V' = (v'_1, \ldots, v'_m).$$

We also indicate the ordering by writing

$$V \to V'.$$

Lemma 3.4 (Graph representation theorem). *Let $(S, <)$ be an abstract partial ordering based on a finite set $S = \{a_1, \ldots, a_n\}$. Then there exists a definite matrix with m column and m rows such that the column ordering is the same as $(S, \leq)$.*

Proof. Let $\mathbb{A} = [a_{i,j}], 1 \leq i, j \leq 1$ be the matrix defined by $a_{i,j} = 1$ iff $a_i \leq a_j$.

This is the characteristic matrix of the ordering. We shall see that we can identify that column V_i with the element a_i
we have

$$V_k \leq V_j$$

iff (by definition)

$$\text{for } i = 1, \ldots, n \text{ we have } a_{i,k} \leq a_{i,j}$$

iff for $i = 1, \ldots, m$

$$a_i \leq a_k \Rightarrow a_i \leq a_j$$

$$\text{iff } \forall x \in S(x \leq a_k \rightarrow x \leq a_j)$$

iff (since $\leq$ is reflexive and transitive

$$a_k \leq a_j. \qquad \blacksquare$$

Remark 3.5. The lemma is important because we can assume that the definition for “nicer” or superiority among ordering can use conditions and properties of the matrix generating them.

So from now on we can assume that every ordering $\tau = (S, \leq)$ comes from a matrix $\mathbb{M} = \mathbb{M}_\tau$, or $\tau = \tau(\mathbb{M})$.

We now want to get some intuition about when one ordering is superior to another. Our strategy is as follows

1. Look at some orderings and give some plausible mathematical definition of when one is superior to another. Such a definition must use topological and mathematical properties of the ordering and in no way have any connection with problems of abduction and reasoning.
2. It is inevitable that such a definition will be partial and incomplete and in many cases will have nothing to say.

We now run our definition on orderings derived from matrices arising from actual reasoning cases where we know what answers we should be getting. We use these cases to make our partial definitions more precise.

If the extra precision required turns out to be topologically meaningful, then we can say we got a good model, because of the intuitions of the reasoning do correspond to topological conditions on the ordering.

Example 3.6 (Examples of ordering). The following (Figure 14) are some examples of orderings. Note that for $b \prec a$ we also write

$$a \rightarrow b$$

or

$$\begin{array}{c} a \\ \downarrow \\ b \end{array}$$

when we present the ordering as a graph.

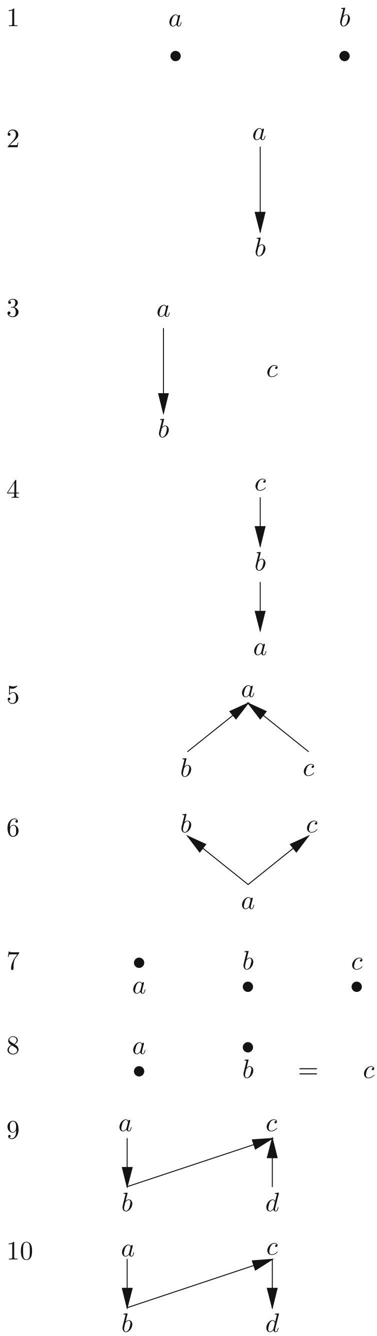


Figure 14.

We now give a partial definition of superiority. To expand the partial definition into a full definition we need to know the application area, and take it into consideration.

DEFINITION 3.7 (Multisets).

1. Let $\mathbb{L}$ be a set of labels $\mathbb{L} = \{\alpha_1, \alpha_2, \ldots\}$. Let $\mathbb{M}(\mathbb{L})$ be the family of all multisets based on $\mathbb{L}$. So these are subsets with copies from $\mathbb{L}$. For example $\{2\alpha, 3\beta\}$, this is a multiset with 2 copies of α and 3 copies of β.
2. Let $\mathbb{M}_1(\mathbb{L})$ be all multisets of the form $\{m\alpha, \beta_1, \ldots, \beta_{k-1}\}$, i.e. at most one element appears with more than one copy. So for example $\{2\alpha, 3\beta, \gamma\}$ is not in $\mathbb{M}_1(\mathbb{L})$.
   We call the number k in $\{m\alpha, \beta_1, \ldots, \beta_{k-1}\}$ the dimension of the element and the number m in its (multi-valued) index.
3. Let E be a finite subset of $\mathbb{M}_1(\mathbb{L})$. Define the dimension and index of E as the maximum of the respective dimensions and indices of its elements.

DEFINITION 3.8 (Multiset representation of ordering). Let $(S, <)$ be a partially ordered set and let $\mathbb{L}$ be a set of labels. A function $\mathbf{f}$ giving for each $x \in S$ a multiset $\mathbf{f}(x)$ be in $\mathbb{M}_1(\mathbb{L})$ is called an $(\mathbb{L}, \mathbf{f})$ realisation of $(S, <)$ iff the following holds:

$$(*) \qquad x \leq y \text{ iff } \mathbf{f}(x) \subseteq \mathbf{f}(y) \text{ where } \subseteq \text{ is a multiset inclusion}$$

The dimension and index the realisation are defined as those of $E = \{\mathbf{f}(x) | x \in S\}$.

LEMMA 3.9. *For every $(S, <)$ there exists an $\mathbb{L}$ and $\mathbf{f}$ such that $(\mathbb{L}, \mathbf{f})$ is a realisation of $(S, <)$.*

PROOF. Let $\mathbb{L} = S$ and let $\mathbf{f}(x) = \{y | y \leq x\}$. ■

DEFINITION 3.10 (Multiset representation of matrices). Let $\mathbb{A} = [a_{i,j}]$ be a definite abduction matrix. Let $V_1, \ldots, V_n$ be its columns and $U_1, \ldots, U_m$ be its rows. Let $\mathbb{L}$ be a set of labels.

1. A function $\mathbf{f}$ giving each column and each row X a multiset $\mathbf{f}(X) \in M_1(\mathbb{L})$ is considered a realisation of $\mathbb{A}$ iff the following holds

   $$(*) \qquad a_{i,j} = 1 \text{ iff } \mathbf{f}(U_i) \supset \mathbf{f}(V_j)$$

2. We say that the matrix $\mathbb{A}$ *has a direction* if it has a representation where the number of labels in $\mathbb{L}$ is strictly less than the number of columns in $\mathbb{A}$.

LEMMA 3.11. *Let $\mathbb{A}$ be a definite matrix. Let U_i, V_j be the rows and columns respectively. Let $\mathbb{L} = \{V_j\}$. Let $\mathbf{f}$ be defined as follows:*

$$\begin{aligned}\mathbf{f}(V_j) &= \{V_j\}\\ \mathbf{f}(U_i) &= \{V_j | a_{i,j} = 1\}.\end{aligned}$$

Then $\mathbf{f}$ is a representation for $\mathbb{A}$.

We have indeed

$$\mathbf{f}(U_i) \supseteq \{V_j\} \text{ iff } a_{i,j} = 1.$$

We now define the concepts we shall use to give a definition of when one ordering τ_1 is superior to another ordering τ_2.

DEFINITION 3.12 (Minimal realisation). Let $(S, <)$ be an ordering and let $(\mathbb{L}, \mathbf{f})$ be a realisation of it. The realisation is said to be *label-minimal* iff there is no other realisation $(\mathbb{L}', \mathbf{f}')$ with less labels, i.e. the dimension of $(\mathbb{L}, \mathbf{f})$ is minimal among all the realisations of $(S, <)$.

The proof of Lemma 3.9 presented a multiset realisation for $(S, <)$, using the same number of labels as the number of points in S. It is important for us to minimise the number of labels needed for the realisation, as we use this number as a simplicity indicator for the ordering. We shall therefore give a construction for obtaining realisations a with minimal number of points.

To explain to the reader the ideas and difficulties with this algorithm, we begin by executing it for the graph in Figure 13 and pointing step-by-step all key points. Examples 3.13, 3.14 and 3.15 do the job.

EXAMPLE 3.13 (Maximal chains). First note some important strategic points. Consider the graph in Figure 15.

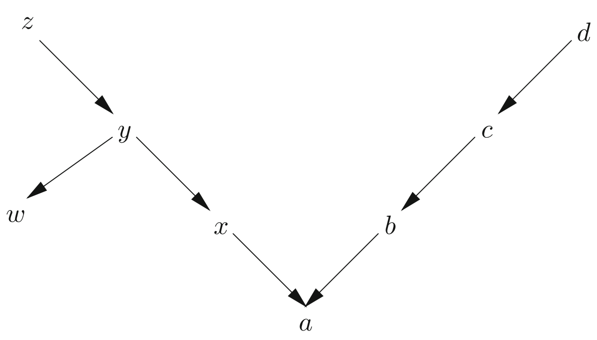


Figure 15.

We want to give it a realisation with a minimal use of labels. We are allowed to duplicate only one label, say α. So we can use $2\alpha, 3\alpha, \ldots$.

For this purpose it is good to identify a long chain and increase the number of copies of the α label along the chain. In Figure 15 we have two chains

$$a < b < c < d$$

and

$$a < x < y < z.$$

The second chain has $w < y$, this contributes a new label to y and so saves us from the need of duplicating α. It is therefore better to increase the α along the other chain.

Figures 16 and 17 show the two options

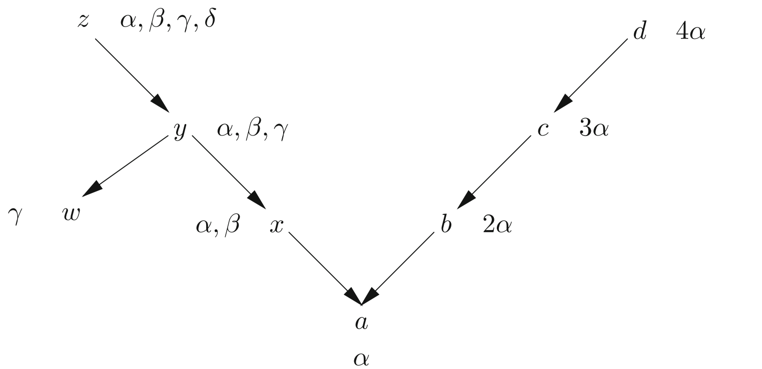


Figure 16.

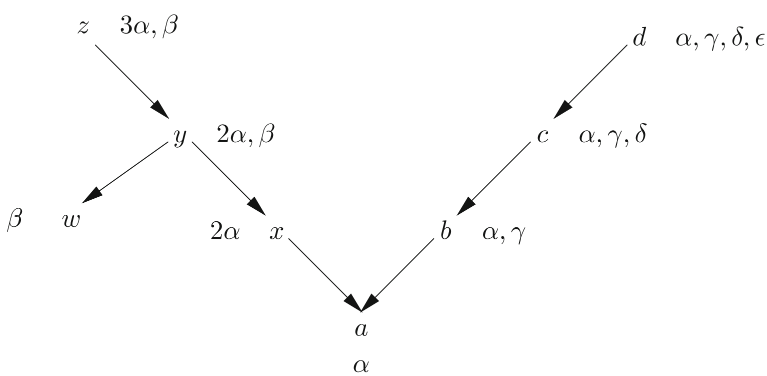


Figure 17.

Obviously Figure 17 requires more labels.

We can be clever and duplicate α along both chains. So in Figure 17 we can make $\delta = \alpha$ (i.e. increase α and have for c $2\alpha, \gamma$). This will not work because it makes $x < c$. Similalry we cannot make $\delta = \alpha$ in Figure 16 because this will make $b < z$. So our strategy is to choose a maximal chain with as little as possible points smaller than members of it.

This is what we called thin chain in Definition 3.1.

There is a trick we can use when the number of points in S is finite, say less than a fixed k. In our Figure 15 the number of points is less than 8.

So we progress with α along the main axis along the progression α, $(m+1)\alpha, (m+2)\alpha$, etc., where m is the number of points remaining in the chain and $m \leq k$.

Using this trick Figure 16 becomes Figure 18.

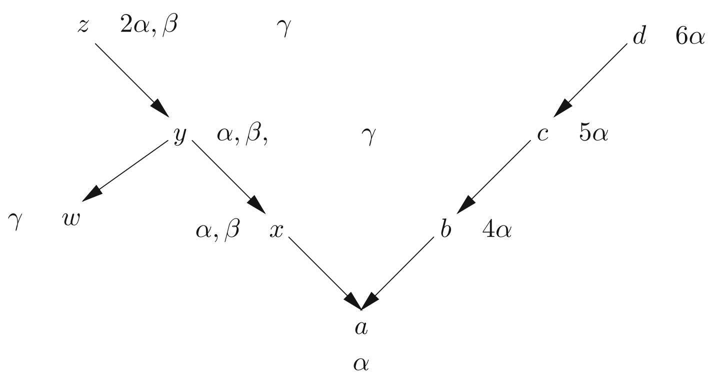


Figure 18.

Similarly Figure 19 can improve on Figure 17.

So the strategy is to choose a good maximal chain and increase α by at most by multiples of $k\alpha$ as necessary along the chain and increase α by 1 in all other directions.

The reason we advance possibly in multiples of $k\alpha$ along the main chain is because of the following possible situation in Figure 20

We would have had $e < x$ if we had not advanced in more than just one α along the main chain $a < c < e < y < z$.

Note that we don't really need to increase the numbers of α by a jump of up to $k\alpha$ at every point of the main chain. We need to do that only after split points such as c in Figure 20. Since at e there is no split y can get 10α and similarly z can get 11α.

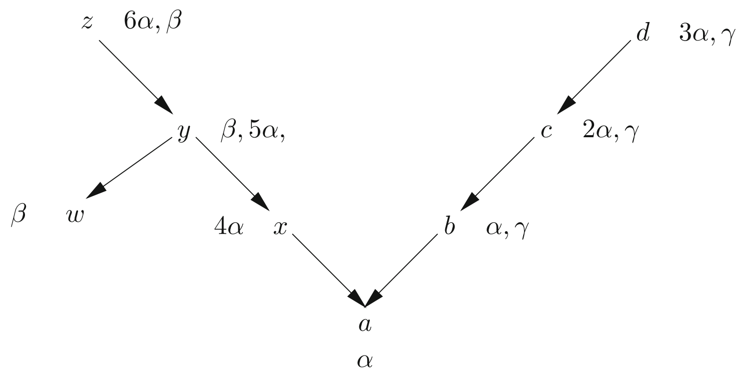


Figure 19.

In fact, we need to increase after a split at point s by at most the number $d(s) + 1$, where $d(s)$ is equal to the remaining points in the chain. In the case of the point c, there were 3 points remaining in the chain so we jumped by 4α.

The reason this is OK is that we are on a maximal chain and so other points in other directions, e.g. x cannot accumulate more αs than the jump.

In fact in many cases we need less than $d(s)+1$. One strategy is to jump in each case by some variable letter $\mathbf{k}(s)$ and adjust it at the end so that all is OK.

Example 3.14 (Critical nodes). This example explains another problem we have to watch for. Consider the situation in Figure 21.

The longest chain in this figure is

$$w < z < y.$$

So we allocate

$$w : \alpha < z : 2\alpha.$$

We allocate $a : \beta$, and $b : \gamma$ and now since x comes immediately above a and b it gets $x : \alpha, \beta$ and similarly y gets $\gamma, \beta, 2\alpha$. This algorithm however makes $x < y$.

We need to recognise the critical points x such that there is a y such that $\sim (x < y)$ and y is such that it is above all the immediate predecessors of x. In such a case to avoid the result $x < y$, we add a label to x. We do not need to worry if x has only one predecessor. In such a case we add additional label to x anyway. It is only when x has more than one predecessor that we need to worry. Hence the definition of critical points in Definition 3.1.

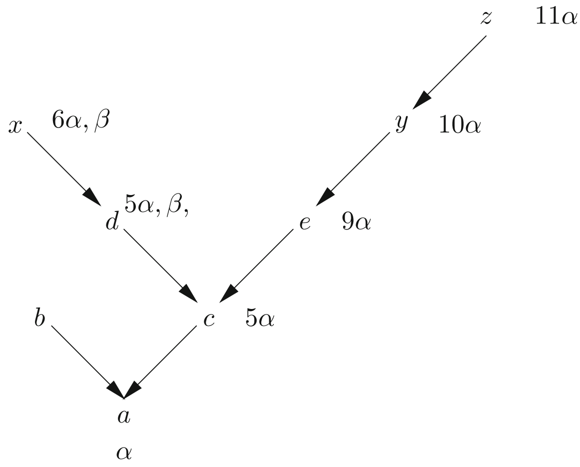


Figure 20.

So the labelling becomes as in Figure 22, where we added an additional label to the critical node x.

Critical points can be identified from the ordering.

In Figure 23, both x and y are critical, and so is z.

However, if we deal with x, y, we do not need to deal with z. Our algorithm will pay attention to do that.

Also if $a = b$, then x, y are not critical because we increase the allocation of both x and y.

EXAMPLE 3.15. We now apply our algorithm to be given in Construction 3.16 Figure 13.

Step 0 (Preparatory step)
Identify the set of critical points C and a maximal thin chain T.

In our case

$$\begin{aligned} &C = \{u\} \\ &T = \{x_1 < x_2 < x_3 < z < x_4 < v\} \end{aligned}$$

Identify the number of points for the graph. In Figure 13, $k = 12$.
Step 1
Consider all points of level 1. Give a different atomic label to each point. Give α to the point which is on the chain T. Let the function be λ. In our case

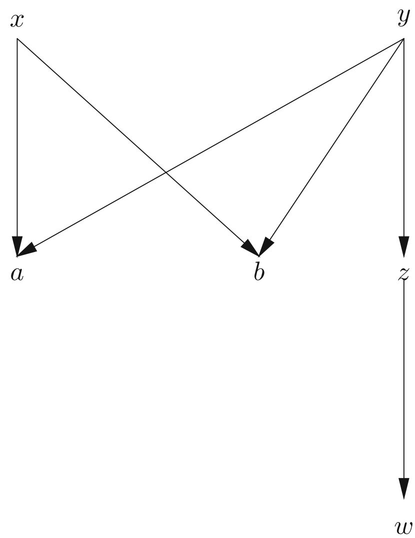


Figure 21.

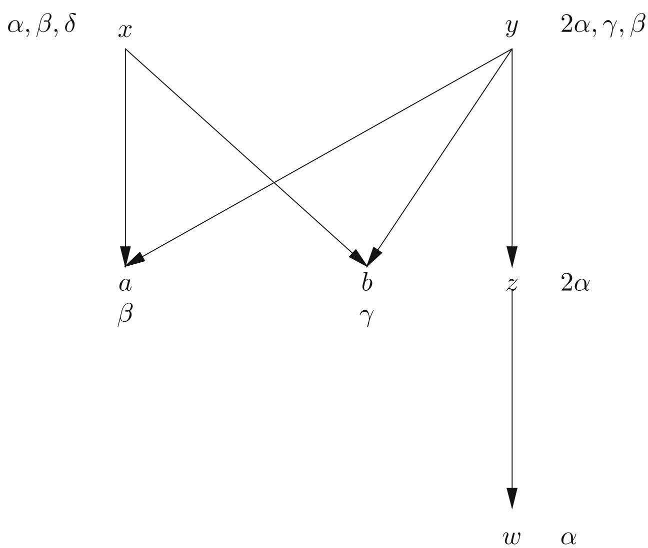


Figure 22.

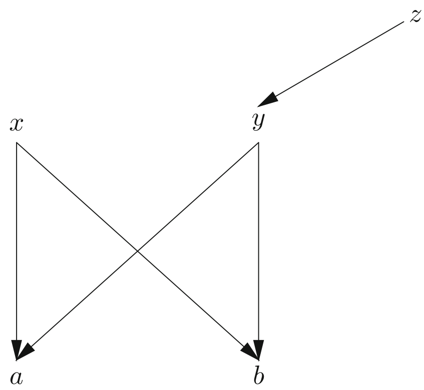


Figure 23.

$$\begin{aligned}\lambda(y_1) &= \beta_1\\ \lambda(y_2) &= \beta_2\\ \lambda(y_3) &= \beta_3\\ \lambda(x_2) &= \alpha\end{aligned}$$

Step 2

Identify the points of level 2.

In this case, we have point x_2 and u. x_2 is on the main chain T_1 and u is a critical point. We allocate

$$\begin{aligned}\lambda(x_2) &= 5\alpha\\ \lambda(u) &= \{\beta_1, \beta_2, \gamma\}.\end{aligned}$$

Step 3

Identify points of level 3. These are x_3 and w.

x_3 is immediately above x_2 and y_1, so we allocate

$$\lambda(x_3) = \{\beta_1, 5\alpha\}$$

w is above u and x_1 so

$$\lambda(w) = \{\alpha, \beta_1, \beta_2, \gamma\}.$$

Step 4

Identify level 4 points. This is z. It is on T, so we advance α (add α to the allocation of its predecessor)

$$\lambda(z) = \{\beta_1, 6\alpha\}$$

Step 5

Consider nodes of level 5. This is x_4. x_4 has two immediate predecessors, z and y_2. It is on T but we do not need to advance α because its allocation will increase anyway from y_2 and z. It is not critical so we do not need to add extra. So

$$\lambda(x_4) = \{\beta_1, \beta_2, 6\alpha\}$$

Step 6

Level 6 points are x_5 and v. We allocate

$$\lambda(x_5) = \{\beta_1, \beta_2, \beta_3, 6\alpha\}$$
$$\lambda(v) = \{\beta_1, \beta_2, 7\alpha\}$$

We advanced α to v because $v \in T$.

To summarise, we get Figure 24. The algorithm can be improved to decide more carefully how much to advance α but we don't do that in order to keep the algorithm simple. See Remark 3.17.

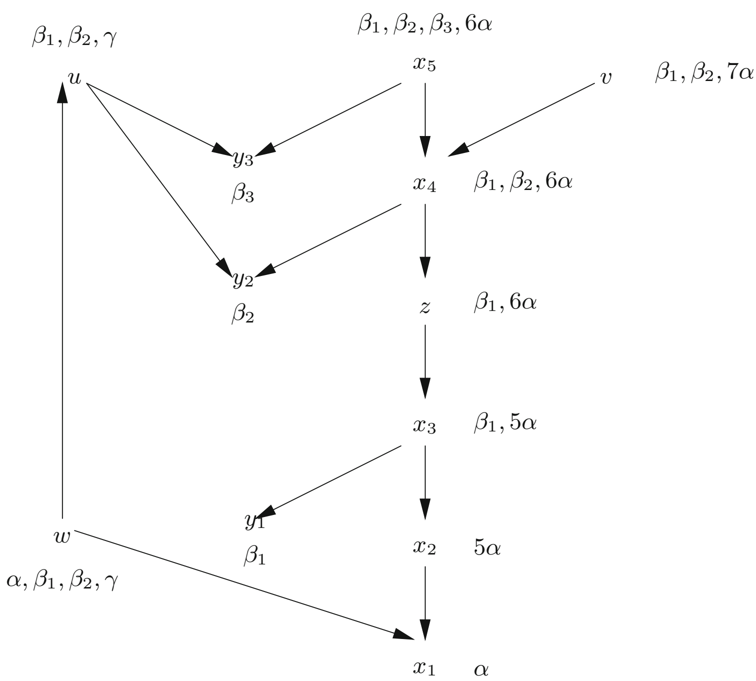


Figure 24.

We are now ready to give the algorithm for allocating multiset labels to any finite graph.

Construction 3.16 (Multiset realisations for finite graph). *Let $(S, <, \prec)$ be a graph, as defined in Definition 3.1. We shall construct a realisation λ on $(S, \prec)$ using the partition into levels of S as presented in Definition 3.1. The construction proceeds by induction on the levels. We assume we have an infinite sequence of labels $\{\alpha, \beta_i\}$ to use as we need. α is the only label that we allow to make copies, $2\alpha, 3\alpha$, etc.*

We define a function λ in steps 1, 2, 3,... defining λ in step n on all points of level n. Step 0 just prepares the ground for the induction by doing some preliminary processing.

Step 0
Identify and choose one maximal thin path in $(S, \prec)$, call it T. For example in Figure 13 this is T_1. Also identify all critical points.
Step 1
Consider all level 1 points (i.e. minimal points). One of them, say x_1 is in T. Let $\lambda(x) = \{\alpha\}$. If the other minimal points are $y_1, \ldots, y_n$ let $\lambda(y_i) = \beta_i$, β_i are all different labels. In Figure 13, $n = 3$.
Step $n + 1$
Consider all level $n + 1$ points. One of them, say w is in T. Let the others be $a_1, \ldots, a_k$. In Figure 13, for $n = 2$, we have $w = x_2$ is in T_1 and $a_1 = u$.

There are the following possibilities for a level $n + 1$ point e.

a. *e has only one predecessor and e is in T.*

b. *e has only one predecessor and e is not in T*

c. *e has several predecessors and e is the only one with these predecessors. We have two subcases:*

   c1. *e is a critical point*

   c2. *e is not critical*

d. *e and $e_1, \ldots, e_k$ have the same set of predecessors and $e \notin T$*

e. *e and $e_1, \ldots, e_k$ have the same set of predecessors and $e \in T$.*

In Figure 13 for level 2, point u is of Case (c1) and point x_2 is of case (a).

We now extend λ to the new level $n + 1$ points of the graph. Let e be of level $n + 1$. We use case analysis and define $\lambda(e)$ as follows:

Case (a)
Let y be the single predecessor of e. Then since $e \in T$, we also have $y \in T$.

We distinguish two subcases:

(a1) *e is the only immediate successor of y.*
Let $\lambda(e) = \lambda(y) \cup \{\alpha\}$.

(a2) *y has other immediate successors besides e. Let $m(y)$ be the remaining number of nodes above y in the chain T. Then let $\lambda(e) = \lambda(y) \cup \{(m+1)\alpha\}$.*

Note that the choice to advance by $(m+1)\alpha$ is safe but in many cases not minimal. Let $\mathbf{k}(y)$ be a variable letter which can take values $1 \leq \mathbf{k}(y) \leq m+1$. We can advance α by $\mathbf{k}(y)$, i.e. $\lambda(e) = \lambda(y) \cup \{\mathbf{k}(y)\alpha\}$. We can carry on the construction until we finish. We get allocations of multisets with some numerically bound variables $\mathbf{k}(y), y \in T$ in it. We can now check by a computer program what values of $\mathbf{k}(y)$ will maintain the graph ordering. These are the values we take. The program terminates because $\mathbf{k}(y) \leq m(y)+1$. See Remark 3.17.

Case (b)

Here $y \notin T$. We distinguish two subcases:

(b1) *$\alpha \in \lambda(y)$. Let $\lambda(e) = \lambda(y) \cup \{\alpha\}$.*

(b2) *$\alpha \notin \lambda(y)$. Let δ be a new atomic label and let $\lambda(e) = \lambda(y) \cup \{\delta\}$,*

(b3) *$\alpha \in \lambda(y)$ and there exists a point z is above y and also above some point in T. In this case let δ be new label and let $\lambda(e) = \lambda(y) \cup \{\delta\}$. See for example Figure 63, node G. Here $T = A < H < G, e = G, y = K$. Point Y is above K and if we advance α and let G have $2\alpha, \beta$ it will become below Y. The reason is that Y is above $H \in T$ and so gets more α from H.*

Case (c)

Let the predecessors of e be $y_1, \ldots, y_k$.

(c1) *e is not critical. Let D be the smallest multiset containing $\lambda(y_i)$ for all $i = 1, \ldots, k$. Let $\lambda(e) = D$.*

(c2) *e is critical. In this case let δ be a new atomic label. Let $\lambda(e) = D \cup \{\delta\}$.*

Case (d)

Let $y_1, \ldots, y_k$ be the predecessors. Let $\delta, \delta_1, \ldots, \delta_k$ be completely new set of labels.

Let $\lambda(e_i) = E \cup \{\delta_i\}$ where E is the smallest multiset containing all of $\lambda(y_i)$. Similarly $\lambda(e) = E \cup \{\delta\}$.

Case (e)

This case is like Case (d) except that $e \in T$. In this case we proceed just like Case (d), we take $\delta_i, i = 1, \ldots, k$ new atoms, let $\lambda(e_i) = E \cup \{\delta_i\}$. However, for e we take $\lambda(e) = E \cup \{(m+1)\alpha\}$.

REMARK 3.17. We are pretty sure that the previous construction gives a minimal realisation as far as the number of different letters is concerned. It does not minimise the number of copies of α. For a practical strategy, we find the number of letters and copies of α we need using the algorithm possibly with some variables $\mathbf{k}(y)$ and then use another complete adjustment program to optimise the allocations and assign values to the variables $\mathbf{k}(y)$.

It is like the Newton method for finding roots of a polynomial. We get an approximate root first and then use a computer to get a better solution. In our cse, get a realisation from the algoirthm possibly with some variables and then simplify and optimise it.

CONSTRUCTION 3.18 (Multiset realisation for matrices). *Let $\mathbb{A}$ be a matrix with $0, 1$ values construct the graph of the columns of $\mathbb{A}$. If V_1 and V_2 are two columns, define $V_1 \leq V_2$ iff for every row in the matrix, the value of V_1 is bigger than that of V_2 (thus larger values are lower in the order. Remember the more 0 in the column, the harder it is to achieve whatever that column represents, hence the column is higher in the ordering).*

We also write graphcially

$$V_1 \leftarrow V_2$$

So we get a graph $(S, \prec)$. Now apply the construction to get a realisation $\mathbf{f}$ for the graph. Since the elements of the graph are all the columns of the matrix, we get a multiset value assigned for each column. Suppose $\mathbb{A} = [a_{i,j}], i = 1, \ldots, m, j = 1, \ldots, n$. Then $V_j = (a_{1,j}, \ldots, a_{m,j})$. $\mathbf{f}(V_j), j = 1, \ldots, n$ is now available.

We can now compute a multiset value for each row. Let $U_i = (a_{i,1}, \ldots, a_{i,n})$ be the ith row. We define $\mathbf{f}(U_i)$ to be the smallest multiset containing all the column multisets $\mathbf{f}(V_j)$ for which $a_{i,j} = 1$. With this definition we get a multiset representation for the matrix $\mathbb{A} = [a_{i,j}]$. We have

$$a_{i,j} = 1 \text{ iff } \mathbf{f}(U_i) \supseteq \mathbf{f}(V_j).$$

This is a more minimal realisation than the one proposed in Lemma 3.11.

DEFINITION 3.19. Let $(S, <)$ be given. Define xRy as $x < y \vee y < x$. Let R^* be the transitive closure of R.

1. For $s \in S$, let $[s]$ be the set of all elements such that sR^*y. We get $S = S_1 \cup \ldots \cup S_k$ where each S_i is R^* connected, and for $i \neq j, S_i \cap S_j = \varnothing$ Let ξ be the number k of connected components.

2. Take two points x and y such that xR^*y. Then there exist $z_1, \dots, z_k$ such that
$$xR_1z_1R_0z_2R_1z_3, \dots, R_iz_kR_{1-i}y$$
where
$$R_1, R_0 \text{ are in } \{<, >\} \text{ and } R_1 \neq R_0$$
Let $\rho(x, y)$ be the minimal number k such that a sequence $z_1, \dots, z_k$ exists.
Let $\rho = \max_{x,y}\rho(x, y)$.
ρ measures the maximal number of changes in direction required to move from one point to another. This is a measure of the complexity of the ordering.

We call ρ the index of directional change is $(S, <, >)$ and ξ the index of connectivity.

Example 3.20. Consider Figure 25

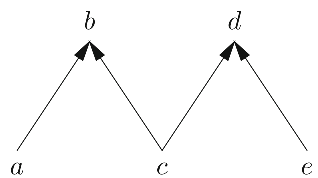


Figure 25.

To get from a to e we change direction three times. From c we can get to any point by changing direction once only. Here $\rho = 3$.

To get a better feel for this example, consider this ordering as a temporal ordering of temporal points. $a < b$ means b is in the future of a and a is in the past of b.

The temporal logic K_t has two temporal connectives FA and PA. We have

- $X \vDash FA$ iff $\exists x(x < y \wedge y \vDash A)$
- $X \vDash PA$ iff $\exists y(y < x \wedge y \vDash A)$

If A were true at point c, then we have
$$a \vDash FPFPA$$
We see that we require three changes of connectives.

$FPFP$ is known as a "modality", and in modal logic there are many theorems for many logics about how their modalities relate to one another.

Remark 3.21 (Summary of topological indices and their meanings).

1. **Number of points**
   This parameter is obvious. We have that less number of points makes a simpler ordering.
2. **Connectivity**
   This is a known topological notion. More connectivity make a simpler graph.
3. **Changes of direction**
   Less changes makes a better graph. As we have seen in Example 3.20, it makes for a simpler logic.
4. **Dimension**
   A realisation with lower dimension and lower index is better. I shows more connections in the graphs.
5. **Other indices**
   Note that the indices we use must have a direction. So if a small number of points is a good index, then the smaller the better. Consider for example, the graph theory criterion of how many arrows go into a point. In Figure 29 for example, node A has index 2 while node N has index 1. We now argue that this topological feature is not a good index.
   Consider Figure 14, the graphs of items 4, 5, and 7. In order of simplicity, 4 is best, 5 is middle and 7 is worst. The proposed graph index is one for 4, two for 5 and 0 for 7. So it has no direction, it just goes up and down.

We now discuss the meaning of a realisation and the meaning of dimension. First observe that giving realisations (or representations) is common in mathematics.

Representing algebraic structures by matrices is very common and also representing orderings by set inclusion. So the idea of representing a partial order by inclusion of multisets is a move every mathematicians will understand. The question is what more does it give us? We mentioned in Section 1.1 that the matrices should have the meaning that the rows are actions and the columns are features generated by the actions. We also argued that all features should be pulling in the same direction. We gave some examples. We give one more which will help us with our notation. Consider a matrix where the rows are types of foods and the columns are health features. Here is a partial matrix (Figure 26):

Eggs are bad for the veins, caffeine for the heart, carrots good for the eyes, etc.

	heart	blood flow	eyes	bones
carrots			1	
eggs		0		
coffee	0	1		
milk				1

Figure 26.

The reason behind this table are the specific ingredients the foods contain. It is the ingredients that do the job. If the columns are represented correctly, i.e. all pulling in the same general direction of better health, then 1 in the caffeine or egg column means quantity while 1 in the carrot column means larger quantity. So in this case, if the egg heart slot gets represented by say $\{3\alpha, \beta, \gamma\}$ this means there are some ingredients in eggs (say β, γ) that give the effects on the heart. α is the only label that can have a strength index, so here 3α means average strength in the direction of health.

Mathematically it is sufficient to allow for only one parameter to indicate strength.

Consider Figure 27

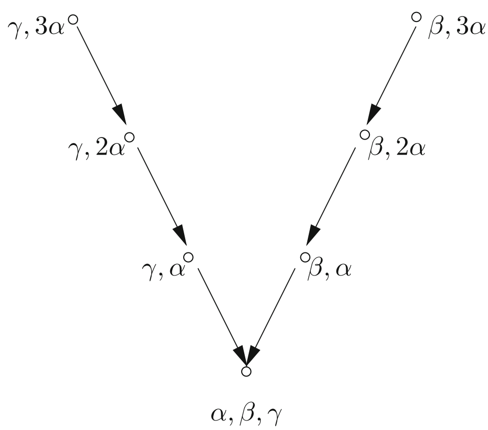


Figure 27.

β, γ give the qualitative directions and α gives the strength. This representation has dimension 3 (3 ingredients) and index 3 (strength of α).[6] So if we have an ordering which can be realised with less ingredients (dimension)

[6]Technically in the finite case we can manage with just α, β. So in our graph of Figure 27, let $\gamma = 3\alpha$ and the left branch becomes $4\alpha, 5\alpha, 6\alpha$. This representation is only technical. We want γ to indicate quality not quantity.

and lower index, then it is a better ordering, giving a more detailed picture of what is going on.

EXAMPLE 3.22 (Restriction on the representation). We saw that a lower dimension on the representation is an indication of a superior ordering. Therefore any restriction imposed on the representation might increase the dimension for the same ordering and we expect it must make sense. We best explain through an example. Consider the matrix of Figure 28

	N	A	P	Y
m	0	1	1	0
h	1	$x=?$	0	0
b	1	1	0	1

Figure 28.

We use the notation of capital letters for columns and small bold letters for rows.

To decide whether we should recommend $x = 0$ or $x = 1$, let us do the graphs for each case, and calculate the realisation using our algorithm

For the choice $x = 1$, we get Figure 29

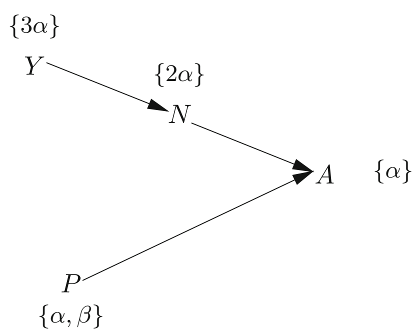


Figure 29.

For the choice $x = 0$ we get Figure 30.

We now have to check the various criteria for the two orderings.

The table in Figure 31 gives the answers:

The dimension and index needs to be checked against all possible realisations of the graphs. The following Figure 32 is a table of realisations with dimension 2 (labels $\{\alpha, \beta\}$). We need to prove that neither graph can manage to have a realisation with only one label $\{\alpha\}$. This is easy to see.

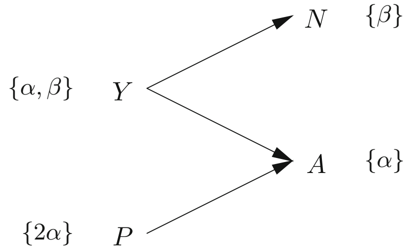


Figure 30.

	Dimension	connectivity	change direction	number points
Case $x = 1$	2	1	1	4
Case $x = 0$	2	1	2	4

Figure 31.

According to this $x = 1$ has a better ordering because it has less change of direction. So the answer to the abduction problem of Figure 28 is to take $x = 1$.

We can now compute the realisation $\mathbf{g}$ for the rows matrix of Figure 28, using the algorithm of Construction 3.18. We get two functions, $\mathbf{g}_1$ for the case $x = 1$ and $\mathbf{g}_0$ for the case $x = 0$. The values of $\mathbf{g}_1$ and $\mathbf{g}_0$ for the columns can be read from Figures 29 and 30 respectively.

To find what ingredient row i has in a representation, we must find a minimal multiset $\mathbf{g}(U_i)$ such that

$$\mathbf{g}(U_i) \supseteq \mathbf{f}(V_j), j = 1, \ldots, n, \quad a_{i,j} = 1.$$

So in the matrix for $x = 1$ we get

$$\begin{aligned} \mathbf{g}_1(\mathbf{m}) &= \{2\alpha, \beta\} \\ \mathbf{g}_1(\mathbf{h}) &= \{2\alpha\} \\ \mathbf{g}_1(\mathbf{b}) &= \{3\alpha\} \end{aligned}$$

For the matrix $x = 0$ we get

$$\begin{aligned} \mathbf{g}_0(\mathbf{m}) &= \{2\alpha\} \\ \mathbf{g}_0(\mathbf{h}) &= \{\beta\} \\ \mathbf{g}_0(\mathbf{b}) &= \{\alpha, \beta\} \end{aligned}$$

Let us now add the restriction that rows $\mathbf{m}$ and $\mathbf{b}$ (i.e. the actions $\mathbf{m}$ and $\mathbf{b}$) must contain an ingredient which is not in $\mathbf{h}$. For example, if the rows

	N	A	P	Y
Case $x = 1$	2α	α	α, β	3α
Case $x = 0$	β	α	2α	α, β

Figure 32.

are foods or medicines, we may know that there is something (e.g. vitamin present in **m** and **b** and not in **h**).

Both realisations $\mathbf{g}_0$ and $\mathbf{g}_1$ fail to satisfy the restriction.

The following Figures 33 for the case $x = 1$ and 34 for the case $x = 0$ give realisations which do satisfy the condition. However, the dimension goes up. We need to prove mathematically that it is not possible to give any realisation of dimension 2 which satisfies the restriction.

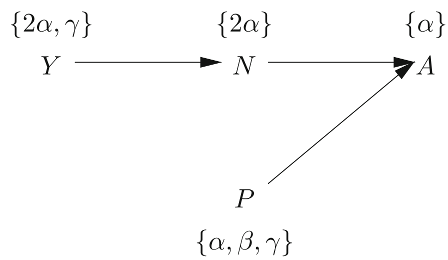


Figure 33.

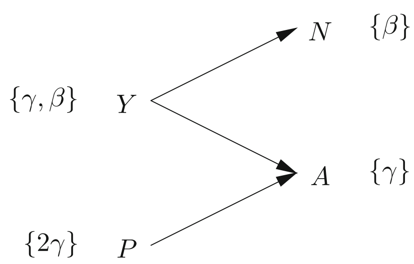


Figure 34.

Again the case $x = 1$ wins because of changes of direction. We get in this case, Figure 35

Clearly γ is an ingredient in **m** and **b** but not in **h**.

DEFINITION 3.23 (Superiority). Given two graphs τ_1 and τ_2 we calculate the value of the parameters for each graph.

	$x = 1$	$x = 0$
$\mathbf{g}(\mathbf{m})$	$\{\alpha, \beta, \gamma\}$	$\{2\gamma\}$
$\mathbf{g}(\mathbf{h})$	$\{2\alpha\}$	$\{\beta\}$
$\mathbf{g}(\mathbf{b})$	$\{2\alpha, \gamma\}$	$\{\gamma, \beta\}$

Figure 35.

1. If graphs τ_1 (resp. τ_2) has better or equal values to all parameters than τ_2, (resp. τ_1), with one parameter strictly better, then τ_1 is superior.
2. If all parameters are equal or if one graph is better in one parameter and the other graph is better in another then the answer is undecided.

We now describe how we reason and argue with these matrices. We imagine a proponent and an opponent.

The proponent puts forward a matrix $\mathbb{A}$ with $a_{i,j} =?$. All the entries except $a_{i,j}$ are considered known and agreed values. He applies the matrix abduction rule to $\mathbb{A}$ and proves (i.e. the algorithm of Section 3 shows) that $x = 1$ is a winning value. Thus the proponent showed non-monotonically using the matrix $\mathbb{A}$ and our rule that $x = 1$.

Remember that these entries have meaning: so for example if the entries are from monadic predicate logic with predicates $A_i(x), i = 1, \dots, m$ and the elements of the domain are $d_1, \dots, d_n$ then the proponent proved that $A_i(d_j) = 1$.

How can the opponent attack? He adds more facts to the argument by adding more columns and rows to the matrix. So assume $\mathbb{A} = [a_{i,j}], 1 \leq i \leq m, 1 \leq j \leq n$. The opponent expands the matrix to $\mathbb{A}^* = [a_{i,j}] 1 \leq i \leq m^*, 1 \leq j \leq n^*$, where $m \leq n^*$ and $n \leq n^*$ and at least one of $m \lneqq m^*$ and $n \nleqq n^*$ holds.

All the new entries in $\mathbb{A}^*$ are in $\{0, 1\}$ (i.e. $a_{i,j} =?$ is still the only unknown, ? entry). Furthermore, when we apply our matrix abduction algorithm to $\mathbb{A}^*$ we get 'undecided' as values.

The proponent can defend by expanding $\mathbb{A}^*$ to $\mathbb{A}^{**}$ where in $\mathbb{A}^{**}$ we do get that $a_{i,j} = 1$ is a winning value. This attack and defence can go on and on until it stops. The last value is the conclusion value.

Example 3.24. Here is a sequence of attacks and counterattacks

In the matrix $\mathbb{A}$ of Figure 36 without column $Y, x = 1$ wins. When we add Y to it to get $\mathbb{A}^*$ we get 'undecided'.

	N	A	Y
h	1	$x =?$	0
b	1	1	1

Figure 36.

Graph of $\mathbb{A}$ for $x = 1$

$$\bullet$$
$$N \quad = \quad A$$

Graph of $\mathbb{A}$ for $x = 0$

$$A \longrightarrow N$$

Here $x = 1$ wins because it has less points ($N = A$)

Figure 37.

Let us check this, see Figures 37 and 38.

Example 3.25 (Screens and cameras). We can now settle the question of whether the NEC2470/WVX LCD screen has stereophonic speakers.[7]

Let us do the graph for the columns of the matrix of Figure 1. This is displayed at Figure 39.

Clearly the case $x = 1$ is superior because it is connected.

We now consider the camera example of Figure 2.

The two graphs for case $x = 1$ and case $x = 0$ are displayed in Figure 40

Again the case $x = 1$ wins, because for $x = 0$ we get a disconnected graph.

Remark 3.26 (Two question marks). How do we deal with abduction matrices which have more than one slot with a question mark? Say, for example $a_{i_0,j_0} =?$ and $a_{i_1,j_1} =?$. The simplest course of action is to substitute all possible $0, 1$ values and see which combination wins.

In our Hebrew paper [1] there are examples of this sort. It is possible to find, for example, that no matter what the value of a_{i_0,j_0} is we have that $a_{i_1,j_1} = 1$ is a winning substitution (over the option $a_{i_1,j_1} = 0$) while when we substitute $a_{i_1,j_1} = 1$ in the matrix, we find no winning value for $a_{i_0,j_0} =?$.

[7]Searched the web again today, 1 March 2009. Could not find a definite answer.

Graph of $\mathbb{A}^*$ for $x = 1$

$$Y \longrightarrow N, A$$

Graph of $\mathbb{A}^*$ for $x = 0$

$$A, Y \longrightarrow N$$

Neither graph wins.

Figure 38.

EXAMPLE 3.27 (Dependency of columns). Assume we are given a definite $\{0, 1\}$ matrix $\mathbb{A}$. Our procedures so far for handling it run through the following steps:

1. Describe the column graph of the matrix
2. Apply an algorithm to find a minimal multiset realisation of the graph
3. Use the above to find a minimal multiset realisation $\mathbf{f}$ of the matrix.

Let us concentrate our attention on (3) above.

First note that if we have a multiset realisation of the matrix we can get (1) and (2) anyway. Namely, we have

(*) $a_{i,j} = 1$ iff $\mathbf{f}(U_i) \supseteq \mathbf{f}(C_j)$ where U_i is row i and C_j is column j of the matrix.

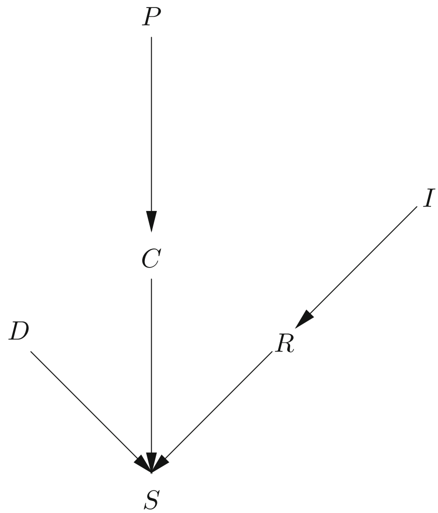
(**) We have $C_j \leq C_k$ iff $\mathbf{f}(C_j) \subseteq \mathbf{f}(C_k)$

Thus $\mathbf{f}$ actually determines everything using (*) and (**).

We now consider a new type of restriction on the realisation $\mathbf{f}$. We call it column dependencies. It arises from applications and it is a new interpretation of the matrix entries.

Consider the matrix of Figure 1. Consider the two columns R-reaction time and D-dot size. We can easily imagine that for technical reasons, the reaction time of Screen 3 is enhanced because Screen 3 has also small dot size. In other words the technical modifications required to make small dot size also help with reaction time. The entry therefore for Screen 3 column

Case $x = 1$



Case $x = 0$

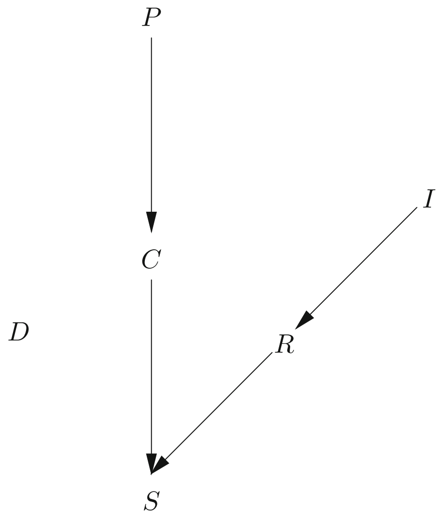
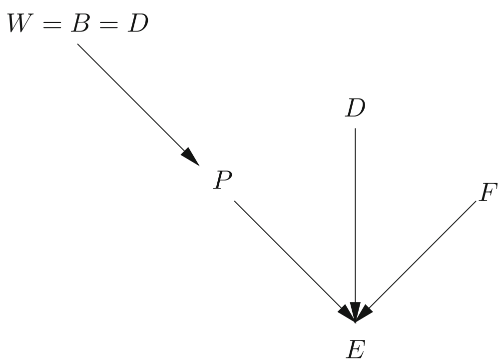


Figure 39.

Case $x = 1$



Case $x = 0$

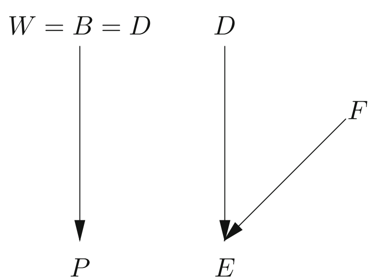


Figure 40.

R depends (or is helped by) the entry for Screen 3 Column D. How do we express this formally? The answer is that it manifests itself in the restrictions on the realisation $\mathbf{f}$ of the matrix of Figure 1.

Instead of the equation(*)

- $a_{3,R} = 1$ iff $\mathbf{f}$(Screen 3) $\supseteq \mathbf{f}(R)$

we have

- $a_{3,R} = 1$ iff $\mathbf{f}$(Screen 3) $\cup \mathbf{f}(D) \supseteq \mathbf{f}(R)$.

It is clear that $\mathbf{f}$(Screen 3) is helped by the multiset $\mathbf{f}(D)$.

This example prompts us to change (*) to (*1) as follows: Given a definite matrix $\mathbb{A}$ with dependencies $\mathbb{D}$ of the form

- $a_{i,j} = 1$ provided column j depends on columns $k_1^{i,j}, \ldots, k_r^{i,j}$.

Then a function $\mathbf{f}$ assigning multisets to rows and columns is a realisation of $(\mathbb{A}, \mathbb{D})$ provided (*1) holds

(*1) $\quad a_{i,j} = 1$ iff $\mathbf{f}(U_i) \cup \bigcup_{m=1}^{r_{i,j}} \mathbf{f}(C_{k_m^{i,j}}) \supseteq \mathbf{f}(C_j)$.

To see the difference in a real Talmudic example, consider the argument of Figure 46 below for the cases $x = 1$ and $x = 0$.

The graphs for it are in Figure 47. The realisation we get from the graphs are $\mathbf{f}_1$ and $\mathbf{f}_0$ as follows. See Figure 41 below.

Case $x = 1$		Case $x = 0$	
$\mathbf{f}_1(A)$	α	$\mathbf{f}_0(A)$	α
$\mathbf{f}_1(N)$	2α	$\mathbf{f}_0(N)$	β
$\mathbf{f}_1(\mathbf{m})$	α	$\mathbf{f}_0(\mathbf{m})$	α
$\mathbf{f}_1(\mathbf{h})$	2α	$\mathbf{f}_0(\mathbf{h})$	β

Figure 41.

The Talmudic argument is for $x = 1$ to win, as indeed it does.

This argument is attacked by claiming that there is a dependency, and that actually the value $a_{\mathbf{h},N}$ at the square $(\mathbf{h}, N)$ depends on the column A. Thus we need a new matrix realisation $\mathbf{f}^*$ which satisfies

$$a_{\mathbf{h},N} = 1 \text{ iff } \mathbf{f}^*(\mathbf{h}) \cup \mathbf{f}^*(A) \supseteq \mathbf{f}^*(N).$$

If this is the case we get the following realisations $\mathbf{f}^*$ for the matrices, for cases $x = 0$ and $x = 1$. See Figure 42.

We need to formulate algorithms to find minimal realisations for matrices $(\mathbb{A}, \mathbb{D})$ with dependencies as well as criteria to compare realisations, but clearly the attack succeeds as Case $x = 1$ of Figure 42 is not superior to Case $x = 0$.

Case $x = 1$		Case $x = 0$	
$\mathbf{f}_1^*(A)$	α	$\mathbf{f}_0^*(A)$	α
$\mathbf{f}_1^*(N)$	2α	$\mathbf{f}_0^*(N)$	1.5α
$\mathbf{f}_1^*(\mathbf{m})$	α	$\mathbf{f}_0^*(\mathbf{m})$	α
$\mathbf{f}_1^*(\mathbf{h})$	α	$\mathbf{f}_0^*(\mathbf{h})$	0.5α

Figure 42.

4. Case study: Sentences for traffic offences

As an additional example of our general method we shall consider an application in the domain of Traffic Offences. We stress that it is not our intention to suggest that traffic judges should actually use a computer system implementing this approach. It is though conceivable that some day in the future decision support systems for judges could incorporate this method.

Let us first briefly survey the area of sentencing within the framework of law and order in society. This is usually the point of view taken by judges about to pass sentence on an offender. One distinguishes four classical approaches to punishment: Retribution, Deterrence, Prevention and Rehabilitation [Lawton L.J., in: Sargeant (1974) 60 Cr. App. Rep. 74 C.A. at pp.77-84].

This classification does not mean that a judge about to pass sentence on an offender asks himself an explicit question: "Which approach shall I use here?" It is generally assumed that he forms an opinion about which approach (or approaches) to apply in an intuitive manner. The next step is then to decide on a sentence appropriate for the specific offender and offence within the sentencing approach (or approaches). This is also an intuitive process: "It comes from within", as several judges have expressed it. Our formulation would be that the sentence chosen by the judge is the one that he intuitively believes includes the appropriate mixture of 'microscopic elements', thus leading to one or more of the four sentencing aims.

The possible punishments for traffic offences are (in Israel): Imprisonment, driving disqualification, fine, community service. These punishments may also be suspended, i.e. applied only if the offender commits a new offence. Sentences are often a combination of the above, e.g., suspended driving disqualification plus fine. In the following we shall not consider community service.

Consider now the following example. A judge has passed sentence on six traffic offenders. He has already decided on part of a sentence for a seventh offender, but has not made up his mind whether to sentence him to

imprisonment or not. Our algorithm will indicate what would be a logical decision, based on the previous six sentences.
Offender 1: Killed a pedestrian. The sentence: Suspended imprisonment and actual disqualification.
Offender 2: Killed pedestrian while driving under licence disqualification. The sentence: Imprisonment, disqualification, and also suspended imprisonment and suspended disqualification.
Offender 3: Drunk driving. The sentence: Disqualification, fine, and also suspended disqualification and suspended fine.
Offender 4: Driving through a red light. The sentence: Fine and suspended disqualification.
Offender 5: Driving while under disqualification. The sentence: Disqualification, fine and suspended fine.
Offender 6: Driving without valid driver's licence. The sentence: Fine and suspended fine.
Offender 7: Driving while under the influence of drugs. The sentence: Disqualification, fine and suspended disqualification and suspended fine. In addition the judge has decided on suspended imprisonment and is thinking about an actual prison sentence.

What advice should we give him, in order to preserve a logical uniformity of sentencing?

It is important to assume that the above sentences were passed by the same judge. Only by this assumption can we be sure that the "microscopic ingredients' are the same. For different judges may use different ingredients in different amounts.

This of course is strongly related to the problem of uniformity of punishment, or rather the lack of uniformity. Were different judges to use the same ingredients in the same amounts the problem of sentencing disparity would diminish. It is interesting to speculate that iterative use of our approach on sentences by different judges could lead to some kind of convergence bringing increased uniformity.

The following table 43 represents the cases described above. The undecided sentence for offender 7 is indicated by a ? sign.

The following diagram 44 represents the choice: actual imprisonment, i.e. the value 1 is substituted for the question mark.

The following diagram 45 represents the choice: No actual imprisonment, i.e., the value 0 is substituted for the question mark.

We immediately realize that the first diagram is superior to the second one. This means that our approach will recommend that the judge imposes an actual prison sentence on offender seven.

	O1	O2	O3	O4	O5	O6	O7
imprisonment	0	1	0	0	0	0	?
suspended imprisonment	1	1	0	0	0	0	1
disqualification	1	1	1	0	0	0	1
suspended disqualification	0	1	1	1	1	0	1
fine	0	0	1	1	1	1	1
suspended fine	0	0	1	0	1	1	1

Figure 43.

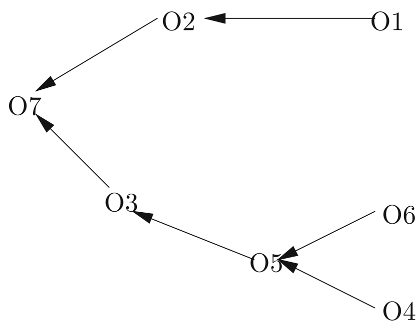


Figure 44.

5. Analysis of the Talmudic Kal-Vachomer from Kidushin 5a-5b

The Talmud was finalised in the fifth century AC. It contains many legal arguments about a variety of issues and one of the rules used was the Kal-Vachomer. The following text is one of the most complicated uses of this rule. The rule has never been properly formulated, though there have been many attempts.

Louis Jacobs [18], distinguishes two types of Kal-Vachomer. The simple one and the more complex one. The simple one has the structure:

- If A has x then B certainly has x.

The complex one has the structure

- If A, which lacks y, has x, then B which has y certainly has x.

The following are examples of the simple case from the Old and New Testaments. We already saw examples of the complex case in Sections 1.2 and 1.3.

The Bible does not contain instances of the complex case. This has emerged later, after the Bible.

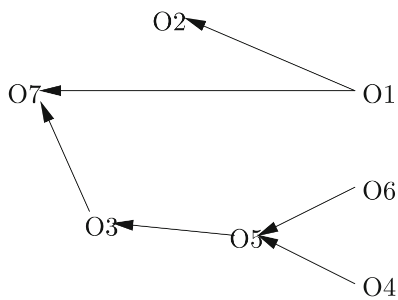


Figure 45.

Example 5.1 (Kal Vachomer in the Old and New Testaments).

Exodus 6
10 And Jehovah spake unto Moses, saying,
11 Go in, speak unto Pharaoh king of Egypt, that he let the children of Israel go out of his land.
12 And Moses spake before Jehovah, saying, Behold, the children of Israel have not hearkened unto me; how then shall Pharaoh hear me, who am of uncircumcised lips?

Deuteronomy 31
27 For I know thy rebellion, and thy stiff neck: behold, while I am yet alive with you this day, ye have been rebellious against Jehovah; and how much more after my death?

Matthew 12
9 And he departed thence, and went into their synagogue:
10 and behold, a man having a withered hand. And they asked him, saying, Is it lawful to heal on the sabbath day? that they might accuse him.
11 And he said unto them, What man shall there be of you, that shall have one sheep, and if this fall into a pit on the sabbath day, will he not lay hold on it, and lift it out?
12 How much then is a man of more value than a sheep! Wherefore it is lawful to do good on the sabbath day.

Luke 13
14 And the ruler of the synagogue, being moved with indignation because Jesus had healed on the sabbath, answered and said to the multitude, There are six days in which men ought to work: in them therefore come and be healed, and not on the day of the sabbath.
15 But the Lord answered him, and said, Ye hypocrites, doth not each one

of you on the sabbath loose his ox or his ass from the stall, and lead him away to watering?
16 And ought not this woman, being a daughter of Abraham, whom Satan had bound, lo, /these/ eighteen years, to have been loosed from this bond on the day of the sabbath?
17 And as he said these things, all his adversaries were put to shame: and all the multitude rejoiced for all the glorious things that were done by him.

Romans 5
8 But God commendeth his own love toward us, in that, while we were yet sinners, Christ died for us.
9 Much more then, being now justified by his blood, shall we be saved from the wrath /of God/ through him.
10 For if, while we were enemies, we were reconciled to God through the death of his Son, much more, being reconciled, shall we be saved by his life;
11 and not only so, but we also rejoice in God through our Lord Jesus Christ, through whom we have now received the reconciliation.

The simple Kal-Vachomer was analysed as an Aristotelian syllogism by A. Schwarz [27]. Compare Barbara with what Moses says:

Barbara:

1. All men are mortal.
2. Socrates is a man,
   therefore
3. Socrates is mortal

1. $\forall x(\text{Men}(x) \to \text{Mortal}(x)$
2. Men(Socrates)
   therefore
3. Mortal(Socrates)

Deuteronomy 31:
Let s be something Moses says or demands. We have

1. $\forall x \neg\, \text{ListenIsrael}(x) \to \neg\text{ListenPharaoh}(x)$
2. $\neg\text{ListenIsrael}(s)$
   therefore
3. $\neg\text{ListenPharaoh}(x)$.

Louis Jacobs refutes the similarity, see [18, chapter 1], see also [22].

Certainly the more complex cases of Kal-Vachomer are not syllogisms at all.

We now analyse one of the most involved arguments in the Talmud. We first quote the text. A detailed analysis of the Kal-Vachomer in general and of this text in particular can be found in our companion paper in Hebrew [1].

Consider the following text from the Talmud, Kidushin 5a–5b.[8]

(1a) Rav Huna said: *Huppa* acquires a fortiori, since money, which does not allow one to eat *teruma* does acquire, *Huppa* which allows one to eat *teruma*, how much more should it acquire.
(1b) And does money not allow one to eat. But ‘Ulla said: According to the Torah, a betrothed Israelite daughter eats of *teruma*, for it is said “*But if a priest acquire any soul, the acquisition of his money*”, and this is the acquisition of his money. For what reason did they say that she does not eat? It was feared that a cup may be poured for her in her father’s house, and she will let her brothers and sisters drink it.
(2) Rather argue thus: If money, which does not finalise, does acquire *Huppa*, which does finalise how much more should it acquire.
(3) As to money, it is because one can redeem it with *heqdeshoth* and the Second Tithe.
(4) Intercourse shall prove it.
(5) As to intercourse, it is because it acquires in the case of a *Yevama*.
(6) Money shall prove it.
(7) And the inference revolves; the character of this is not like the character of that is not like the character of this:
(8) the feature common to them is that they acquire elsewhere and they acquire here. I can also bring *Huppa*, which acquires elsewhere and acquires here.
(9) As to the feature common to them, that is that their enjoyment is great!
(10) ‘Writ’ shall prove.
(11) As to writ, that it sets free an Israelite daughter,
(12) money and intercourse shall prove.
(13) Again the inference revolves. The character of this is not like the character of that and the character of that is not like the character of this.

[8] The insert numbers refer to the steps in arguments. The translation is from the El-Am edition.

(14) The feature common to them is that they acquire elsewhere and they acquire here. I can also bring *Huppa* which acquires elsewhere and acquires here.
(15) As to the feature common to them, it is that they are possible by compulsion.
(16) And Rav Huna? Money, however, we do not find in matrimony by compulsion.[9]

We now give the argument sequence of the text above. First, some background material. When a boy wants to wed a girl as his wife, he can do it in stages. First he can give her a ring and if she accepts they are engaged. The text refers to this state as *Kidushin*. It has to be done by giving the girl a ring or something of value. The important point is the value. So the text refers to the act as "money" (i.e. something of value). The next step is the marriage ceremony which the text calls '*Huppa*'. It is known that the ceremony is essential for marriage and cannot be replaced by another ring. So if the boy gives the girl another ring this does not make her his wife. She just gets a second ring for noting. He has to go through the ceremony. There are other options for marriage. For example they can be together in a 'familiar way', which can be anything you are not supposed to do otherwise, e.g. a kiss or a short period alone in a room (long enough to be naughty), etc. This is why for example, in a marriage ceremony the boy is allowed to kiss the bride. The text calls this 'intercourse'. The question we ask here is whether the marriage ceremony can do the job of the ring. So imagine that you are ready to get married and your silly best man forgot the ring. Can we go on or do we actually have to wait for the ring? The first argument by the proponent named Rav Huna is to prove that the ceremony itself can do the job of the ring, i.e. it can do the engagement as well as the marriage itself. Figure 46 is this argument (item 2 in the text).

	N = married	A = engaged
m (ring)	0	1
h (*Huppa*)	1	$x = ?$

Figure 46.

[9] Glossary
Money = ring
Kidushin = engaged
Huppa = religious marriage ceremony
Writ = official document/contract.

We do the graphs for $x = 1$ and $x = 0$, and get Figure 47

1. Graphs for $x = 1$

$$\begin{array}{ccc} N & \longrightarrow & A \\ 2\alpha & & \alpha \end{array}$$

2. Graph for $x = 0$

$$\begin{array}{cc} \bullet & \bullet \\ N & A \\ \beta & \alpha \end{array}$$

Figure 47.

Clearly the case $x = 1$ is a better graph. It is more connected. We also wrote the multiset assignment in the figure.

The opponent (the audience to Rav Huna) attacks this in item 3 in the text and adds column 3 to the matrix. See Figure 48.

	N	A	P
m	0	1	1
h	1	$x =?$	0

Figure 48.

The two graphs are now in Figure 49.

In Figure 49 the result is undecided. The graph of $x = 1$ is better in the aspect of being connected, and the graph for $x = 0$ is better in the aspect of having less points. So it is a draw and the verdict is 'undecided'.

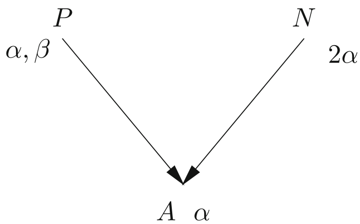
This means the proof of the proponent fails to be conclusive.

Now the proponent tries again (item 4 in the text) and presents a different table, using **b**= Intercourse, see Figure 50.

The two graphs are in Figure 51, and clearly $x = 1$ wins.

The opponent now attacks by adding column Y for the case of Yevama (item 5 of the text). We explain this case: an unmarried man must marry the widow of his brother if she is without children by biblical law. This cannot be done by ceremony (*Huppa*) but must be done by familiarity, **b**. In practice of course, if they don't want to marry then a 'divorce'-like procedure must be done.

1. Graph for $x = 1$



2. Graph for $x = 0$

$$\begin{array}{cc} N\bullet & \bullet A = P \\ \beta & \alpha \end{array}$$

Figure 49.

	N	A
b	1	1
h	1	$x = ?$

Figure 50.

1. Graph for $x = 1$

$$\begin{array}{ccc} \bullet & & \\ A & = & N \\ \alpha & & \end{array}$$

2. Graph for $x = 0$

$$\begin{array}{ccc} A & & N \\ & \longrightarrow & \\ 2\alpha & & \alpha \end{array}$$

Figure 51.

We get Figure 52

	N	A	Y
b	1	1	1
h	1	$x = ?$	0

Figure 52.

The graphs we get are in Figure 53

1. Graph for $x = 1$

$$\begin{array}{ccc} Y & \longrightarrow & N = A \\ 2\alpha & & \alpha \end{array}$$

2. Graph for $x = 0$

$$\begin{array}{ccc} A = Y & \longrightarrow & N \\ 2\alpha & & \alpha \end{array}$$

Figure 53.

Clearly they are of equal strength and the answer is undecided.

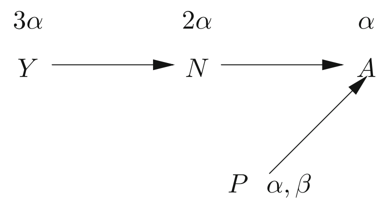
The proponent now combines both tables to get $x = 1$ to win(items 6–8 of the text). We get Figure 54

	N	A	P	Y
m	0	1	1	0
h	1	$x = ?$	0	0
b	1	1	0	1

Figure 54.

The two graphs are in Figure 55.

1. Graph for $x = 1$



2. Graph for $x = 0$

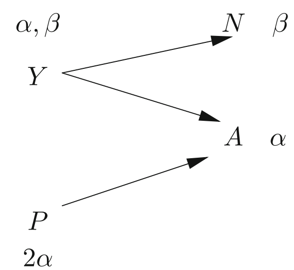


Figure 55.

The case $x = 1$ wins because it has only value 1 for change of direction. In the graph for $x = 0$, to get from N to P we have to change direction twice.

The opponent now attacks by adding a column of H (pleasurable activity, item 9 in the text). He argues that money (**m**) and intercourse (**b**) give pleasure, while the marriage ceremony **h** does not.

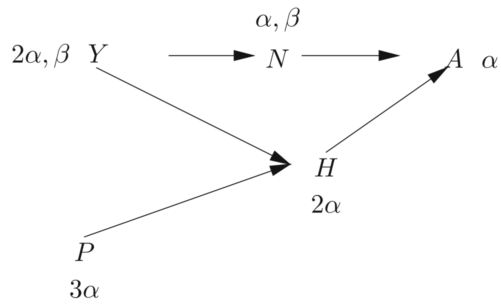
We get Figure 56

	N	A	P	Y	H
m	0	1	1	0	1
h	1	$x = 1$	0	0	0
b	1	1	0	1	1

Figure 56.

The graphs are in Figure 57 and the result is undecided

1. Graph for $x = 1$



2. Graph for $x = 0$

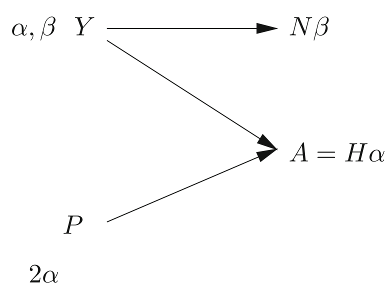


Figure 57.

Clearly case $x = 1$ has the advantage in change of direction index (1 change of direction while the graph for $x = 0$ has 2 changes), but the case $x = 0$ identifies $A = H$ and this gives it advantage over the case $x = 1$. The result is a draw.

The proponent tries again by adding a column G for divorce which can be done by a writ, **w**. This discussion is in items 9–14 of the text. Figure 58 gives the table.

The graphs for this are given in Figure 59.

The case $x = 1$ wins because its graph has only one change of direction and the graph of $x = 0$ has two changes.

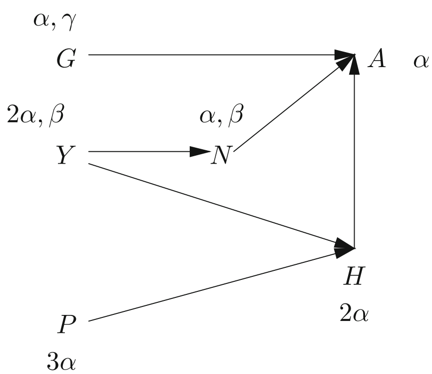
The opponent now attacks by adding a column with K — meaning without consent, like a girl being married by her parents without her consent. A practice still followed by some parts of the world. This is item 15 of the text.

We get the following matrix of Figure 60.

	N	A	P	Y	H	G
m	0	1	1	0	1	0
h	1	$x = ?$	0	0	0	0
b	1	1	0	1	1	0
w	0	1	0	0	0	1

Figure 58.

1. Graph $x = 1$



2. Graph $x = 0$

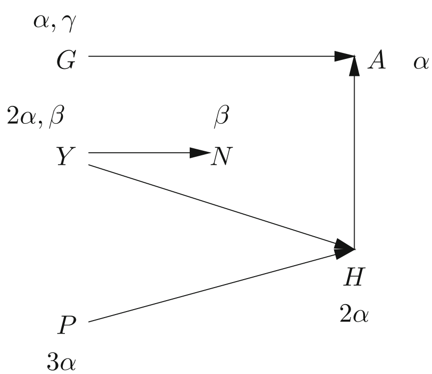


Figure 59.

	N	A	P	Y	H	G	K
m	0	1	1	0	1	0	1
h	1	$x =?$	0	0	0	0	0
b	1	1	0	1	1	0	1
w	0	1	0	0	0	1	1

Figure 60.

The two graphs are given in Figure 61.

The comparison of the two graphs is undecided. The graph for $x = 1$ has the advantage of one change of direction, as compared with that of $x = 0$ which has 2. On the other hand, the graph for $x = 0$ has the advantage of making $A = K$ i.e. has less points.

So it is an undecided draw and so the proponent has not successfully proved that $x = 1$ wins.

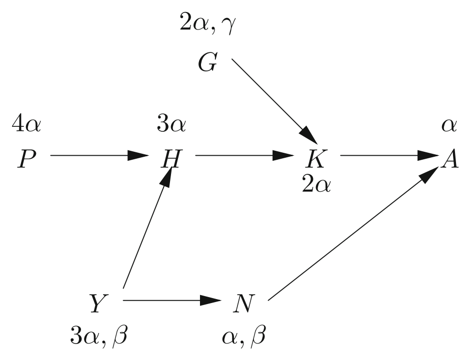
The proponent counters that he disagrees with the matrix of Figure 60, in which the opponent put value 1 in the slot $(\mathbf{m}, K)$. The proponent's opinion is that a value 0 should be there. This gives us the matrix of Figure 62. This corresponds to item 16 in the text.

The two graphs are in Figure 63

Clearly Case $x = 0$ is inferior because it has 2 changes in direction to get from G to N.

This completes the analysis of the text.

1. Graph for $x = 1$



2. Graph for $x = 0$

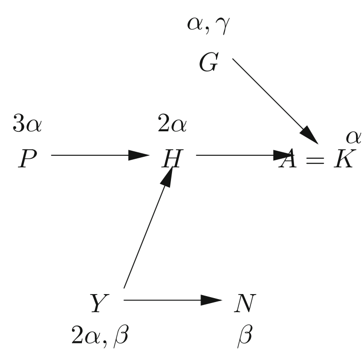


Figure 61.

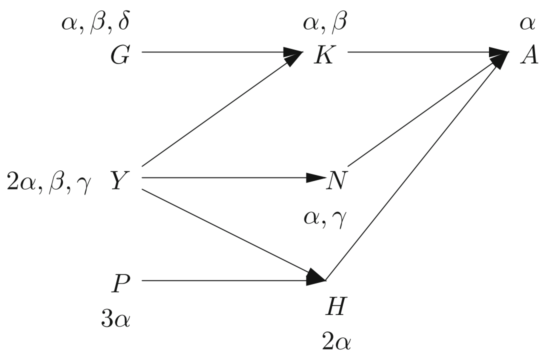
Remark: Achievement

Let us discuss what we have done here. We built a matrix abduction model whose components and concepts use only topologically meaningful notions (and hence the model is culturally independent) and we used it to analyse an involved Talmudic argument and we got a perfect and meaningful fit.

	N	A	P	Y	H	G	K
m	0	1	1	0	1	0	0
h	1	?	0	0	0	0	0
b	1	1	0	1	1	0	1
w	0	1	0	0	0	1	1

Figure 62.

1. Graph for $x = 1$



2. Graph for $x = 0$

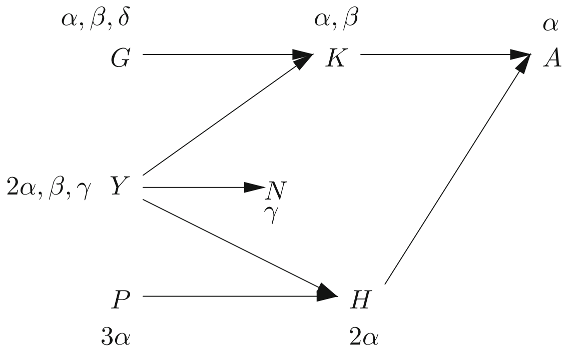


Figure 63.

6. Conclusion and discussion

This paper introduced a new method of abduction, which we called matrix abduction and showed that it can be applied in a variety of areas. The method arose directly from the study of the Talmudic non-deductive inference rules of Kal-Vachomer, the Argumentum A Fortiori. See our Hebrew paper [1] for a very detailed analysis.

We would like in this concluding section to make some epistemological comments.

Jacob Neusner [23] has argued (1987) that Talmudic thinking is very differnet from western thinking that produced science. This explains why the Jewish people through the ages did not make scientific achievements to the level of other nations. This view has been strongly criticised by other writers such as M. Fisch [10], who argues to the contrary, that rabbinic thinking is very similar to that of western science.[10]

We put forward to the reader that our paper exemplifies and supports the claim by M. Fisch. Matrix abduction is a new form of induction, arising from the Talmud, which can solve problems currently in the scientific community. (See [13] for a comprehensive treatise on abduction.)

We will venture to say that the logic of the Talmud is far richer and complex than currently available western logic. We hope to systematically investigate the logic in the Talmud in a series of papers and monographs.

Appendix

General applications of matrix abduction

A. Application to argumentation networks

Argumentation networks were introduced by P. M. Dung in a seminal paper in 1995. Since then a strong community arose working in the area. Our matrix abduction ideas can make a contribution to this area, as we shall now discuss.

An abstract argumentation network has the form (S, R), where S is a nonempty set of arguments and $R \subseteq S \times S$ is an attack relation. When $(x, y) \in R$, we say x attacks y.

The elements of S are atomic arguments and the model does not give any information on what structure they have and how they manage to attack

[10] Jews have had hard life throughout history. We don't think they had the same opportunities as western scientists.

each other.

The abstract theory is concerned with extracting information from the network in the form of a set of arguments which are winning (or 'in'), a set of arguments which are defeated (or are 'out') and the rest are undecided. There are several possibilities for such sets and they are systematically studied and classified. See Figure 64 for a typical situation. $x \to y$ in the figure represents $(x, y) \in R$.

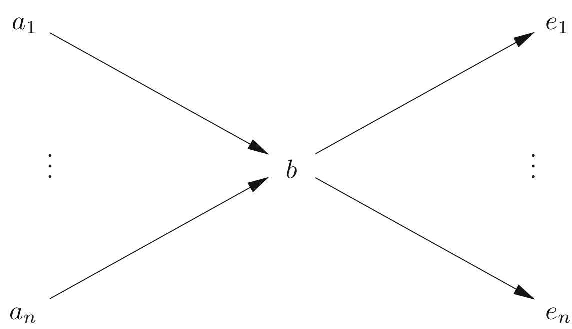


Figure 64.

A good way to see what is going on is to consider a Caminada labelling. This is a function λ on S distributing values $\lambda(x), x \in S$ in the set {in, out, ?} satisfying the following conditions.

1. If x is not attacked by any y then $\lambda(x) = 1$.
2. If $(y, x) \in R$ and $\lambda(y) = 1$ then $\lambda(x) = 0$.
3. If all y which attack x have $\lambda(y) = 0$ then $\lambda(x) = 1$.
4. If one y which attack x has $\lambda(y) =?$ and all other y have $\lambda(y) \in \{0, ?\}$ then $\lambda(x) =?$.

Such λ exist whenever S is finite and for any such λ, the set $S^+_\lambda = \{x \mid \lambda(x) = 1\}$ is the set of winning arguments, $S^-_\lambda = \{x \mid \lambda(x) = 0\}$ is the set of defeated arguments and $S^?_\lambda = \{x \mid \lambda(x) =?\}$ is the set of undecided arguments.

The features of this abstract model are as follows:

1. Arguments are atomic, have no structure.
2. Attacks are stipulated by the relation R; we have no information on how and why they occur.

3. Arguments are either 'in' in which case all their attacks are active or are 'out' in which case all their attacks are inactive. There is no in between state (partially active, can do some attacks, etc.). Arguments can be undecided.
4. Attacks have a single strength, no degrees of strength or degree of transmission of attack along the arrow, etc.
5. There are no counter attacks, no defensive actions allowed or any other responses or counter measures.
6. The attacks from x are uniform on all y such that $(x, y) \in R$. There are no directional attacks or coordinated attacks. In Figure 64, $a_1, \ldots, a_n$ attack b individually and not in coordination. For example, a_1 does not attack b with a view of stopping b from attaching e_1 but without regard to $e_2, \ldots, e_n$.
7. The view of the network is static. We have a graph here and a relation R on it. So Figure 64 is static. We seek a λ labelling on it and we may find several. In the case of Figure 64 there is only one such λ. $\lambda(a_i) = 1, \lambda(b) = 0, \lambda(e_j) = 1, i = 1, \ldots, , j = 1, \ldots, n$.
   We do not have a dynamic view, like first a_i attack b and b then (if it is not out dead) tries to attack e_i. Or better still, at the same time each node launches an attack on whoever it can. So a_i attack b and b attacks e_i and the result is that a_i are alive (not being attacked) while b and e_j are all dead.
   We use the words 'there is no progression in the network' to indicate this. The network is static.

We have addressed point 4 above in our paper [3], but points 1–3, 5–7 were addressed in [9].

There are several authors who have already addressed some of these questions. See [5, 8].

Obviously, to answer the above questions we must give contents to the nodes. We can do this in two ways. We can do this in the metalevel, by putting predicates and labels on the nodes and by writing axioms about them or we can do it in the object level, giving internal structure to the atomic arguments and/or saying what they are and defining the other concepts, e.g. the notion of attack in terms of the contents.

EXAMPLE A.1 (Metalevel connects to nodes). Figure 65 is an example of a metalevel extension.

The node a is labelled by α. It attacks the node b with transmission factor ε. Node b is labelled by β. The attack arrow itself constitutes an

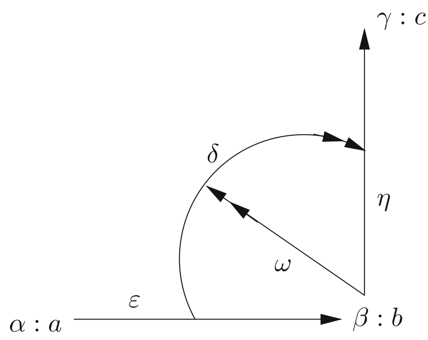


Figure 65.

attack on the attack arrow from b to c. This attack is itself attacked by node b. Each attack has its own transmission factor. We denoted attacks on arrows by double arrows.

Formally we have a set S of nodes, here

$$S = \{a, b, c\}.$$

the relation R is more complex. It has the usual arrows $\{(a, b), (b, c)\} \subseteq R$ and also the double arrows, namely, $\{((a, b), (b, c)), (b, ((a, b), (b, c)))\} \subseteq R$. We have a labelling function $\mathbf{l}$, giving values

$$\begin{aligned}
&\mathbf{l}(a) = \alpha, \mathbf{l}(b) = \beta, \mathbf{l}(c) = \gamma, \\
&\mathbf{l}((a, b)) = \varepsilon, \mathbf{l}((b, c)) = \eta, \\
&\mathbf{l}(((a, b), (b, c))) = \delta \\
&\mathbf{l}((a, ((a, b), (b, c)))) = \omega.
\end{aligned}$$

We can generalise the Caminada labelling as a function from $S \cup R$ to some values which satisfy some conditions involving the labels. We can write axioms about the labels in some logical language and these axioms will give more meaning to the argumentation network. See [3] for some details along these lines. The appropriate language and logic to do this is Labelled Deductive Systems (LDS) [11].

We shall not pursue the metalevel extensions approach in this paper except for one well known construction which will prove useful to us later.

Our matrix methods allow us to give new kind of content to the nodes and define a new mode of attack. This we now discuss.

Consider Figure 66

$$a \longrightarrow b \longrightarrow c \longrightarrow d$$

Figure 66.

As an argumentation network, any finite acyclic graph is very simple. We start with the nodes x that are not attacked, they get $\lambda(x) = 1$ and then we propagate λ along the arrows. In Figure 66 we get

$$\lambda(a) = 1, \lambda(b) = 0, \lambda(c) = 1, \lambda(d) = 0.$$

The story becomes more interesting when we try and give contents to the nodes. The main way of doing this in the literature is proof theoretical. The nodes are theories or proofs and one node x attacks another node y if it obstructs its proof by proving the opposite.

Figure 67 is such an example

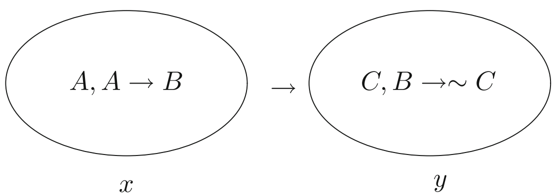


Figure 67.

x attacks y because x proves B but y proves $\sim B$.

Besnard and Hunter in [5] take the classical logic approach. The paper of Amgoud and Caminada [8] surveys other approaches where the logic may be nonmonotonic, i.e. we may have several defeasible arrows.

Anyway, all existing approaches are proof theoretical or classical semantical involving consistency.

Our matrix system can give a completely different content to an abstract argumentation network. Section 5 is an example of a series of attacks. Figure 68 is an example.

x attacks y by joining it from the right to form Figure 69

Our matrix abduction will tell y it must put 1 in the blank "?" place. In Figure 69, the result of the attack by x, we get that "?" remains undecided.

We can have new kind of attacks, as in Figure 70. z attacks y by wrapping around it.

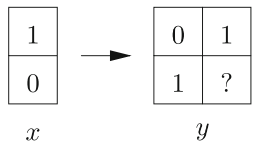


Figure 68.

0	1	1
1	?	0

Figure 69.

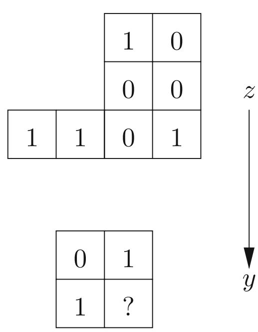


Figure 70. z wraps around y

The result is Figure 71

0	1	1	0
1	?	0	0
1	1	0	1

Figure 71.

Figure 72 is a schematic joint attack.

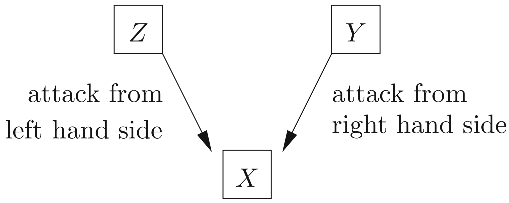


Figure 72.

We get Figure 73

Z	X	Y

Figure 73.

Our matrix abduction can deal only with rectangular matrices. So we need to be careful with joint attacks, unless we extend our algorithms to deal with general shapes, as for example in Figure 74.

We shall stop here. The full machinery can be developed in another paper.[11]

[11] One way to deal with general shapes is to expand the shape into a rectangle and regard all missing squares as having N/A (not applicable) value. This turns the matrix $\mathbb{A} = [a_{i,j}]$ into a partial function on the index (i, j). It can give $a_{i,j} = 0$ or $a_{i,j} = 1$ or $a_{i,j} =$ N/A $=$ undefined, and exactly once it can give $a_{i,j} =?$. The columns become partial functions into $\{0, 1\}$ and the ordering graph can be defined between columns as $\leq$ on all coordinates in which they are both defined. See our paper in Hebrew [1] for examples of such matrices arising from Talmudic reasoning.

0	1	
1	x=?	0
1	1	0

Figure 74.

Remark A.2 (Summary). Let us summarise what our matrix example does for argumentation theory.

Consider again the chain in Figure 66. a attacks b by joining it and forming a new matrix ab, which joins c to form $(ab)c$ and then we get $((ab)c)d$. The new matrix may still have 1 as a solution in which case the attack did not kill d. It may have undecided or 0 as a solution in which case the attack succeeded. In any case, we present a new form of attack in argumentation networks.

B. Application to Voting paradoxes

We now give a voting case study, (see [7]).

A group of 13 farmers from southern Germany rent a bus and go to London for a week's holiday. They are offered 3 extra events in London, for which they have to pay individually, in addition tot he agreed holiday package cost. These are:

T Evening at the theatre (the Globe)

D Fancy dinner at a posh London restaurant

B A tour in a boat along the Thames.

The farmers are asked to vote. We get the following result, where 1 means 'I want it', see Figure 75

The question is what to do? Where to go?

If we consider the options as packages, then packages

$$\begin{aligned} C &= \mathbf{T} \wedge \sim \mathbf{D} \wedge \sim \mathbf{B} \\ D &= \sim \mathbf{T} \wedge \sim \mathbf{D} \wedge \mathbf{B} \\ G &= \sim \mathbf{T} \wedge \mathbf{D} \wedge \sim \mathbf{B} \end{aligned}$$

each received three votes and so they draw the winning package. We cannot, however, decide between them.

If we regard the voting procedure as collecting individual votes for each of the options $\{\mathbf{T}, \mathbf{D}, \mathbf{B}\}$, then we get Figure 76.

	A	B	C	D	E	F	G
T	0	1	1	0	1	1	0
D	1	1	0	0	0	1	1
B	1	1	0	1	1	0	0
No. of farmers voting for this column	1	1	3	3	1	1	3

Figure 75.

	No. of farmers voting yes	No. of farmers voting no
T	6	7
D	6	7
B	6	7

Figure 76.

The winning combination is $\sim \mathbf{T} \wedge \sim \mathbf{D} \wedge \sim \mathbf{B}$, namely, go to no event.

This is known as the multiple voting paradox, for three issues ($\mathbf{T}, \mathbf{D}$ and $\mathbf{B}$) and 13 voters. We can generate a paradox for four issues for example, in this case we need 31 voters, see [7].

The paradox is that by majority vote we get a result that nobody wants, in our case this result is not to go anywhere! Nobody voted for $(0, 0, 0)$ as a column.

Let us see whether our matrix abduction point of view can help. Our first question is whether the method of matrix abduction is applicable to the voting problem. Do the criteria discussed in Section 1.1 apply here? The answer is yes. Each farmer wants something. Each option has ingredients to offer. If the option can satisfy what the farmer wants he will vote for it. This example is also slightly different. There are connections between the rows, maybe giving rise to constraints (see Example 3.20. There is also the question of cost, some farmers must have chosen 2 out of the three options simply because they didn't have enough money.

Let us construct the graph for this matrix. We get Figure 77.[12] The table in Figure 78 tells us what are the ingredients of the options. The way we get the ingredients, is to look at Figure 75. **T** has 1 for columns B, C, E, F. So **T** must have enough ingredients to satisfy the needs of each of B, C, E, F. These needs we get from the allocation of Figure 77. So **T** needs $\{2\alpha, \beta\}$. If we take into account how many voters voted for each column, then **T** needs $\{6\alpha, 3\beta\}$, since 3 voters voted for C. Thus we put in Figure 78 $\{6\alpha, 3\beta\}$ for **T**. Similarly we calculate $\{6\alpha, 3\gamma\}$ for **D** and $\{3\alpha, 3\beta, 3\gamma\}$ for **B**. Remember, we give **T** not the union of allocations of the relevant columns, but the minimum that we need!

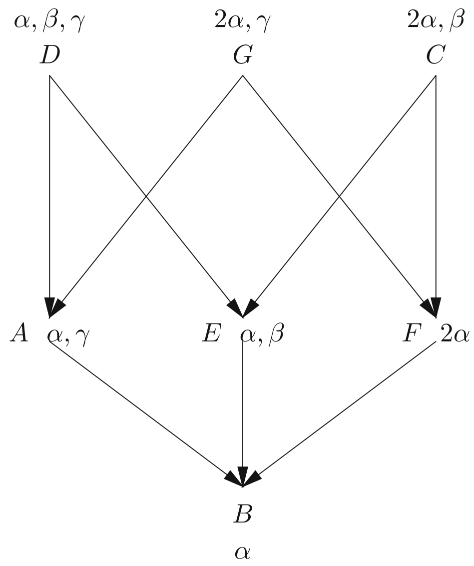


Figure 77.

[12]Note that the voting matrix has columns corresponding to all Boolean combinations of $\{\mathbf{T}, \mathbf{D}, \mathbf{B}\}$.

Thus in Figure 77 if we were to annotate with sets (not multisets) and annotated B with $\varnothing$, we would have annotated as follows:

$$A : \{\alpha\}, E : \{\beta\}, F : \{\gamma\}$$
$$D : \{\alpha, \beta\}, G : \{\alpha, \gamma\}, C : \{\beta, \gamma\}.$$

These are Boolean allocations representing the Boolean algebra with **T**,**D**,**B** as a Boolean algebra of subsets of $\{\alpha, \beta, \gamma\}$ with Truth $=\varnothing$, Falsity $=\{\alpha, \beta, \gamma\}$ and $\wedge$ becomes $\vee$ and $\vee$ becomes $\wedge$.

We get nothing new. So the two new mathematical steps we are making are assigning α to B and using multiples of α.

	Ingredients
T	$6\alpha, 3\beta$
D	$6\alpha, 3\gamma$
B	$3\alpha, 3\beta, 3\gamma$

Figure 78.

Let us now try and guess what α, β, γ can be. **D** and **B** have γ in common. Our guess is that we can take γ to be 'non-intellectual activity'.

B and **T** have β in common. Our guess would be that β is a sightseeing factor. Food you get in Germany. Theatre and the Thames are characteristic of London.

α is common to all. Our guess is that it can be cost. A boat ticket costs less (per person) than a theatre ticket or a dinner check.

So how can this help our decision about what to do?

We would go for **B** and only one of **T** and **D**. Let us check the vote for the $(\mathbf{T}, \mathbf{D})$ component only. We have four possibilities. See Figure 79

	No of votes
$\mathbf{T} \wedge \mathbf{D}$	2
$\mathbf{T} \wedge \sim \mathbf{D}$	4
$\sim \mathbf{T} \wedge \mathbf{D}$	4
$\sim \mathbf{T} \wedge \sim \mathbf{D}$	3

Figure 79.

We get a tie between $\mathbf{T} \wedge \sim \mathbf{D}$ and $\sim \mathbf{T} \wedge \mathbf{D}$. Which one to choose? Looking at the ingredients, they are symmetrical, $\{6\alpha, 3\beta\}$ compared to $\{6\alpha, 3\gamma\}$. It is really a vote between β and γ. We must ask the farmers.

Note that the graph is symmetrical. So we could have assigned the α, β, γ differently.

There are two more possibilities, assign in Figure 77 the value 2α to E or assign the value 2α to A.

We get Figures 80 and 81 respectively.

The following table, Figure 82, summarises the possible ingredients for $\mathbf{T}, \mathbf{D}, \mathbf{B}$ according to which figure we use. It extends Figure 78.

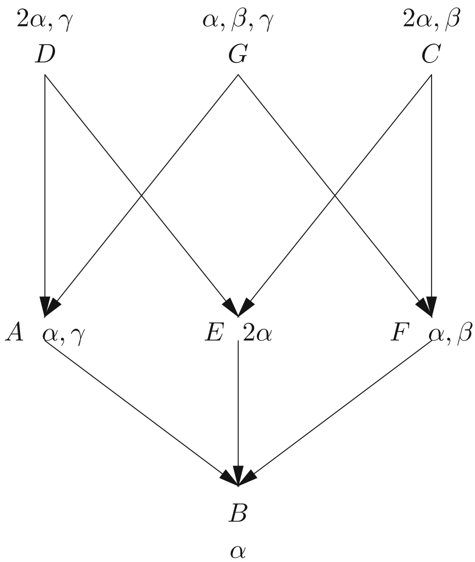


Figure 80.

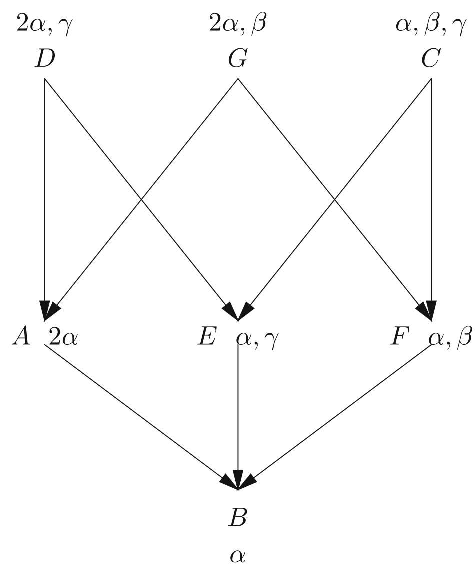


Figure 81.

	Ingredients Figure 78	Ingredients Figure 80	Ingredients Figure 81
T	$6\alpha, 3\beta$	$6\alpha, 3\beta$	$3\alpha, 3\beta, 3\gamma$
D	$6\alpha, 3\gamma$	$3\alpha, 3\beta, 3\gamma$	$6\alpha, 3\beta$
B	$3\alpha, 3\beta, 3\gamma$	$6\alpha, 3\gamma$	$6\alpha, 3\gamma$

Figure 82.

We see that the graphs and the tables, including Figure 75 are completely symmetrical. So the conclusion is that the farmers should do any two events. Only a combination of two events can have enough ingredients for all columns. Namley the vote concludes with

$$\mathbf{V} = (\mathbf{T} \wedge \mathbf{D} \wedge \sim \mathbf{B} \bigvee \mathbf{T} \wedge \sim \mathbf{D} \wedge \mathbf{B} \bigvee \sim \mathbf{T} \wedge \mathbf{D} \wedge \mathbf{B}).$$

Note that this matrix abduction approach is pretty revolutionary. We follow neither the drawing contenders for the maximal vote (of 3, namely, C, D and G), i.e. we do not take exactly one of $\{\mathbf{T}, \mathbf{D}, \mathbf{G}\}$, i.e. $\mathbf{T} \wedge \sim \mathbf{D} \wedge \sim \mathbf{B}$ or $\sim \mathbf{T} \wedge \mathbf{D} \wedge \sim \mathbf{B}$ or $\sim \mathbf{T} \wedge \sim \mathbf{D} \wedge \mathbf{G}$, nor do we follow the result of the component vote of Figure 76, namely $\sim \mathbf{T} \wedge \sim \mathbf{D} \wedge \sim \mathbf{B}$.

We follow a compromise, as suggested by the matrix abduction.

This is already more than the voting procedures can give us. Also we know what to ask the farmers in order to decide the matter!

Let us now check what happens in the case of the four issues, 31 voters paradox, see [7]. This paradox has issues $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$. $\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} \wedge \mathbf{d}$ is a winning combination, and yet, calculated coordinatewise we get complete reversal, $\sim \mathbf{a} \wedge \sim \mathbf{b} \wedge \sim \mathbf{c} \wedge \sim \mathbf{d}$, which is an option for which nobody voted!

Figure 83 gives the voting table.

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{10}	X_{13}	X_{14}	X_{15}
a	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
b	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0
c	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
d	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
No of Voters for this column	5	1	1	1	1	1	1	4	1	1	1	4	1	4	4

Figure 83.

If we count the numbers of voters who voted for a column containing 1 for **a** (i.e. who wanted **a**) we get 15, as opposed to 16 voters who wanted $\sim \mathbf{a}$. Similarly we have for **b**,**c** and **d**. This is why we have a paradox.

If we follow the majority package vote we have to go for $\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} \wedge \mathbf{d}$ because 5 voters went for it, the biggest number of voters. This makes 26 other voters unhappy. If we go for the issue by issue result, then $\sim \mathbf{a} \wedge \sim \mathbf{b} \wedge \sim \mathbf{c} \wedge \sim \mathbf{d}$ wins, since each issue got voted 0 by 17 against 16 who voted 1. But then this is not good either since everyone voted for something. Nobody wanted the 'nothing' option.

So we propose the matrix method. Figure 84 draws the graph of the columns of Figure 83, with an indicated allocation of ingredients.

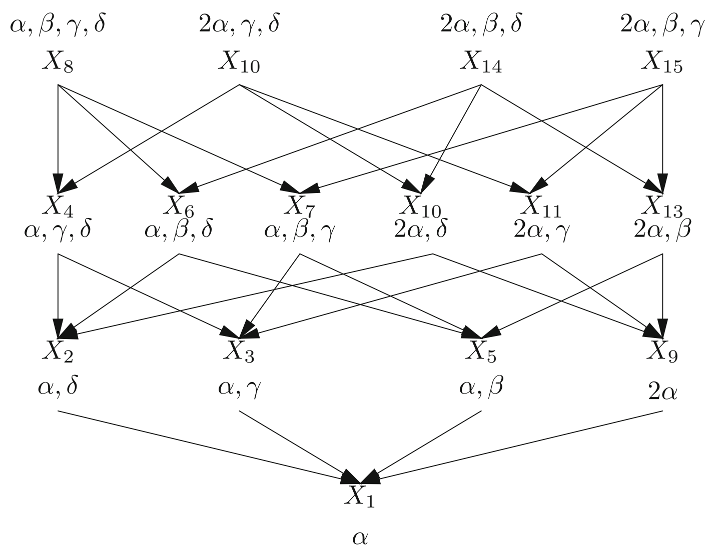


Figure 84.

If we collect the ingredients from Figure 84 using the table of Figure 83 we get Figure 85. Note that we make allowance for the number of voters.

To satisfy the needs of all voters we need at least two of $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$. We do not take more because each individual $\mathbf{a}, \mathbf{b}, \mathbf{c}$, or **d** was voted 0 by the majority. The winning single package **a** does not have enough α to satisfy the X_8, X_{10}, X_{14} and X_{15} voters.

Thus we need two from $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$. Note that we relied here on the allocation of $\alpha, \beta, \gamma, \delta$ on the graph. There are three other ways of doing this allocation, giving α to X_1 but 2α either to X_2, X_3 or X_5.

	ingredients
a	$5\alpha, 4\beta, 4\gamma, 4\delta$
b	$8\alpha, 4\gamma, 4\delta$
c	$8\alpha, 4\beta, 4\delta$
d	$8\alpha, 4\beta, 4\gamma$

Figure 85.

Since both the graph of Figure 84 and the table of Figure 83 are completely symmetrical in the swapping of X_2, X_3, X_5 and X_9, our conclusion that we need two of $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ to make everyone happy does stand!

Thus our recommendation for the vote of Figure 83 is

$$\begin{aligned}\mathbf{F} \;=\; & (\mathbf{a} \wedge \mathbf{b} \wedge \sim \mathbf{c} \wedge \sim \mathbf{d} \bigvee \mathbf{a} \wedge \sim \mathbf{b} \wedge \mathbf{c} \wedge \sim \mathbf{d} \\ & \bigvee \mathbf{a} \wedge \sim \mathbf{b} \wedge \sim \mathbf{c} \wedge \mathbf{d} \bigvee \sim \mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} \wedge \sim \mathbf{d} \\ & \bigvee \sim \mathbf{a} \wedge \mathbf{b} \wedge \sim \mathbf{c} \wedge \mathbf{d} \bigvee \sim \mathbf{a} \wedge \sim \mathbf{b} \wedge \mathbf{c} \wedge \mathbf{d})\end{aligned}$$

Again note that this is contrary to both the winning package vote ($\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} \wedge \mathbf{d}$ which is winning by 5 voters) and the winning issue by issue vote ($\sim \mathbf{a} \wedge \sim \mathbf{b} \wedge \sim \mathbf{c} \wedge \sim \mathbf{d}$ where each $\sim \mathbf{a}, \sim \mathbf{b}, \sim \mathbf{c}, \sim \mathbf{d}$ wins by 17 against 16 votes).

Again a revolutionary compromise (suggested by the matrix method) between the two extreme components of the paradox.

Remark B.1. We stress that we are not necessarily offering here the matrix method as new voting procedure. The matrix method relies on ingredients, not on numerical aggregation of votes. It can help when the voting aggregation is parasdoxical.

We shall examine the matrix method as a voting procedure in a future paper.

To show the coherence and solidity of our matrix approach, let us increase the number of voters by 2, from 31 to 33, and let them vote for X_1. Figure 83 will change in the bottom of column X_1 from 5 to 7.

X_1 will continue to be the winning package but now the paradox will disappear. Each of $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and $\mathbf{d}$ will get coordinatewise 17 votes as opposed respectively 16 to $\sim \mathbf{a}, \sim \mathbf{b}, \sim \mathbf{c}$ and $\sim \mathbf{d}$.

The graph in Figure 84 remains the same, with its allocations of $\alpha, \beta, \gamma, \delta$. What will change is Table 85, which describes the amount of ingredients $\alpha, \beta, \gamma, \delta$ to each row. Each of $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ will get 2 more α. So $\mathbf{a}$ now has 7α.

Still not enough to satisfy the needs of the voters of X_{10}, X_{14} and X_{15} which require 8α.

So although there is no paradox, the majority of X_1 is still not strong enough to go alone. If we add one more voter who votes for X_1, (bringing the number of voters to 34) then **a** will get 8α, now enough to satisfy on its own all voters.

We see here that our method is different but still reasonable and paradox free.

As we said, a detailed study of this approach will be pursued in another paper.

C. Application to Paradoxes of judgement aggregation

We begin with the *doctrinal paradox*, identified by Kornhauser and Sager [20, 21]. The paradox arises when majority voting can lead a group of rational agents to endorse an irrational collective judgement. Consider the question of liability following a breach of contract. Three judges have to decide whether

a= there was a binding contract

and

b= there was a breach of that contract.

We get liability only when both **a** and **b** are upheld.

The table in Figure 86 describes the situation.

	Judge A	Judge B	Judge C	majority vote
a	1	0	1	1
b	0	1	1	1
liability $\mathbf{a} \wedge \mathbf{b}$	0	0	1	$x = 0$

Figure 86.

Judges A and C think the evidence for **a** is convincing but not so for **b**. Judges B and C think the evidence for **b** is convincing but not so for **a**. Thus each judge individually would express judgement as in row $\mathbf{a} \wedge \mathbf{b}$. Therefore two judges will give verdicts of no liability and only one, Judge C, will give a verdict of liability. Going by majority of verdicts — the final verdict is $x = 0$.

On the other hand, if we were to take majority judgement first on **a** and **b** individually, then we get that both **a** and **b** get 1 having a majority of two judges,and so x must be 1 and not 0.

This is the paradox.

We see this paradox as a special matrix abduction problem.

We can now formulate the *general matrix aggregation problem.*

Definition C.1 (Matrix aggregation problem).

1. Let $V = (x_1, \ldots, x_n)$ be a vector of numbers in $\{0, 1\}$. An *aggregation function* $\mathbf{g}$ is a function giving a value $\mathbf{g}(V) \in \{0, 1\}$, for any such vector. For example

$$\begin{array}{l} \mathbf{g}_{\text{majority}}(V) = 1 \text{ iff } \sum_{i=1}^{n} x_i > \frac{1}{2} \\ \mathbf{g}_{\wedge}(V) = 1 \text{ iff } x_i = 1 \text{ for all } i \\ \mathbf{g}_{\vee}(V) = 1 \text{ iff } x_i = 1 \text{ for some } i \end{array}$$

2. Let $\mathbf{g}_{\text{row}}$ and $\mathbf{g}_{\text{column}}$ be two aggregation functions.

A matrix $\mathbb{A}$ with $m+1$ rows and $n+1$ columns is a *matrix aggregation problem* if it has the form described in Figure 87.

	A_1	$\ldots$	A_n	Row aggregation
$\mathbf{a}_1$				$\mathbf{g}_{\text{row}}(\mathbf{a}_1)$
$\vdots$				
$\mathbf{a}_m$				$\mathbf{g}_{\text{row}}(\mathbf{a}_m)$
Column aggregation	$\mathbf{g}_{\text{column}}(A_1)$		$\mathbf{g}_{\text{column}}(A_n)$	$x =?$

Figure 87.

The row aggregation column gives the aggregated value for each row. The column aggregation row gives the aggregated value for each column. We get a matrix abduction problem if we ask what should $x =?$ be. Do we aggregate the column above the $x =?$ square, i.e. let $x = \mathbf{g}_{\text{column}}(\mathbf{g}_{\text{row}}(\mathbf{a}_1), \ldots, \mathbf{g}_{\text{row}}(\mathbf{a}_m))$ or do we aggregate the row to the left of the $x =?$ square, i.e. let $x = \mathbf{g}_{\text{row}}(\mathbf{g}_{\text{column}}(A_1), \ldots, \mathbf{g}_{\text{column}}(A_n))$ or do we do some matrix abduction algorithm $\mathcal{A}$ on the matrix and get a value for x?

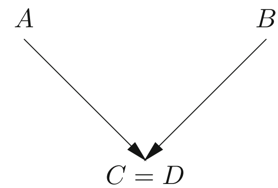
Figure 88 describes the situation of Figure 86 seen as a matrix aggregation problem.

If we apply our graph technique to this figure, we get the graphs in Figure 89.

	A	B	C	D=$\mathbf{g}_{\text{majority}}$
$\mathbf{a}$	1	0	1	1
$\mathbf{b}$	0	1	1	1
$\mathbf{c} = \mathbf{g}_\wedge$	0	0	1	$x = ?$

Figure 88.

Case $x = 1$



Case $x = 0$

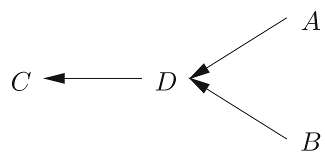


Figure 89.

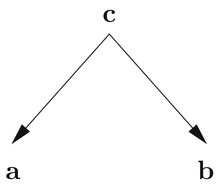
Clearly by our criteria of Section 3, $x = 1$ wins. Of course we need to formulate new criteria suitable for the voting and the aggregation application area. We have here different kinds of matrices. So we need to look at some examples and match the intuitive ideas embedded in the examples with criteria on graphs.

There is also symmetry between rows and columns in this problem. In general we have two aggregation functions without any special conditions on them. So we must also consider the graphs arising from the rows. This we show in Figure 90 for our specific example.

Again, according to our criteria of Section 3, case $x = 1$ wins.

The interested reader can look up a recent penetrating analysis of the paradox in [16]. The paper uses probabilistic methods, looking at the reliability of the judges involved and aggregating accordingly. If we adopt the reliability idea into our matrix we get the matrix in Figure 91

Case $x = 1$



Case $x = 0$

$\mathbf{a}\ \bullet \qquad \mathbf{b}\ \bullet \qquad \mathbf{c}\ \bullet$

Figure 90.

	Judge A	Reliability of A	Judge B	Reliability of B	Judge C	Reliability of C	Majority vote
$\mathbf{a}$	1	$r(A, \mathbf{a})$	0	$r(B, \mathbf{a})$	1	$r(C, \mathbf{a})$	
$\mathbf{b}$	0	$r(A, \mathbf{b})$	1	$r(B, \mathbf{b})$	1	$r(C, \mathbf{b})$	
$\mathbf{a} \wedge \mathbf{b}$	0		0		1		

Figure 91.

The reliabilities $r(A, \mathbf{a}), r(A, \mathbf{b})$ are the numbers in $\{0, 1\}$ telling us whether Judge A is reliable on issues **a** and **b**. Similarly, $r(B, \mathbf{a}), r(B, \mathbf{b})$, $r(C, \mathbf{a})$ and $r(C, \mathbf{b})$.

It was suggested to us by S. Hartmann that we process the columns first in the matrix and then aggregate. We use the following processing formula.

- x with reliability 1 is processes as $x' = x$
- x with reliability 0 is processed as $x' = 1 - x$.

So the formula is

$$x' = x \cdot r(x) + (1 - x)(1(r(x))$$

So in the matrix of Figure 91 we first process the pairs of columns and get the matrix of Figure 92.

Note that such preprocessing is done in the Talmud. We discuss this in our Hebrew paper [1, Part 2, Section A].

	new Judge A	new Judge B	new Judge C	vote
$\mathbf{a}$	$r(A, \mathbf{a})$	$1 - r(B, \mathbf{a})$	$r(C, \mathbf{a})$	
$\mathbf{b}$	$1 - r(A, \mathbf{b})$	$r(B, \mathbf{a})$	$r(C, \mathbf{b})$	
$\mathbf{a} \wedge \mathbf{b}$				

Figure 92.

Summary

The general theory of the matrix aggregation problem needs to be developed. We see however already at this stage that we have a clear mathematical formulation of the problem and we have a machinery to offer a solution. This is good news for the judgment aggregation community.

We hope to address this problem in a subsequent paper.

We note that we have not explained away the aggregation paradox but offered a possible third computation to bail us out of the aggregation problem.

D. Learning, Labelling and Finite Models

We continue to examine the example of monadic predicate logic introduced in Figure 8. We have a finite model with predicates $A_1, \ldots, A_m$ and elements $d_1, \ldots, d_n$. Let us assume that the definite matrix $\mathbb{A} = [a_{i,j}], i = 1, \ldots, m$ and $j = 1, \ldots, n$ describes the model. That is we have for all i, j

$$a_{i,j} = 1 \text{ iff } A_i(d_j) \text{ is true.}$$

Our question is what can we learn from this data?

To explain how we can make use of our matrix realisation method, let us take an example we have already analysed. Consider the matrix of Figure 60 for the case $x = 1$.

We consider this matrix as a matrix of a model with elements

$$\begin{aligned} d_1 &= N \\ d_2 &= A \\ d_3 &= P \\ d_4 &= Y \\ d_5 &= H \\ d_6 &= G \\ d_7 &= K \end{aligned}$$

and predicates

$$\begin{aligned} A_1 &= \mathbf{m} \\ A_2 &= \mathbf{n} \\ A_3 &= \mathbf{b} \\ A_4 &= \mathbf{w} \end{aligned}$$

The graph for this matrix is in Figure 61.

Let us insist on a realisation for this graph using sets, not multisets. This means we do not allow multiples of α.

Therefore we would get the following realisation $\mathbf{f}$ in Figure 93.

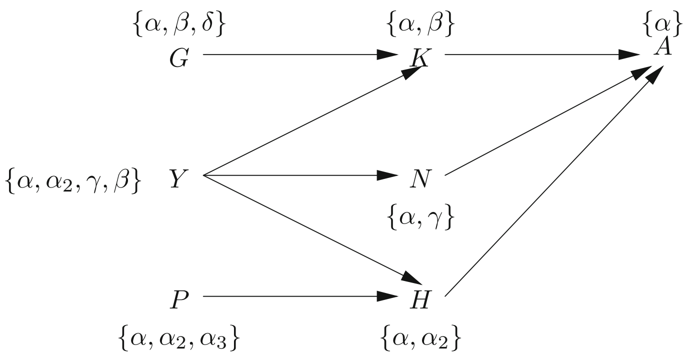


Figure 93.

Figure 94 gives the corresponding matrix realisation.[13]

$$A_i(d_j) = \text{ True iff } \mathbf{f}(A_i) \supseteq \mathbf{f}(d_j).$$

Note that instead of a model with 7 elements, we found a model of 6 elements (a saving in the number of elements) which contains all the information. In this new smaller model we have that A_i, d_j are all predicates and the meaning of $A_i(d_j)$ is $\forall x(d_j(x) \to A_i(x))$.

Independently of whether we save on the number of points, this method suggests a translation of the monadic first order theory into itself.

---

[13]This is hardly surprising. Let $\mathbb{A} = [a_{i,j}]$ be a definite matrix with m rows and n columns, $m \leq n$. Let $\mathbf{a}_1, \ldots, \mathbf{a}_m$ be the rows and $C_1, \ldots, C_n$ be the columns. Then each column C is a subset $\mathbf{C}$ of the set $M = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$ of all rows. We have $\mathbf{a}_i \in \mathbf{C}_j$ iff $a_{i,j} = 1$. Thus the columns are elements of the Boolean algebra of the powerset 2^M. The graph of the columns is a subgraph of the lattice of the algebra 2^M with M as the smallest element and $\varnothing$ as the top element. Since we give α to the bottom element of the graph, this observation implies that we can get a set realisation to the graph with $m+1$ elements. In practice many graphs can manage with less, but some might need more, since we need to realise the cancellation of edges and some possible restrictions as well.

$$\begin{array}{l} \mathbf{m} : \{\alpha, \alpha_2, \alpha_3\} \\ \mathbf{h} : \{\alpha, \gamma\} \\ \mathbf{b} : \{\alpha, \alpha_2, \beta, \gamma\} \\ \mathbf{w} : \{\alpha, \beta, \gamma\} \\ N : \{\alpha, \gamma\} \\ A : \{\alpha\} \\ P : \{\alpha, \alpha_2, \alpha_3\} \\ Y : \{\alpha, \alpha_2, \beta, \gamma\} \\ H : \{\alpha, \alpha_2\} \\ G : \{\alpha, \beta, \delta\} \\ K : \{\alpha, \beta\} \end{array}$$

Figure 94.

We prepare the ground for the translation by observing that all monadic models for m monadic predicates $\{A_1, \ldots, A_n\}$ can be reduced equivalently to models with at most 2^m elements. This holds since each element d in the domain has a type $\bigwedge_i \pm A_i(d)$, and there are at most 2^m such types. So any formula with quantifiers can be rewritten to a formula without quantifiers. If we have 2^m constants in the language $\mathbf{d}_1, \ldots, \mathbf{d}_{2^m}$. We write $\forall x \varphi(x) \equiv \bigwedge_{i=1}^{2^m} \varphi(\mathbf{d}_i)$ and $\exists x \varphi(x) \equiv \bigvee_{i=1}^{2^m} \varphi(\mathbf{d}_i)$.

So any ϕ becomes ϕ^* by eliminating the quantifiers in this manner. So, for example

$$\phi = \forall x \exists y (A(x) \to B(y))$$

becomes

$$\phi^* = \bigwedge_i \bigvee_j (A(\mathbf{d}_i) \to B(\mathbf{d}_j))$$

We can translate now any ϕ into ϕ^{**} as follows:

Let $D_1, \ldots, D_{2^m}$ be additional predicates. Translate any $A_i(\mathbf{d}_j)$ into $\forall x (D_j(x) \to A_i(x))$. Now given any closed formula ϕ translate first into ϕ^* by using $\mathbf{d}_j$. ϕ^* will have no quantifiers. Now replace in ϕ^* any $A_i(\mathbf{d}_j)$ as above and get ϕ^{**}.

We have

LEMMA D.1. *ϕ has a monadic model iff ϕ^{**} has a monadic model.*

PROOF. ϕ has a monadic model iff ϕ has a monadic model $\mathbf{M}$ with 2^m elements iff ϕ^* has this same monadic model. We now construct the matrix $\mathbb{A}$

for the model **M**, and construct a set realisation of it using a set of labels $\mathbb{L}$. Then ϕ^* holds in **M** iff ϕ^{**} holds in $\mathbb{L}$. ■

REMARK D.2. We are aware that we can have a similar translation by regarding any element d_j of the domain as a predicate D_j with an extension of exactly the element d_j. Thus $A_i(d_j)$ becomes

$$\forall x(D_j(x) \to A_i(x)).$$

We need to add the axioms

1. $\exists x D_j(x), j = 1, \ldots, n$
2. $\sim \exists x[D_j(x) \wedge D_i(x)], i \neq j$

We have ϕ has a model of n elements iff $(1) \wedge (2) \wedge \phi^{**}$ has (the same) model.

This translation does not decrease the number of elements because (1) and (2) ensure the same number of elements is used.

Furthermore, we can use the Lemma for the previous translation and a theorem Prover to find a minimal model for ϕ^{**}. This will give us the minimal number of labels for the columns withot using the graphs. We summarise

LEMMA D.3. *Let $\mathbb{A}$ be a definite $m \times n$ matrix. Consider it as a model with rows as predicates and columns as elements as in Figure 8.*

Let φ be the following formula

$$\varphi_{\mathbb{A}} = \bigwedge_{a_{i,j}=1} \forall x(D_j(x) \to A_i(x)).$$

Let a theorem prover find a minimal model **M** *for $\varphi_{\mathbb{A}}$.*

Then the sets

$$D_j = \{a \in \mathbf{M} \mid D_j(a) \text{ is true}\}$$

form a set realisation for $\mathbb{A}$.

PROOF. Follows from our previous constructions and lemmas. ■

REMARK D.4. How do we use a theorem prover to find a multiset realisation?

Let $\mathbb{A}$ be given. Write $\varphi_{\mathbb{A}}$. Add constants to the language $\alpha_1, \alpha_2, \ldots, \alpha_n$. We do not need more than n since we know we can find a realisation with n elements. We think of α_k as $k\alpha$. This means that we add the axiom

$$\alpha = \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n-1} D_i(\alpha_{j+1}) \to D_i(\alpha_j))$$

This means that if $(j+1)\alpha$ labels a node then so does $j\alpha$.

We use a theorem prover to find a model for $\alpha \wedge \varphi$ and minimise the number of elements in the model which are not α_j.

E. Applications to Access Control Reasoning

Matrix abduction can be also employed in reasoning with incomplete information about access control policies.

Access control consist of determining whether a principal (machine, user, program ...) which issues a request to access a resource should be trusted on its request, i.e. if it is authorized.

A classical way of representing access control policies is the employment of an "Access Control Matrix" which characterizes the rights of each principal with respect to every object in the systems.

For instance, suppose we have an access control policy expressed in Figure 95, where in cell (i, j) we place 1 if principal i can read $file_j$.

	$file_1$	$file_2$	$file_3$
A	1	1	1
B	0	1	0
C	1	?	0

Figure 95. Access Control Matrix

Where the '?' means that the specification of the access control policy is unknown or incomplete. The question is: what to do? Should a reference monitor deny the access to read $file_2$ to C or not?

There are many ways to reply to the raised questions, here we report two simple examples

- The reference monitor can query his knowledge about the principals A, B and C. For instance, if C has more power than A (maybe C is *root* and A a *user*), the reference monitor can than derive that C has the right to read $file_2$.
- The reference monitor can rely on some knowledge about how the files are organized, so if $file_1$ has a higher protection level the reference monitor may assume that $file_2$ can be read by C.

Another possibility is to employ the methods of Matrix Abduction proposed in this article. We leave for future research the formalization of an ordering between components of access control matrices in order to be able to craft access control policies that can be employed in reasoning without complete knowledge of the domain. Recently, in [4] abduction in access control policies has been also applied to compute a specification of missing credentials in decentralized authorization languages. We believe that matrix abduction can provide a practical tool to craft new access control models, as future work we plan to extend the logical framework presented in [6, 2] with abductive reasoning methods.

Acknowledgements. We are grateful to S. Hartmann, D. Makinson and O. Rodrigues for helpful comments.

References

[1] Abraham, M., D. M. Gabbay, and U. Schild, Paper 340 Kal-Vachomer in Hebrew, BDD Journal, Bar Ilan University, 112 pages.

[2] Barker, S., G. Boella, D. M. Gabbay, and V. Genovese, 'A Meta-model of Access Control in a Fibred Security Language', *Studia Logica* 92: 437–476, 2009.

[3] Barringer, H., D. M. Gabbay, and J. Woods, 'Temporal dynamics of argumentation networks', in D. Hutter and W. Stephan (eds.), *Mechanising Mathematical Reasoning*, LNCS 2605, Springer, 2005, pp. 59–98.

[4] Becker, M. Y., J. F. Mackay, and B. Dillaway, 'Abductive Authorization Credential Gathering. http://research.microsoft.com/pubs/80508/becker2009ieee-policy-submission.pdf

[5] Besnard, P., and A. B. Hunter, *Elements of Argumentation*, MIT Press, 2008, 300 pages.

[6] Boella, G., D. M. Gabbay, V. Genovese, and L. van der Torre, 'Fibred Security Language', *Studia Logica* 92: 395–436, 2009.

[7] Brams, S. J., D. M. Kilgour, and W. S. Zwicker, 'The paradox of multiple elections', *Social Choice and Welfare* 15: 211–236, 1998.

[8] Caminada, M. W. A., and L. Amgoud, 'On the evaluation of argumentation formalisms', *Artificial Intelligence* 171 (5–6): 286–310, 2007.

[9] Dung, Phan Minh, 'An argumentation theoretic foundation for logic programming, *Journal of Logic Programming* 22 (2): 151–171, 1995.

[10] Fisch, M., *Rational Rabbis: Science and Talmudic Culture*, Indiana University Press, Bloomignton, 1997.

[11] Gabbay, D. M., *Labelled Deductive Systems*, OUP, 1996.

[12] Gabbay, D. M., and A. Garcez, 'Logical modes of attack in argumentation networks'. To appear in *Studia Logica*.

[13] Gabbay, D. M., and J. Woods, *The Reach of Abduction*, Elsevier, 2005.

[14] Gabbay, D. M., and J. Woods, 'Resource origins of non-monotonicity', *Studia Logica* 88: 85–112, 2008.

[15] Grabenhorst, T. K., *Das Argumentum A Fortiori*, Peter Lang, 1990.

[16] Hartmann, S., G. Pigozzi, and J. Sprenger, 'Reliable methods of judgement aggregation', Draft, 23 February 2009.

[17] Hasan, A., *Analogical reasoning in Islamic jurisprudence*, 1986, Republished Adam Publishers, 2007, 486 pages.

[18] Jacobs, L., *Studies in Talmudic Logic and Methodology*, London, Vallentine-Mitchell, 1061. Republished paperback, 2006.

[19] Kamali, M. H., *Principles of Islamic Jurisprudence*, Islamic Text Society, 3rd Revised Eition, 2002, 550 pages.

[20] Kornhauser, L., and L. Sager, 'Unpacking the court', *Yale Law Journal* 96: 82–117, 1986.

[21] Kornhauser, L., and L. Sager, 'The one and the many: adjudication in collegial courts', *California Law Review* 81: 1–51, 1993.

[22] Kunst, A., 'An overlooked type of inference', *Bulletin of the School of Oriental and African Studies*, X, part 4, 1942, pp. 976–991.

[23] Neusner, J., 'The making of the mind of Judaism; the formative age', *Brown Judaic Studies*, vol. 133, Scholars press, Atlanta, 1987.

[24] Pigozzi, G., and L. van der Torre, 'Premise independence in judgement aggregation', Dagstuhl Seminar 07531, 2007.

[25] Prakken, H., and G. Sartor, 'Argument based extended logic programming with defeasible priorities', *Journal of Applied Non-classical Logics* 7: 25–75, 1997.

[26] Schild, U. J., 'Criminal Sentencing and Intelligent Decision Support', *Artificial Intelligence and Law* 6: 2–4, 1998.

[27] Schwarz, A., *Der Hermeneutische Syllogismus in der Talmudischen*, Ltitteratur, Karlsruhe, 1901.

[28] Stebbing, V. L. S., *A Modern Introduction to Logic*, London, 1945.

M. Abraham
U. Schild
Bar-Ilan University, Israel
mdabraham@bezeqint.net
schild@cs.biu.ac.il

Dov M. Gabbay
Department of Computer Science
King's College, London, UK
and
Bar-Ilan University, Israel
dov.gabbay@dcs.kcl.ac.uk