Belief Revision using Table Transformation

George Flouris and Dimitris Plexousakis
{fgeo,dp}@csi.forth.gr
Institute of Computer Science
Foundation for Research and Technology - Hellas
Science and Technology Park of Crete P.O.Box 1385
GR 711 10 Heraklion, Crete, Greece
Tel: +30(81)39 16 00
Fax: +30(81)39 16 01

TR-290 July 2001,
ICS-FORTH

Abstract. A most crucial problem in knowledge representation is the revision of knowledge when new, possibly contradictory, information is obtained (belief revision). In this report, we address this problem when the knowledge base is a set of expressions in propositional logic. We introduce a new representation of propositional expressions using 2-dimensional matrices and provide the theoretical underpinnings of this representation. We prove several propositions regarding matrices and their similarity with classical logic and show why this representation is more expressive than classical propositional logic. We exploit this increased expressiveness to devise a solution to the problem of belief revision in propositional knowledge bases and describe a simple method to perform revisions and contractions under this new notion. Finally, we compare our method with proposed algorithms from the literature.

1. Introduction

The problem of revising beliefs is the problem of adapting a given piece of knowledge to accommodate a new piece of information regarding the world being modeled. It is a most crucial problem in knowledge bases (KB), as we generally have to deal with dynamic worlds, where changes are quite frequent. Moreover, we must base our knowledge on information that could be incomplete, or even faulty. Therefore, new updates could contradict the given knowledge, and we should be able to track and remove the contradictions created by each update.

The updating methods of knowledge are by no means obvious, even when we are concerned with the intuitive processes only. Let us consider the simple example of knowing a fact A, as well as the proposition A→B. One obvious implication of the above is the fact B (by modus ponens), which could be inserted into our KB as a new fact. Let us now consider the negation of B (¬B) entering into the base as a new piece of knowledge (ie an update). This contradicts our assumption that B is true, so we will have to give up some (or all) of our previous beliefs or we would result in an inconsistent KB. Alternatively, we could reject the update as non-valid. Even in this trivial example, it is not clear which approach should be taken. Extra-logical factors should be taken into account, like the source and reliability of each piece of information or some kind of bias towards or against updates.

This and similar problems have been addressed by several scientists, including philosophers, computer scientists, logicians and others, in an effort to provide us with an intuitively correct method of belief updating. An excellent introductory survey of such efforts by Gärdenfors may be found in [8]. This report concentrates on the description of a new method of representing propositional expressions and the effects of this representation on the problem of belief revision. For a short, but partial description of our technique, see [7].

2. Previous Work

One of the first attempts to solve the belief revision problem is due to Fagin, Ullman and Vardi [6]. However, as proven in the same paper, their method forces us to completely abandon the old knowledge whenever we update the base with an inconsistent piece of information. This may be unacceptable in most applications.

Dalal in [2, 3] proposed another, more promising method of revising beliefs. He provided a specific algorithm for updating propositional databases which was based on four basic principles, namely:

1) *Irrelevance of Syntax*: Logically equivalent databases (and updates), should give logically equivalent update results.
2) *Primacy of New Information*: New knowledge is always assumed more reliable than old knowledge.
3) *Persistence of Prior Knowledge*: As much as possible of the old data should be retained in the database; we should only retract the minimum knowledge needed to keep the database consistent.
4) *Fairness*: All things being equal, when we have more than one choice for the result of the update, none of the choices should be arbitrarily chosen, in order to preserve determinism.

Dalal also formalized the notion of *minimal change* (third principle) and proved ([3]) that his method generally retained more knowledge in cases of inconsistent updates than any of the up-to-then proposed algorithms.

An alternative approach was presented by Alchourron, Gärdenfors and Makinson in a series of papers ([1, 8, 14]). Their idea was to recede from the search of any specific algorithm and attempt to formalize the notion of update. As a result, a set of widely accepted properties of any belief revision algorithm was introduced, in the form of postulates which were in fact a set of logical propositions (named *AGM postulates* after the initials of the authors). By specifying those postulates, a series of important theoretical results could be proved.

The AGM postulates inspired a series of other works, like [11] by Katsuno and Mendelzon who proposed a different theoretical foundation of update functions by reformulating the AGM postulates in terms of formulas. They also provided an elegant representation based on orderings of belief sets. Other works investigated generalizations of the postulates into the knowledge level ([15]), and there were a number of alternative propositions ([5, 10]). Most of the above works are concerned with the theoretical foundation of the belief revision problem, so they are mostly of theoretical interest. Williams in [16, 17] followed a more practical approach by providing implementations of algorithms based on the AGM paradigm.

3. Properties of Belief Revision

In order to address the problem of belief revision we must first examine some of its properties. One primary consideration is the concurrence of the results with human intuition. This consideration is formally expressed by the principles of Dalal and the AGM postulates. However, it is not absolutely clear how humans revise their beliefs, despite the efforts by psychologists in the area.

One example of disagreement is the representation of knowledge in the human brain. There are two general types of theories concerning this representation: *foundation* and *coherence* theories ([9]). Foundational theorists argue that knowledge should consist of a set of reasons. According to this theory, knowledge has the form of a pyramid, where only some beliefs (called *foundational*, or *reasons*) can stand by themselves; the rest being derived by the most basic (foundational) beliefs. On the other hand, coherence theorists believe that each piece of knowledge has an independent standing and needs no justification, as long as it does not contradict with other beliefs. Surprisingly, experiments have shown that the human brain actually uses the coherence paradigm ([9]). However, there has been considerable debate on the explanation of the experiments' results. The experiments showed that people tend to ignore causality relationships once a belief has been accepted as a fact, even if this belief has been accepted solely by deduction from other beliefs. The followers of the coherence approach argue that what actually happens is that humans do not actually *ignore* the causality relationships, but *forget* them. The subject will be very willing to reject any beliefs whose logical support no longer exists, but only if he is reminded of this fact.

The approach (foundational or coherence) chosen greatly influences the algorithms considered. Foundational KBs need to store the reasons for beliefs, along with the beliefs themselves. KBs based on the coherence paradigm need to store the set of beliefs only. Reasons should be taken into account when revising a KB only if the foundational approach is selected. The coherence paradigm practically considers all beliefs equal and ignores any causality relationships.

The set of beliefs of any KB includes the derived beliefs. It is generally the case that the derived beliefs are too many, or even infinitely many. This is a serious drawback, so it has been proposed that instead of the whole set of beliefs (*belief set*), a small number of propositions could be stored (*belief base*), enough to reproduce the whole set via deduction. Belief sets are useful theoretic constructions, but cannot be directly used in implementations, due to their size. Belief bases are more useful when it comes to applications. The use of belief bases does not necessarily force us to use the foundational paradigm; the causality relationships possibly implied by the use of the theorem prover that performs the deduction could or could not be used, depending on the approach.

The use of belief bases gives rise to another problem which is the selection of the belief base. In general, a given belief set can be derived from several bases. Different selections of bases may give different reasons (deductions) and this difference is crucial under the foundational approach, but irrelevant under the coherence approach.

Another important consideration is the problem of *iterated revisions*. All the algorithms described so far are concerned with just one update. There are cases when this is not entirely correct. It can be shown that there are sequences of revisions which give counter-intuitive results if we process each one individually. A solution to this problem is to process the sequence of revisions as a whole ([4, 13]). The main problem regarding the one-update algorithms is the fact that the belief base is not properly selected after each update, because the algorithms are only concerned with the result of the update and not with how this result occurred. This can cause the loss of valuable information. The proposed solution is based on the principle that two KBs should be considered equivalent if, in addition to the logical equivalence of the bases themselves, they will give equivalent results to all possible updates as well. This is the basic principle governing the algorithms of iterated belief revision ([13]). It is also pointed out in [4], where the difference between *belief sets* (knowledge only) and *epistemic states* (knowledge including information on how to revise it) is discussed.

An additional difficulty of the problem of belief revision is the fact that the result of an update may also depend on the *source* of the data. Let us suppose that there are two lamps, A and B, in a room and we know that exactly one of them is on. Our knowledge can be represented by the proposition: $(A\wedge\neg B)\vee(\neg A\wedge B)$. If we make the observation that lamp A is on, the update could be described by the proposition A and the intuitively correct result for the update is the proposition $A\wedge\neg B$, as we know now that B is off.

On the other hand, if a robot is sent into the room in order to turn lamp A on, then we would again have the update A. The proper intuitive result of the update is the proposition A in this case, as we know nothing about the state of lamp B; it could have been on or off before sending the robot in the room (and stayed so). This example shows that even identical (not just equivalent) databases can give different intuitively correct results with identical updates!

In order to overcome the problem, two different types of updates have been defined in [12], namely *revision* and *update*. Revision is used when new information about a static world is obtained. This is the first case of our example where the observation did not change the state of A and B. The AGM postulates and the algorithms presented deal with revisions. A revision is performed when the source of the data is an observation regarding the world. Update is used when the world dynamically changes and we have to record that change. In the second case of our example, the robot changed the state of lamp A, and consequently the state of the world being modeled. Therefore, the result of the update must be different. An update is performed when the reason of change is an action, instead of an observation. An excellent study on the problem may be found in [12], where a new set of postulates, adequate for update, is presented. As we will see, our approach will deal with both kinds of updates interchangeably, so throughout this report we will use both terms to refer to both kinds of updates.

4. Driving Considerations

Before describing how our approach deals with the above problems, we will make a short description of the logical framework used. We will restrict ourselves in KBs whose knowledge can be expressed using a finite number of propositional expressions. The underlying propositional language will contain the usual operators $(\neg,\wedge,\vee,\rightarrow,\leftrightarrow)$, parentheses, the logical constants (T,F) and a finite number of atoms $(\alpha_1,\alpha_2,\ldots,\alpha_n)$. The language will be denoted by L and the set of all propositions resulting from the language L will be denoted by L^*.

Interpretations will be viewed as an assignment of logical values (T,F) to the atoms of the language. In most cases, it will be more convenient to use the constant 0 instead of F and the constant 1 instead of T. Under this notion, an interpretation is an ordered finite sequence of size n (equal to the number of atoms in the propositional language), consisting of elements from the set {0,1}.

Regarding the representation of the base, we will adopt Nebel's proposition ([15]). Nebel proposes the use of a belief base consisting of propositional sentences representing specific observations, experiments, rules etc, out of which our knowledge is derived. In effect, under this notion, the belief base consists of the individual updates. This approach follows the foundational paradigm, as the propositions in the KB are the foundational beliefs and the rest are implied directly or indirectly by them.

We believe that the foundational approach is more compatible with common sense, and that, in principle, knowledge is actually derived from the observations we make about the world. Therefore, the storage of the observations themselves is the best way to describe our knowledge ([15]). The derived facts may be used for faster deduction and query answering, but they are of no value as far as the actual knowledge is concerned. This deals with the problem of iterated revisions as well, because each observation is actually one revision and is explicitly stored, allowing us to process the whole sequence of revisions, if this proves necessary.

At any given point in time, the real world can be represented by a unique assignment of truth-values to the atoms describing it. In other words, the real world is uniquely identified by one interpretation, which may change through time. Our goal is to find this interpretation. Usually, an observation will only give partial knowledge regarding the world, by specifying the truth-values of some, but not all, atoms in the real world. Of course, the update (observation) may contain disjunctions, which means that each disjunction describes a different possible world. Moreover, we cannot be sure in advance that the update is correct. We may only assume its correctness and if our assumption is wrong, this may lead us away from the real world.

The above remark leads us to another consideration, regarding the approval or rejection of each new update. Whenever a contradictory update is performed, the contradiction may be resolved in two ways; either we consider the update correct and try to change the KB to accommodate the update, or we consider the KB correct and reject or try to change the update in order to be accommodated in the KB.

Under most updating schemes, updates are considered more reliable than the old data. This is generally a good practice, as updates are usually regarded as the latest information about the world and can be assumed correct. However, this may not be true in several cases, as the new data may come from a noisy or otherwise unreliable source. In order to overcome the problem, we will assign a non-negative real number to each belief, which will represent its reliability. We will call this number the *Reliability Factor* of the belief and we will use the abbreviation *RF*. When a piece of contradictory information is used to revise our knowledge, at least some of the existing data (or the update itself) must be rejected. The use of the RF implies that the piece(s) of data to be rejected should be the one(s) with the lowest RF. Notice however that there are cases where small changes in existing pieces of data (or the update) are enough to accommodate the update without introducing any inconsistencies. No data rejection is necessary in this case. The selection of the data to reject or change is not an easy one, because there may be several ways to accommodate an update. Finding all possible ways to do so requires evaluating all possible subsets of our belief base and comparing all the ways of removal of the inconsistency in terms of RF cost. This is a computationally expensive operation.

5. Table Transformation

Considerations such as the above, led us to the search of an algorithm that would give us the contradictions, the possible ways of removal and the cost of removal per case at the same time. This can be done by the application of the table transformation, which transforms an expression of any finite propositional language into a 2-dimensional matrix of complex numbers.

The transformation can be applied to any proposition; however, for technical reasons it is better to use the proposition's disjunctive normal form (DNF). Any wellformed formula in propositional logic has an equivalent DNF expression, so this is not a restriction. The transformation returns a 2-dimensional matrix of complex numbers. In [7] we used ordered pairs of non-negative numbers instead of complex numbers, but the idea is pretty much the same, as ordered pairs can be assigned to complex numbers and vice-versa.

In short, each atom of the language is assigned to one column of the matrix, and there are as many lines in the matrix as the number of disjuncts in the DNF of the propositional expression. In the figure below, we show the transformation of the expression $P=(\alpha\wedge b\wedge\neg c)\vee(\alpha\wedge\neg b\wedge d)\vee(c\wedge e)$ into its respective matrix. We suppose that the language consists of 5 atoms, namely α, b, c, d and e, so the matrix has 5 columns. The expression is already in DNF and it has 3 disjuncts, so the number of lines in the matrix is 3. As far as the contents of the matrix are concerned (called *elements* hereof), the procedure is the following: an element has the value of 1 if and only if the respective atom in the respective disjunct appears as a positive atom; if negative the value is i; if the atom does not appear at all, then the value of the element is 0. The application of these rules on the expression P results in the matrix below:

$$(a\wedge b\wedge\neg c)\vee(a\wedge\neg b\wedge d)\vee(c\wedge e) \longrightarrow \begin{array}{c} \begin{matrix} \alpha & b & c & d & e \end{matrix} \\ \begin{bmatrix} 1 & 1 & i & 0 & 0 \\ 1 & i & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

Example of a table transformation

Each element in the matrix represents the type of membership of one atom in one disjunct of the proposition. This membership may be *positive* (1), *negative* (i) or *zero* (0), depending on how the atom appears in the respective disjunct. Additionally, we could assign weights (representing RF) to atoms. Moreover, elements of the form x+yi where both x and y are positive, are allowed and are called *contradictory*. A line that contains at least one contradictory element is a *contradiction*. Matrix A below has weights for each element and it has only one non-contradictory line, the second one.

$$A=\begin{bmatrix} 1+5i & 0.2+i & 5 \\ 2i & 0 & 1 \\ i & 3i & 1+i \end{bmatrix} \quad \text{(example of a weighted matrix with contradictions)}$$

Note that a contradictory element indicates the existence of both an atom and its negation in a conjunction, which is a contradiction in propositional logic. Finally, we allow elements with negative real and/or imaginary part, representing our "stiffness" in the acceptance of an atom and/or its negation.

Let us suppose that our knowledge is represented by Matrix A above. Matrix A has 3 lines, each one representing a disjunction. This indicates that the world being modeled by A has 3 *possible states*, under our current knowledge. In reality, the world has only one state, as mentioned before, but our knowledge is incomplete, so we don't know which is the correct state yet. Therefore, each line in a matrix represents one possible world. Moreover, each element of the matrix shows our confidence in each atom (or its negation), per possible world. This will prove very important later on, as new updates may cast doubts on some of our beliefs and we should know which one is more reliable.

As we will see in the following sections, the above way of defining the table transformation is not the only possible way to do so. In fact, the use of the above definition presents serious problems when complex numbers with a negative real or imaginary part are used in the matrices and when applied to propositions not in DNF. However, it's one of the simplest ways to do so if we restrict ourselves to matrices with elements with non-negative real and imaginary parts and expressions in DNF.

6. Formal Definitions

In order to define the above transformation more formally, we will first introduce some notations. As usual, we denote by $\mathbb{C}$ the set of complex numbers, by i the imaginary unit ($i=\sqrt{-1}$), by $\mathbb{R}$ the set of real numbers and by $\mathbb{R}^{(+)}$ the set of non-negative real numbers. Analogously, we denote by $\mathbb{C}^{(+)}$ the set:
$\mathbb{C}^{(+)}=\{x+yi\in\mathbb{C}|x,y\in\mathbb{R}^{(+)}\}$, ie the complex numbers whose real and imaginary part are both non-negative.

For matrices, we define $\mathbb{C}^{m\times n}$ to be the set of matrices with m lines and n columns whose elements are complex numbers, and $\mathbb{C}^{*\times n}$ the set:
$\mathbb{C}^{*\times n}=\{A\in\mathbb{C}^{m\times n}$ for some $m\in\mathbb{N}^*\}$, ie the union of $\mathbb{C}^{m\times n}$ for all $m\in\mathbb{N}^*$. In other words, $\mathbb{C}^{*\times n}$ is the set of matrices with n columns, whose elements are complex numbers. Analogously, we define the sets $\mathbb{C}^{(+)m\times n}$ and $\mathbb{C}^{(+)*\times n}$, as well as the sets $\mathbb{R}^{m\times n}$, $\mathbb{R}^{*\times n}$, $\mathbb{R}^{(+)m\times n}$ and $\mathbb{R}^{(+)*\times n}$, for matrices of real numbers.

We will use the usual notation for addition and multiplication of matrices, as well as for the multiplication of a number with a matrix. Moreover, we define the operation of *juxtaposition* as follows:

Definition 6.1

Let $A,B\in\mathbb{C}^{*\times n}$. The *juxtaposition* of A and B, denoted by A|B, is the matrix that results by placing the lines of A followed by the lines of B, ie:

$$A\,|\,B=\begin{bmatrix} A \\ --- \\ B \end{bmatrix}.$$

We will also define a partitioning on $\mathbb{C}$:

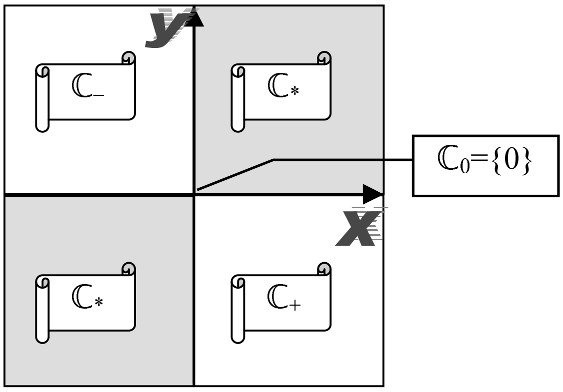
Definition 6.2

Let $z\in\mathbb{C}$ and x=Re(z), y=Im(z) its real and imaginary part, respectively. Then we define:

- If $x\geq 0$, $y\leq 0$, then z is called *positive*.
- If $x\leq 0$, $y\geq 0$, then z is called *negative*.
- If $x\cdot y>0$, then z is called *contradictory* (when $x>0$ and $y>0$ or $x<0$ and $y<0$).
- If z is both positive and negative, then z is called *zero* (when $x=y=0$).

We will denote by $\mathbb{C}_0$ the set of zero complex numbers, $\mathbb{C}_+$ the set of positive complex numbers, $\mathbb{C}_-$ the set of negative complex numbers and $\mathbb{C}_*$ the set of contradictory complex numbers.

It is obvious that $\mathbb{C}_0=\{0\}$, $\mathbb{C}_+\cap\mathbb{C}_-=\mathbb{C}_0$, $\mathbb{C}_+\cap\mathbb{C}_*=\varnothing$, $\mathbb{C}_-\cap\mathbb{C}_*=\varnothing$, $\mathbb{C}_-\cup\mathbb{C}_+\cup\mathbb{C}_*=\mathbb{C}$. Notice that this partitioning has very much to do with the informal definition of the transformation as defined above, as positive literals are assigned to positive complex numbers and negative literals are assigned to negative complex numbers. Moreover, zero complex numbers indicate no literal and contradictory complex numbers indicate both a literal and its negation, ie a contradiction. Furthermore, the real part of a number represents the RF for the positive literal, whereas the imaginary part represents the RF for the negative literal. For example, the number $-1+i\in\mathbb{C}_-$ indicates belief in the negation of the atom with a reliability of 1 and disbelief in the atom itself by reliability of 1; this implies belief in the atom's negation. The above remarks are illustrated in the figure below:



The following matrices will be useful at later stages:

Definition 6.3

- Let $A\in\mathbb{C}^{1\times n}$ be a matrix of the form [0 … 0 1 0 … 0], where the element 1 is in the k-th column. Matrix A will be called a *k-atom* or a *positive k-atom* and we will use the notation A_k.
- Let $A=z\cdot A_k\in\mathbb{C}^{1\times n}$, for some $z\in\mathbb{C}$. Matrix A will be called a *generalized k-atom*, and we will use the notation $A_k(z)$. In the special case where z=i, matrix A will be called an *inverse k-atom* or a *negative k-atom* and we will use the notation A'_k or $A_k(i)$.
- The matrix $A=[0\ 0\ \dots\ 0]\in\mathbb{C}^{1\times n}$ is called the *n-true matrix* and we will use the notation T_n.
- The matrix $A=[1+i\ \ 1+i\ \dots\ 1+i]\in\mathbb{C}^{1\times n}$ is called the *n-false matrix* and we will use the notation F_n.

The informal analysis made in the previous section shows that our goal is to assign the k-atoms to the atoms of our propositional language. T_n will be assigned to the constant T and F_n will be assigned to the constant F. Moreover, notice that the matrices T_n, F_n and $A_k(z)$, for $z\in\mathbb{C}^{(+)}$ also belong to set $\mathbb{C}^{(+)1\times n}$.

The following two propositions are immediate:

Proposition 6.1

For any matrix $A\in\mathbb{C}^{1\times n}$, there exist unique $z_j\in\mathbb{C}$, j=1,2,…,n, such that: $A=\sum_{j=1}^{n} A_j(z_j)$.

Proof

Immediate by selecting z_j to be the elements of the matrix A.
Uniqueness is derived from the definition of the generalized atoms.

Proposition 6.2

For any $m\in\mathbb{N}^*$ and any matrix $A\in\mathbb{C}^{m\times n}$, there exist unique $z_{kj}\in\mathbb{C}$, k=1,2,…,m, j=1,2,…,n, such that: $A=(\sum_{j=1}^{n}A_j(z_{1j}))\,|\,(\sum_{j=1}^{n}A_j(z_{2j}))\,|\,...\,|\,(\sum_{j=1}^{n}A_j(z_{mj}))$.

Proof

Immediate by selecting z_{kj} to be the elements of the matrix A.
Uniqueness is derived from the definition of the generalized atoms.

The above form (proposition 6.2) of a matrix A will be called the *extended normal form* of A, and denoted by ENF(A).

Definition 6.4

We define the truth constants $F=0\in\mathbb{C}$ and $T=1\in\mathbb{C}$.
Any ordered n-sized sequence of numbers in the set {0,1}={F,T} is called an *interpretation* of space $\mathbb{C}^{*\times n}$. The set of all interpretations of space $\mathbb{C}^{*\times n}$ will be denoted by I(n).

Notice that matrix interpretations can be directly assigned to logical interpretations and vice-versa, as they both are ordered finite sequences consisting of elements from the set {0,1}.

Definition 6.5

Let $I=(\alpha_1,\alpha_2,\ldots,\alpha_n)\in I(n)$ an interpretation and $A\in\mathbb{C}^{1\times n}$ a matrix such that: $A=\sum_{j=1}^{n}A_j(a_j+(1-a_j)\cdot i)$. The matrix A is called an *interpretation matrix* of space $\mathbb{C}^{*\times n}$.

Notice that there is a direct 1–1 and onto relationship between interpretations and interpretation matrices. We will use the term interpretation for the interpretation matrices whenever there is no risk of confusion. Moreover, the number $z=\alpha_j+(1-\alpha_j)\cdot i$ can be either z=1 (for $\alpha_j=1$) or z=i (for $\alpha_j=0$). Therefore, an interpretation matrix is a matrix of the set $\mathbb{C}^{1\times n}$, whose elements are from the set {1,i}.

Definition 6.6

Let $I=(\alpha_1,\alpha_2,\ldots,\alpha_n)\in I(n)$ an interpretation and $A\in\mathbb{C}^{1\times n}$. We say that A is *satisfied by* I iff the following condition holds:

$\exists z_1,z_2,...,z_n\in C^{(+)}:A=\sum_{j=1}^{n}[A_j((2\cdot a_j-1)\cdot\overline{z_j})]$, where $\overline{z_j}$ is the conjugate complex of $z_j\in\mathbb{C}^{(+)}$.

In general, if $A\in\mathbb{C}^{m\times n}$ such that $A=A^{(1)}|A^{(2)}|\ldots|A^{(m)}$, $A^{(j)}\in\mathbb{C}^{1\times n}$, j=1,2,…,m, then we say that A is *satisfied by* I iff there exists $j\in\{1,2,\ldots,m\}$ such that $A^{(j)}$ is satisfied by I.
The set of interpretations that satisfies A will be denoted by mod(A) and called the *set of models* of A.

Notice that the quantity $(2\cdot\alpha_j-1)$ can only take the values ± 1. Therefore, the quantity $x_j=(2\cdot\alpha_j-1)\cdot\overline{z_j}$ has the property that either $Re(x_j)\geq 0$ and $Im(x_j)\leq 0$ (for $\alpha_j=1$) or $Re(x_j)\leq 0$ and $Im(x_j)\geq 0$ (for $\alpha_j=0$). This supports our previous definition on positive and negative complex numbers.

Definition 6.7

Let $A\in\mathbb{C}^{*\times n}$.
If mod(A)=I(n) then A is called a *tautology*.
If mod(A)=∅ then A is called an *antinomy*.

Definition 6.8

Let $A,B\in\mathbb{C}^{*\times n}$. We will call the two matrices *equivalent* iff mod(A)=mod(B). We will denote this fact by the symbol ≅, ie $A\cong B\Leftrightarrow \text{mod}(A)=\text{mod}(B)$.

It can be proven that:

Proposition 6.3

The relation ≅ is an equivalence relation.

Proof

It is easily verified that $\forall A,B,C\in\mathbb{C}^{*\times n}$:
$\text{mod}(A)=\text{mod}(A)\Rightarrow A\cong A$,
$A\cong B\Rightarrow \text{mod}(A)=\text{mod}(B)\Rightarrow \text{mod}(B)=\text{mod}(A)\Rightarrow B\cong A$,
$(A\cong B)\wedge(B\cong C)\Rightarrow(\text{mod}(A)=\text{mod}(B))\wedge(\text{mod}(B)=\text{mod}(C))\Rightarrow \text{mod}(A)=\text{mod}(C)\Rightarrow A\cong C$.

Proposition 6.4

Let $A=[w_1\ w_2\ \ldots\ w_n]\in\mathbb{C}^{1\times n}$. Then $\text{mod}(A)=I_1\times I_2\times\ldots\times I_n$, where for any $j\in\{1,2,\ldots,n\}$ $I_j\subseteq\{0,1\}$ and:

- $0\in I_j$ iff $w_j\in\mathbb{C}_-$
- $1\in I_j$ iff $w_j\in\mathbb{C}_+$

Proof

We define $S=I_1\times I_2\times\ldots\times I_n$, where I_j as above, j=1,2,…,n.
We will prove that S=mod(A).
For any $j\in\{1,2,\ldots,n\}$, it follows that:

- $I_j=\{0\}$ iff $w_j\in\mathbb{C}_-\backslash\mathbb{C}_+$ (negative, but not positive)
- $I_j=\{1\}$ iff $w_j\in\mathbb{C}_+\backslash\mathbb{C}_-$ (positive, but not negative)
- $I_j=\{0,1\}$ iff $w_j\in\mathbb{C}_0$ (both positive and negative, ie zero)
- $I_j=\varnothing$ iff $w_j\in\mathbb{C}\backslash(\mathbb{C}_+\cup\mathbb{C}_-)=\mathbb{C}_*$ (neither positive nor negative, ie contradictory)

Let $I=(\alpha_1,\alpha_2,\ldots,\alpha_n)\in I(n)$ be an interpretation.
If $I\in\text{mod}(A)$, then, by definition 6.6:

$$\exists z_1,z_2,\ldots,z_n\in C^{(+)}: A=\sum_{j=1}^{n}[A_j((2\cdot a_j-1)\cdot\overline{z_j})].$$

By definition 6.3, for any $j\in\{1,2,\ldots,n\}$ we have that:

$$w_j=0+\ldots+0+(2\cdot a_j-1)\cdot\overline{z_j}+0+\ldots+0.$$

Thus: $w_j=(2\cdot a_j-1)\cdot\overline{z_j}$, for any $j\in\{1,2,\ldots,n\}$ (1).

At first, let us suppose that:

$S=\varnothing \Leftrightarrow \exists\ j\in\{1,2,\ldots,n\}: I_j=\varnothing \Leftrightarrow \exists\ j\in\{1,2,\ldots,n\}: w_j\in\mathbb{C}_*$ (2).

Then, if $I\in mod(A)$, by (1) together with the fact that $z_j\in\mathbb{C}^{(+)}$ and:
$\alpha_j\in\{0,1\}\Rightarrow(2\cdot\alpha_j-1)\in\{1,-1\}$,
it follows that either:

- $Re(w_j)\leq 0$, $Im(w_j)\geq 0 \Rightarrow Re(w_j)\cdot Im(w_j)\leq 0$, or:
- $Re(w_j)\geq 0$, $Im(w_j)\leq 0 \Rightarrow Re(w_j)\cdot Im(w_j)\leq 0$.

So, in any case: $Re(w_j)\cdot Im(w_j)\leq 0$, which is a contradiction because w_j is contradictory by assumption, ie $Re(w_j)\cdot Im(w_j)>0$.
This means that if $S=\varnothing$, then there is no $I\in I(n)$ such that $I\in mod(A)$, so $mod(A)=\varnothing=S$.
Let us now suppose that $S\neq\varnothing$, and $I\in S$.
Then $\alpha_j\in I_j$ for all $j\in\{1,2,\ldots,n\}$ (3).

If for any $j\in\{1,2,\ldots,n\}$ $w_j\in\mathbb{C}_*$, then by (2) we have that $S=\varnothing$, contradiction.

Therefore, for all $j\in\{1,2,\ldots,n\}$, $w_j\in\mathbb{C}_+\cup\mathbb{C}_-$ (either positive or negative, or both).
Now, for any $j\in\{1,2,\ldots,n\}$, we do the following:

- If $w_j\in\mathbb{C}_+\backslash\mathbb{C}_-$ (positive but not negative), then $Re(w_j)\geq 0$, $Im(w_j)\leq 0$ and $w_j\neq 0$. We set $z_j=\overline{w_j}$, so we have that $Re(z_j)\geq 0$, $Im(z_j)\geq 0$, ie $z_j\in\mathbb{C}^{(+)}$. In that case, by definition of the I_j, we have that $I_j=\{1\}$ and $\alpha_j\in I_j=\{1\}$ by (3), so $\alpha_j=1$.
- Similarly, if $w_j\in\mathbb{C}_-\backslash\mathbb{C}_+$ (negative but not positive), then $Re(w_j)\leq 0$, $Im(w_j)\geq 0$ and $w_j\neq 0$. We set $z_j=-\overline{w_j}$, so we have that $Re(z_j)\geq 0$, $Im(z_j)\geq 0$, ie $z_j\in\mathbb{C}^{(+)}$. In that case, by definition of the I_j, we have that $I_j=\{0\}$ and $a_j\in I_j=\{0\}$ by (3), so $\alpha_j=0$.
- Finally, if $w_j\in\mathbb{C}_0$ (both positive and negative, ie zero), then it follows that $w_j=0$. In that case, we set $z_j=0\in\mathbb{C}^{(+)}$. Moreover, we have that $I_j=\{0,1\}$ and $\alpha_j\in I_j=\{0,1\}$. So we have either $\alpha_j=0$ or $\alpha_j=1$.

In any of the above three cases, for the $z_j\in\mathbb{C}^{(+)}$ and $\alpha_j\in\{0,1\}$ as defined above, it can be easily verified that for any $j\in\{1,2,\ldots,n\}$ we have $w_j=(2\cdot a_j-1)\cdot\overline{z_j}$.
Thus, for the interpretation I and the z_j above, we have that the condition of the definition of satisfiability, rephrased in (1), holds, therefore $I\in mod(A)\Rightarrow S\subseteq mod(A)$.
Conversely, if $mod(A)=\varnothing$, then by the above result, we have that: $S\subseteq mod(A)=\varnothing\Rightarrow S=\varnothing=mod(A)$. So the equation $S=mod(A)$ holds.
If $mod(A)\neq\varnothing$, then we assume that the interpretation $I\in mod(A)$. Let us suppose that $I\notin S$. Then for some $j\in\{1,2,\ldots,n\}$ we have that $\alpha_j\notin I_j$. For that j we have:

- If $w_j\in\mathbb{C}_0$ then by definition $I_j=\{0,1\}$. Because of the fact that $\alpha_j\in\{0,1\}$ it follows that $\alpha_j\in I_j$, contradiction.
- If $w_j\in\mathbb{C}_*$ then by definition $I_j=\varnothing$, so $S=\varnothing$. We have proven that if $S=\varnothing$ then $mod(A)=\varnothing$, which is a contradiction by our assumption that $mod(A)\neq\varnothing$.
- If $w_j\in\mathbb{C}_+\backslash\mathbb{C}_-$ then by definition $I_j=\{1\}$, so together with the fact that $\alpha_j\in\{0,1\}$ and $\alpha_j\notin I_j$ it follows that $\alpha_j=0$. By substituting $\alpha_j=0$ in (1), it follows that $w_j=-\overline{z_j}$. But $z_j\in\mathbb{C}^{(+)}\Rightarrow Re(z_j)\geq 0$ and $Im(z_j)\geq 0\Rightarrow Re(w_j)\leq 0$ and $Im(w_j)\geq 0\Rightarrow w_j\in\mathbb{C}_-$, contradiction.
- Similarly, if $w_j\in\mathbb{C}_-\backslash\mathbb{C}_+$, then by definition $I_j=\{0\}$, so together with the fact that $\alpha_j\in\{0,1\}$ and $\alpha_j\notin I_j$ it follows that $\alpha_j=1$. By substituting $\alpha_j=1$ in (1), it follows that

$w_j = \overline{z_j}$. But $z_j \in \mathbb{C}^{(+)} \Rightarrow Re(z_j) \geq 0$ and $Im(z_j) \geq 0 \Rightarrow Re(w_j) \geq 0$ and $Im(w_j) \leq 0 \Rightarrow w_j \in \mathbb{C}_+$, contradiction.

Thus, we have reached a contradiction in all the cases. This means that $I \in S \Rightarrow mod(A) \subseteq S$.

This fact together with $S \subseteq mod(A)$ gives that $S = mod(A)$, and the proof is complete.

The above proposition has some important consequences, whose proofs are immediate:

Corollary 6.1

Let $A=[w_1\ w_2\ \ldots\ w_n] \in \mathbb{C}^{1\times n}$. Then A is an antinomy iff $\exists j \in \{1,2,\ldots,n\}$ such that $w_j \in \mathbb{C}_*$.

The following proposition may be more useful from time to time:

Corollary 6.2

Let $A=[w_1\ w_2\ \ldots\ w_n] \in \mathbb{C}^{1\times n}$. Then $mod(A)=I_1 \times I_2 \times \ldots \times I_n$, where for any $j \in \{1,2,\ldots,n\}$:

- $I_j=\{0\}$ iff $w_j \in \mathbb{C}_- \backslash \mathbb{C}_+$
- $I_j=\{1\}$ iff $w_j \in \mathbb{C}_+ \backslash \mathbb{C}_-$
- $I_j=\{0,1\}$ iff $w_j \in \mathbb{C}_0$
- $I_j=\emptyset$ iff $w_j \in \mathbb{C}_*$

Corollary 6.3

Let $A=[w_1\ w_2\ \ldots\ w_n] \in \mathbb{C}^{1\times n}$. If A is not an antinomy, then $|mod(A)|=2^k$ where k is the number of zero elements in the matrix.

Corollary 6.4

The following equations hold:

- For any $k \in \{1,2,\ldots,n\}$, $mod(A_k)=I_1 \times I_2 \times \ldots \times I_n$, where $I_j=\{0,1\}$, for $j \in \{1,2,\ldots,n\} \backslash \{k\}$ and $I_k=\{1\}$.
- For any $k \in \{1,2,\ldots,n\}$, $z \in \mathbb{C}$, $mod(A_k(z))=I_1 \times I_2 \times \ldots \times I_n$, where $I_j=\{0,1\}$, for $j \in \{1,2,\ldots,n\} \backslash \{k\}$ and:
  - $I_k=\{0\}$ iff $z \in \mathbb{C}_- \backslash \mathbb{C}_+$
  - $I_k=\{1\}$ iff $z \in \mathbb{C}_+ \backslash \mathbb{C}_-$
  - $I_k=\{0,1\}$ iff $z \in \mathbb{C}_0$
  - $I_k=\emptyset$ iff $z \in \mathbb{C}_*$
- $mod(F_n)=\emptyset$
- $mod(T_n)=I(n)$

Corollary 6.5

Let $A=[w_1\ w_2\ \ldots\ w_n] \in \mathbb{C}^{1\times n}$. A is a tautology iff all $w_j \in \mathbb{C}_0$.

The above corollary actually says that the only tautology in $\mathbb{C}^{1\times n}$ is the matrix $T_n \in \mathbb{C}^{1\times n}$.

Corollary 6.6

Let A∈$\mathbb{C}^{m\times n}$, such that:

$$A=\begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \dots & \dots & \dots & \dots \\ w_{m1} & w_{m2} & \dots & w_{mn} \end{bmatrix}$$

Then $\mathrm{mod}(A)=\bigcup_{j=1}^{m}(I_{j1}\times I_{j2}\times\ldots\times I_{jn})$, where for any j∈{1,2,…,m}, k∈{1,2,…,n}:

- $0\in I_{jk}$ iff $w_{jk}\in\mathbb{C}_-$
- $1\in I_{jk}$ iff $w_{jk}\in\mathbb{C}_+$

The above corollary gives us an immediate, constructive method to calculate mod(A) for any matrix A. The proposition below provides another useful method:

Proposition 6.5

Let $m\in\mathbb{N}^*$ and $A\in\mathbb{C}^{m\times n}$.

Moreover, let: $A=(\sum_{=1}^{n} A\ \ (z_1\ \ ))\,|\,(\sum_{=1}^{n} A\ \ (z_2\ \ ))\,|\ldots|\,(\sum_{=1}^{n} A\ \ (z_m\ \ ))$, be its ENF.

Then: $\mathrm{mod}(A)=\bigcup_{k=1}^{m}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))$.

Proof

Initially, let us suppose that $m=1\in\mathbb{N}^*$. Then: $A=\sum_{=1}^{n} A\ \ (z_1\ \ )$.

It holds that for any j∈{1,2,…,n}, $z_{1j}\in\mathbb{C}$, $\mathrm{mod}(A_j(z_{1j}))=I_{1j}\times I_{2j}\times\ldots\times I_{nj}$, where $I_{kj}=\{0,1\}$, for k∈{1,2,…,n}\{j} and:

- $I_{jj}=\{0\}$ iff $z_{1j}\in\mathbb{C}_-\setminus\mathbb{C}_+$
- $I_{jj}=\{1\}$ iff $z_{1j}\in\mathbb{C}_+\setminus\mathbb{C}_-$
- $I_{jj}=\{0,1\}$ iff $z_{1j}\in\mathbb{C}_0$
- $I_{jj}=\varnothing$ iff $z_{1j}\in\mathbb{C}_*$

We have that:

$$\bigcup_{k=1}^{1}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))=\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{1j}))=\bigcap_{j=1}^{n}(I_{1j}\times I_{2j}\times\ldots\times I_{nj})\Rightarrow$$

$$\Rightarrow\bigcup_{k=1}^{1}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))=(\bigcap_{j=1}^{n}I_{1j})\times(\bigcap_{j=1}^{n}I_{2j})\times\ldots\times(\bigcap_{j=1}^{n}I_{nj})\ (1).$$

However, for all k,j∈{1,2,…,n}, it holds that $I_{kj}=\{0,1\}$ whenever k≠j and $I_{kj}\subseteq\{0,1\}$ whenever k=j, therefore:

$$(1)\Rightarrow\bigcup_{k=1}^{1}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))=I_{11}\times I_{22}\times\ldots\times I_{nn}\ (2).$$

Furthermore, $\mathrm{mod}(A)=I^{(1)}\times I^{(2)}\times\ldots\times I^{(n)}$, where for any j∈{1,2,…,n} $I^{(j)}\subseteq\{0,1\}$ and:

- $I^{(j)}=\{0\}$ iff $z_{1j}\in\mathbb{C}_-\setminus\mathbb{C}_+$
- $I^{(j)}=\{1\}$ iff $z_{1j}\in\mathbb{C}_+\setminus\mathbb{C}_-$

- $I^{(j)}=\{0,1\}$ iff $z_{1j}\in\mathbb{C}_0$
- $I^{(j)}=\varnothing$ iff $z_{1j}\in\mathbb{C}_*$

It follows that for any $j\in\{1,2,\ldots,n\}$ we have $I_{jj}=I^{(j)}$, therefore:
$I^{(1)}\times I^{(2)}\times\ldots\times I^{(n)}=I_{11}\times I_{22}\times\ldots\times I_{nn}\Rightarrow$

$\mathrm{mod}(A)=\bigcup_{k=1}^{1}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))$, and the proof is complete for m=1.

In the general case, where $m\in\mathbb{N}^*$, we have by definition 6.6:

$$\mathrm{mod}(A)=\mathrm{mod}((\sum_{j=1}^{n}A_j(z_{1j}))\,|\,(\sum_{j=1}^{n}A_j(z_{2j}))\,|\ldots|\,(\sum_{j=1}^{n}A_j(z_{mj})))\Rightarrow$$

$$\Rightarrow\mathrm{mod}(A)=\bigcup_{k=1}^{m}\mathrm{mod}(\sum_{j=1}^{n}A_j(z_{kj}))\ (3).$$

But, for any $k\in\{1,2,\ldots,m\}$, $\sum_{j=1}^{n}A_j(z_{kj})\in C^{1\times n}$, therefore:

$\mathrm{mod}(\sum_{j=1}^{n}A_j(z_{kj}))=\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))$, thus, combining with (3):

$\mathrm{mod}(A)=\bigcup_{k=1}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))$, and the proof is complete.

Now that we are able to calculate the models of any matrix, it may be useful to define the quotient space of $\mathbb{C}^{*\times n}$ with respect to the equivalence relation ≅:

Definition 6.9

We define the space $\mathbb{C}^{*\times n/\cong}\equiv\mathbb{C}^{*\times n}/\cong$, the quotient space of matrices with respect to the relation ≅. Similarly, we define $\mathbb{C}^{k\times n/\cong}$, $\mathbb{C}^{(+)*\times n/\cong}$ and $\mathbb{C}^{(+)k\times n/\cong}$ for any $k\in\mathbb{N}^*$.

All the above spaces are finite and they are all isomorphic to each other, because they are all isomorphic with the space $P(\{0,1\}^n)\equiv P(I(n))$, ie the power set of interpretations of n atoms.

Our next step will be to define operations on matrices that will emulate the usual operations of disjunction, conjunction and negation in classical propositional logic:

Definition 6.10

We define three classes of functions in $\mathbb{C}^{*\times n}$, denoted by $\mathcal{F}_\vee$, $\mathcal{F}_\wedge$ and $\mathcal{F}_\neg$, called the *class of disjunction functions*, the *class of conjunction functions* and the *class of negation functions* respectively.

- A function $f_\vee:\mathbb{C}^{*\times n}\times\mathbb{C}^{*\times n}\to\mathbb{C}^{*\times n}$ is said to belong in the class $\mathcal{F}_\vee$ iff for any $A,B\in\mathbb{C}^{*\times n}$, $\mathrm{mod}(f_\vee(A,B))=\mathrm{mod}(A)\cup\mathrm{mod}(B)$.
  We will also use the notation $A\vee B$ for the result of $f_\vee(A,B)$.
- A function $f_\wedge:\mathbb{C}^{*\times n}\times\mathbb{C}^{*\times n}\to\mathbb{C}^{*\times n}$ is said to belong in the class $\mathcal{F}_\wedge$ iff for any $A,B\in\mathbb{C}^{*\times n}$, $\mathrm{mod}(f_\wedge(A,B))=\mathrm{mod}(A)\cap\mathrm{mod}(B)$.
  We will also use the notation $A\wedge B$ for the result of $f_\wedge(A,B)$.
- A function $f_\neg:\mathbb{C}^{*\times n}\to\mathbb{C}^{*\times n}$ is said to belong in the class $\mathcal{F}_\neg$ iff for any $A\in\mathbb{C}^{*\times n}$, $\mathrm{mod}(f_\neg(A))=I(n)\setminus\mathrm{mod}(A)$.

We will also use the notation ¬A for the result of $f_{\neg}(A)$.

<u>*Definition 6.11*</u>

The space $\mathbb{C}^{*\times n}$, equipped with three functions denoted by ∨, ∧, ¬ belonging to the classes $\mathcal{F}_{\vee}$, $\mathcal{F}_{\wedge}$ and $\mathcal{F}_{\neg}$ respectively ($\vee\in\mathcal{F}_{\vee}$, $\wedge\in\mathcal{F}_{\wedge}$, $\neg\in\mathcal{F}_{\neg}$), is called a *logically complete matrix space of dimension n*. We will denote such spaces with the quadruple: $(\mathbb{C}^{*\times n},\vee,\wedge,\neg)$.

Notice that we do not set any restrictions on the selection of the three operations, as long as they satisfy the conditions of the definition of the three classes of functions. Obviously, there is more than one possible selection for these operators. Different selections for the operators ∨, ∧, ¬ result in different logically complete spaces. However, the selection is irrelevant when it comes to quotient spaces. Moreover, in this work, we assume that the definition of additional operators is not needed. Therefore, operators such as the implication (→), will not be defined between matrices.

The above definitions and propositions are enough to support the definition of the *table transformation function*:

<u>*Definition 6.12*</u>

Let $(\mathbb{C}^{*\times n},\vee,\wedge,\neg)$ be a logically complete matrix space of dimension n and $L=\{T,F,(,),\vee,\wedge,\neg,\alpha_1,\alpha_2,\ldots,\alpha_n\}$ a finite propositional language. As usual, we denote by α_j the atoms of the propositional language and by A_j the atoms of $\mathbb{C}^{*\times n}$.

We define the *table transformation function*, $TT:L^*\to\mathbb{C}^{*\times n}$, recursively as follows:

- $TT(T)=T_n$
- $TT(F)=F_n$
- For any $j\in\{1,2,\ldots,n\}$: $TT(\alpha_j)=A_j$
- $TT(p\vee q)=TT(p)\vee TT(q)$, for any $p,q\in L^*$
- $TT(p\wedge q)=TT(p)\wedge TT(q)$, for any $p,q\in L^*$
- $TT(\neg p)=\neg TT(p)$, for any $p\in L^*$

Similarly, we define the *inverse table transformation function*, $TTI:\mathbb{C}^{*\times n}\to L^*$, recursively, as follows:

- For any $j\in\{1,2,\ldots,n\}$, $z\in\mathbb{C}$, we define:
  - $TTI(A_j(z))=\alpha_j$, iff $z\in\mathbb{C}_+\setminus\mathbb{C}_-$
  - $TTI(A_j(z))=\neg\alpha_j$, iff $z\in\mathbb{C}_-\setminus\mathbb{C}_+$
  - $TTI(A_j(z))=T$, iff $z\in\mathbb{C}_0$
  - $TTI(A_j(z))=F$, iff $z\in\mathbb{C}_*$
- In general, for any $m\in\mathbb{N}^*$ and any matrix $A\in\mathbb{C}^{m\times n}$, whose ENF is $A=(\sum_{j=1}^{n}A_j(z_{1j}))\,|\,(\sum_{j=1}^{n}A_j(z\ \ _j))\,|\ldots|\,(\sum_{j=1}^{n}A_j(z_{mj}))$, we define:

  $$TTI(A)=\underset{k=1}{\vee}(\bigwedge_{j=1}^{n}TTI(A_j(z_{kj})))$$

The transformations above have a very important property:

Proposition 6.6

Let $(\mathbb{C}^{*\times n},\vee,\wedge,\neg)$ be a logically complete matrix space of dimension n and $L=\{T,F,(,),\vee,\wedge,\neg,\alpha_1,\alpha_2,\ldots,\alpha_n\}$ a finite propositional language. Then:

- For any proposition $p\in L^*$ we have: mod(p)=mod(TT(p)).
- For any matrix $P\in\mathbb{C}^{*\times n}$ we have: mod(P)=mod(TTI(P)).

Proof

Let $p\in L^*$.

- If p=T then mod(p)=I(n) and mod(TT(p))=mod(T_n)=I(n)=mod(p).
- If p=F then mod(p)=$\varnothing$ and mod(TT(p))=mod(F_n)=$\varnothing$=mod(p).
- If $p=\alpha_j$ for some $j\in\{1,2,\ldots,n\}$, then $\text{mod}(p)=\{I=(i_1,i_2,\ldots,i_n)\in I(n):\ i_k\in\{0,1\}$ for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $i_j=1\}=I_1\times I_2\times\ldots\times I_n$, where $I_k=\{0,1\}$ for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $I_j=\{1\}$. By corollary 6.4, $\text{mod}(TT(p))=\text{mod}(A_j)=J_1\times J_2\times\ldots\times J_n$, where $J_k=\{0,1\}$ $k\in\{1,2,\ldots,n\}\setminus\{j\}$, $J_j=\{1\}$. Thus, $I_j=J_j$, for all $j\in\{1,2,\ldots,n\}\Rightarrow$mod(p)=mod(TT(p)).
- In general, if $p\in L^*$, then it is known that p is expressed as a finite sequence of atoms and constants, separated by operators $(\vee,\wedge,\neg)$.
  For any $q,r\in L^*$, we have that $\text{mod}(q\vee r)=\text{mod}(q)\cup\text{mod}(r)$ and $\text{mod}(TT(q\vee r))=\text{mod}(TT(q)\vee TT(r))=\text{mod}(TT(q))\cup\text{mod}(TT(r))$.
  Similarly, for any $q,r\in L^*$, we have that $\text{mod}(q\wedge r)=\text{mod}(q)\cap\text{mod}(r)$ and $\text{mod}(TT(q\wedge r))=\text{mod}(TT(q)\wedge TT(r))=\text{mod}(TT(q))\cap\text{mod}(TT(r))$.
  Moreover, for any $q\in L^*$, we have that $\text{mod}(\neg q)=I(n)\setminus\text{mod}(q)$ and $\text{mod}(TT(\neg q))=\text{mod}(\neg TT(q))=I(n)\setminus\text{mod}(TT(q))$.
  The initial proposition p can be broken down into simpler propositions within a finite number of steps, and mod(p) can be expressed using mod(α_j), $j\in\{1,2,\ldots,n\}$, I(n), $\varnothing$ and the operations of union ($\cup$), join ($\cap$) and subtraction ($\setminus$). Moreover, we already prove that mod(α_j)=mod(TT(α_j)) for any $j\in\{1,2,\ldots,n\}$, mod(T)=mod(TT(T)) and mod(F)=mod(TT(F)). This recursively proves that mod(p)=mod(TT(p)).

Now, let $P\in\mathbb{C}^{*\times n}$ and let $P=(\sum_{j=1}^{n}A_j(z_{1j}))\,|\,(\sum_{j=1}^{n}A_j(z_{\ \ j}))\,|\ldots|\,(\sum_{j=1}^{n}A_j(z_{mj}))$, be its ENF.

Then, by definition:

$$\text{mod}(TTI(P))=\text{mod}(\mathop{\vee}_{k=1}^{m}(\mathop{\wedge}_{j=1}^{n}TTI(A_j(z_{kj}))))=\bigcup_{k=1}^{m}\bigcap_{j=1}^{n}\text{mod}(TTI(A_j(z_{kj}))\ \ (1).$$

Let $j\in\{1,2,\ldots,n\}$, $z\in\mathbb{C}$. Then:

- If $z\in\mathbb{C}_+\setminus\mathbb{C}_-$, $TTI(A_j(z))=\alpha_j\Rightarrow\text{mod}(TTI(A_j(z)))=\text{mod}(\alpha_j)=\{I=(i_1,i_2,\ldots,i_n)\in I(n):\ i_k\in\{0,1\},\ k\in\{1,2,\ldots,n\}/\{j\}$ and $i_j=1\}=I_1\times I_2\times\ldots\times I_n$, where $I_k=\{0,1\}$, for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $I_j=\{1\}$. By corollary 6.4, we have that $\text{mod}(A_j(z))=J_1\times J_2\times\ldots\times J_n$, where $J_k=\{0,1\}$, for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $J_j=\{1\}$, since $z\in\mathbb{C}_+\setminus\mathbb{C}_-$. Thus $I_j=J_j$ for all $j\in\{1,2,\ldots,n\}\Rightarrow\text{mod}(TTI(A_j(z)))=\text{mod}(A_j(z))$.
- If $z\in\mathbb{C}_-\setminus\mathbb{C}_+$, $TTI(A_j(z))=\neg\alpha_j\Rightarrow\text{mod}(TTI(A_j(z)))=\text{mod}(\neg\alpha_j)=\{I=(i_1,i_2,\ldots,i_n)\in I(n):\ i_k\in\{0,1\},\ k\in\{1,2,\ldots,n\}/\{j\}$ and $i_j=0\}=I_1\times I_2\times\ldots\times I_n$, where $I_k=\{0,1\}$, for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $I_j=\{0\}$. By corollary 6.4, we have that $\text{mod}(A_j(z))=J_1\times J_2\times\ldots\times J_n$, where $J_k=\{0,1\}$, for $k\in\{1,2,\ldots,n\}\setminus\{j\}$ and $J_j=\{0\}$, since $z\in\mathbb{C}_-\setminus\mathbb{C}_+$. Thus $I_j=J_j$ for all $j\in\{1,2,\ldots,n\}\Rightarrow\text{mod}(TTI(A_j(z)))=\text{mod}(A_j(z))$.
- If $z\in\mathbb{C}_0$, $TTI(A_j(z))=T\Rightarrow\text{mod}(TTI(A_j(z)))=\text{mod}(T)=I(n)$. Moreover, $A_j(z)=A_j(0)=T_n$, so $\text{mod}(A_j(z))=\text{mod}(T_n)=I(n)=\text{mod}(TTI(A_j(z)))$.

- Finally, if $z \in \mathbb{C}_*$, $TTI(A_j(z))=F \Rightarrow mod(TTI(A_j(z)))=mod(F)=\varnothing$. Moreover, $A_j(z)$ will have at least one element which is contradictory, the one in the j-th column, whose value is $z \in \mathbb{C}_*$. Thus $mod(A_j(z))=\varnothing=mod(TTI(A_j(z)))$.

Thus, for all $z \in \mathbb{C}$ and all $j \in \{1,2,\ldots,n\}$ we have that $mod(TTI(A_j(z)))=mod(A_j(z))$.

Thus: $(1) \Rightarrow \mathrm{mod}(TTI(P)) = \bigcup_{k=1}^{m}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj}))$.

Combining the above formula with proposition 6.5 we get:

$\mathrm{mod}(TTI(P)) = \bigcup_{k=1}^{m}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{kj})) = \mathrm{mod}(P)$, and the proof is complete.

The above proposition shows that the transformation of a matrix to a logical expression and vice-versa, does not cause any loss of information. This is true because any interpretation that satisfies a given matrix satisfies its respective logical expression (via the TTI function) and any interpretation that satisfies a given logical expression satisfies its respective matrix (via the TT function).

Moreover, notice that the above proposition holds regardless of the selection of the operations ∨, ∧, ¬. Similar results, applied to matrices of the space $\mathbb{C}^{(+)^* \times n}$ can be found in [7] and a more formal foundation of the properties of the space $\mathbb{C}^{(+)^* \times n}$ can be found in section 12 ("Positive and Negative Knowledge") of this report.

7. Belief Revision

Our initial goal on defining the table transformation was to use this transformation in belief revision. In order to solve this problem, we will assume that both the knowledge and the update are represented by matrices. This way we skip for the moment the need for using the TT function, which depends on the definition of the operators ∧, ∨, ¬. Moreover, the use of matrices gives us flexibility on the RF of each atom. By using the TT function, we are obliged to use fixed RFs for all atoms, which is a serious constraint.

For the moment, we will additionally assume that both the base K and the update M have only one line, so they both represent only one possible world. In this special case, the update will be defined as the addition of the two matrices, because the inclusion of the update M in our knowledge increases our trust in all the information that M carries, effectively increasing our reliance in each atom and/or its negation. This adjustment may or may not be enough to force us to change our beliefs. Let us see one example:

$$K = \begin{bmatrix} i & 3 & 0 \end{bmatrix}, M = \begin{bmatrix} 3 & 2 & 1 \end{bmatrix}$$

Using the TTI function we can easily verify that the matrices represent the expressions K=¬A∧B∧T≅¬A∧B (base) and M=A∧B∧C (update). The matrices additionally show the reliability per atom in each proposition. In this case, the negation of atom A (¬A) is believed with an RF of 1 in K, but M should force us to abandon this belief as atom A is believed with an RF of 3 in M. In atom B, there is no contradiction between K and M; however this revision should increase our confidence in the truth of B. Finally, in atom C, we know now that C is true, with a reliance of 1; we had no knowledge regarding C before. The resulting matrix is K′ below where the symbol "•" stands for the operation of update:

$$K' = K \bullet M = \begin{bmatrix} i & 3 & 0 \end{bmatrix} + \begin{bmatrix} 3 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 3+i & 5 & 1 \end{bmatrix}$$

The proposition related (via the TTI function) to the resulting matrix is: F∧B∧C≅F. Note that the resulting matrix K′ is a contradictory matrix (and the related proposition is a contradiction too). This is not generally acceptable, as a contradictory KB actually contains no information. The result should have been A∧B∧C, as showed by the previous analysis. We will deal with this problem later (which is in fact not a problem at all!).

In the general case where one (or both) of the matrices contain more than one line, each line represents one possible world. In order to be sure that the correct world will be represented by a line in the resulting matrix, we must add each line of matrix K with each line of matrix M, creating one line per pair in the resulting matrix K′.

Formally, we define the operation of revision as follows:

<u>*Definition 7.1*</u>

Let $A,B \in \mathbb{C}^{*\times n}$, where $A=A^{(1)}|A^{(2)}|\ldots|A^{(k)}$, $B=B^{(1)}|B^{(2)}|\ldots|B^{(m)}$, for some $k,m \in \mathbb{N}^*$, $A^{(j)} \in \mathbb{C}^{1\times n}$, $j \in \{1,2,\ldots,k\}$ and $B^{(j)} \in \mathbb{C}^{1\times n}$, $j \in \{1,2,\ldots,m\}$. We define the operation of revision, denoted by •, between those two matrices as follows:

$$A \bullet B = \mathop{|}_{h=1,j=1}^{h=k,j=m} (A^{(h)} + B^{(j)}) .$$

It is easy to verify that this definition concurs with the above informal description of the revision operator.

8. Queries and Contradictory Lines

In general, the resulting matrix of a revision may contain contradictory lines, like in the previous example. One may argue that contradictory lines represent contradictory possible worlds, so they contain false information and they could as well be deleted. However, this is not entirely true. A single faulty revision may create a contradiction in an otherwise correct possible world (line), so we must be careful before discarding a contradictory line as non-true. On the other hand, even if a line (possible world) contains no contradictions at all, this is by no means a guarantee that it is entirely true. It could be too far from the real world; we just don't know it yet. Therefore, our policy is to keep the matrix as-is, even if some (or all) lines are contradictory.

In order to execute queries upon the KB we must transform the matrix representing the knowledge into a logical expression. We cannot use the TTI function directly, as this could in some cases result to an unsatisfiable proposition, like in the example above, where the matrix consists of contradictory lines. In fact, the TTI function ignores the contradictory lines (they are equivalent to the constant F), which is wrong according to the above discussion. The solution to the problem of contradictory lines is to transform (some of) them to non-contradictory ones before applying the TTI function. Whenever a query is executed on the KB, our answer must be based on the "most correct" lines, ignoring the rest. The criterion for correctness of a line is a quantity named *Line Reliability* (RL). We also define the *Element Reliability* (RE). The reliability of an element should depend only on the element itself and the reliability of a line should depend on the reliability of its elements.

The number of lines to be selected as most correct for the query should be a user-defined parameter depending on the application. For example, we could take the k most reliable ones; or all lines with reliability more than a given number; or use any other method that the user may find suitable. The function that selects some (or all) of the lines of a matrix will be denoted by LS and the term *Line Selection Function* will be used. The result is a submatrix of the original KB.

Subsequently, we define the *Normalized Matrix* of a given matrix A. Each element x+yi of the matrix A is changed into the element x–y. Under this transformation, when x<y the result will be a negative real number, ie a complex number belonging to the set $\mathbb{C}_-$; when x>y the result will be a positive real number, ie a complex number belonging to the set $\mathbb{C}_+$; if x=y the result will be number 0, ie a number belonging to the set $\mathbb{C}_0$. The function that performs this operation on the matrix will be called *Matrix Normalization Function* and denoted by MN. In fact, MN is applied only on the part of the matrix that was selected by the LS function.

Following that, the inverse of the table transformation (TTI) is used to transform the normalized matrix that resulted from the above function into a logical expression. The result is the related logical expression of the matrix and represents the base upon which the query is executed. The whole process (function) of transforming a matrix to a proposition for the needs of queries will be denoted by QT (*Query Transformation*). It is clear by the above analysis that QT is in fact a composite function. Notice that the above operation does not actually change the matrix; it temporarily transforms it to a logical expression for the needs of queries. The next update will be executed upon the original matrix and the whole process of query transformation should be repeated after the update to calculate the new proposition.

9. Formalizing the Query Transformation

In the previous section, we described the informal process that is followed in order to extract the information contained in the matrix representing the KB. In this section, we will attempt to formalize the query transformation. It is clear that the process of extracting a proposition from a matrix is composed of several steps. Some of them are based on known and clearly defined transformations (like the TTI function), whereas others are not clearly set (RE, RL, LS functions). This is done on purpose, in order to allow the parameterization of the process of query answering. However, these parameters should not be set in an arbitrary sense; instead, they must satisfy some rationality constraints. This is done for the same reason that we defined desired properties regarding the logical operators $\wedge$, $\vee$, $\neg$. Such functions can be freely defined, but they are subject to specific constraints in order to be rational.

In order to formalize the notion of element and line reliability we define some new classes of functions:

<u>*Definition 9.1*</u>

A function $RE_0:\mathbb{C}\to\mathbb{R}^+$ is said to belong in the class $\mathcal{F}_{RE0}$ of *Primitive Element Reliability Functions* iff RE_0 has the following properties:

- $RE_0(x+yi)=RE_0(y+xi)$ for all $x,y\in\mathbb{R}$ (reliability is symmetric with respect to the first diagonal axis)
- $RE_0(z)=RE_0(-z)$ for all $z\in\mathbb{C}$ (reliability is symmetric with respect to 0)

A function $RE:\mathbb{C}^{*\times n}\to\mathbb{R}^{(+)*\times n}$ is said to belong in the class $\mathcal{F}_{RE}$ of *Element Reliability Functions* iff for any $A\in\mathbb{C}^{*\times n}$, $A=[a_{ij}]$, there exists a function $RE_0\in\mathcal{F}_{RE0}$ such that $RE(A)=[RE_0(a_{ij})]$.

A function $RL_0:\mathbb{R}^{(+)1\times n}\to\mathbb{R}^{(+)}$ is said to belong in the class $\mathcal{F}_{RL0}$ of *Primitive Line Reliability Functions* iff RL_0 has the following properties:

- For any $j,k \in \{1,2,\ldots,n\}$, $j<k$ and $x_1,x_2,\ldots,x_n \in \mathbb{R}^{(+)}$, the following equation holds: $RL_0(x_1,x_2,\ldots,x_j,\ldots,x_k,\ldots,x_n)=RL_0(x_1,x_2,\ldots,x_k,\ldots,x_j,\ldots,x_n)$ (swapping of the variables' indexes does not change the result of the function)
- For any $x,y,x_2,\ldots,x_n \in \mathbb{R}^{(+)}$, such that $y \geq x$, it holds that $RL_0(y,x_2,\ldots,x_n) \geq RL_0(x,x_2,\ldots,x_n)$ (RL_0 is monotonically increasing with respect to the first of its variables)

A function $RL:\mathbb{R}^{(+)*\times n} \to \mathbb{R}^{(+)*\times 1}$ is said to belong in the class $\mathcal{F}_{RL}$ of *Line Reliability Functions* iff for any $k \in \mathbb{N}$ and $A \in \mathbb{R}^{(+)k\times n}$, such that $A=A^{(1)}|A^{(2)}|\ldots|A^{(k)}$, $A^{(j)} \in \mathbb{R}^{(+)1\times n}$, $j \in \{1,2,\ldots,k\}$, there exists a function $RL_0 \in \mathcal{F}_{RL0}$ such that:

$$RL(A) = \begin{bmatrix} RL_0(A^{(1)}) \\ RL_0(A^{(2)}) \\ \ldots \\ RL_0(A^{(k)}) \end{bmatrix}.$$

Notice that the functions belonging to $\mathcal{F}_{RL0}$ are also monotonically increasing with respect to each one of their variables, because of the first property of such functions.

Some comments regarding the notion behind the definition of these properties for the classes $\mathcal{F}_{RE}$ and $\mathcal{F}_{RL}$ are in order. First, it is clear that an element's reliability should depend solely on the element itself and not on other elements of the matrix, even if they are in the same line or column. This is the reason for the definition of the separate class of $\mathcal{F}_{RE0}$ functions. The same thoughts led us to the definition of the $\mathcal{F}_{RL0}$ class of functions, as a line's reliability depends only on the reliabilities of the elements of the line itself.

Regarding the properties of the class $\mathcal{F}_{RE0}$, the first property indicates the fact that we should have no bias towards positive or negative atoms. In other words, for a complex number x+yi, which can be equivalently expressed as a pair (x,y), the fact that x refers to the RF of the positive atom and y refers to the RF of the negative atom, should be irrelevant as far as the element's reliability is concerned. Therefore, x+yi should have the same reliability as y+xi.

Moreover, the semantics of the RF indicate that our willingness to believe or disbelieve an atom (or its negation) depends on the absolute value of its RF (|RF|). The sign of the RF indicates whether it refers to *believing* or *disbelieving*. Therefore, the only effect that the RF's sign should have on the element's reliability has to do with whether the atom's RF (real part of the complex number) and the RF of the atom's negation (imaginary part of the complex number) have the same sign or not. If the signs are the same, then it is irrelevant whether they refer to believing or disbelieving; it is a contradiction anyway. Thus, by changing the sign of both the real and the imaginary part, a contradiction remains a contradiction and its intensity remains the same, as the absolute values of the RFs do not change. Similarly, if the RF signs are not the same, then we believe in either an atom or its negation; by changing both signs we simply switch between believing the atom and believing the atom's negation. In any case, given that there should be no bias on whether we believe the atom or its negation, the element's reliability is unaffected by this switching, as the absolute values of RFs do not change. Summarizing, the simultaneous switching of the signs of the real and imaginary part of a complex number should not affect this number's reliability. This is the fact expressed by the second property attached to the $\mathcal{F}_{RE0}$ class of functions.

The properties of the $\mathcal{F}_{RL0}$ class of functions are more easily justified. The first property refers to the obvious fact that a line's reliability should be unaffected by the order with which the atoms appear in the matrix. The second property expresses the requirement that the increase of an element's reliability could not possibly cause the decrease of the overall reliability of the line it belongs to.

Regarding the LS function, we have the following two definitions:

Definition 9.2

A function $LS:\mathbb{R}^{(+)*\times 1}\rightarrow P(\mathbb{N}^*)$ is said to belong in the class $\mathcal{F}_{LS}$ of *Line Selection Functions* iff for any $k\in\mathbb{N}^*$ and any $A\in\mathbb{R}^{(+)k\times 1}$, the following condition holds: $\varnothing\subset LS(A)\subseteq\{1,2,\ldots,k\}$ (we must select at least one index, in the range 1…k).

The only property of the $\mathcal{F}_{LS}$ class is included in order for the indices returned by the LS function to have "legitimate" values, ie to describe at least one line of the matrix and to be within the acceptable range.

Definition 9.3

We define the *submatrix selection* function $MS:\mathbb{C}^{*\times n}\times P(\mathbb{N}^*)\rightarrow\mathbb{C}^{*\times n}$. For any $k\in\mathbb{N}^*$, $A\in\mathbb{C}^{k\times n}$, $S\subseteq\mathbb{N}^*$, such that $A=A^{(1)}|A^{(2)}|\ldots|A^{(k)}$, $A^{(j)}\in\mathbb{C}^{1\times n}$, $j\in\{1,2,\ldots,k\}$, we define:

$MS(A,S)=A$, iff $S=\varnothing$ or there exists $m\in S$ such that $m>k$,

$MS(A,S)=\underset{j\in S}{|}A^{(j)}$, otherwise.

The LS and MS functions are used to select a submatrix of the original matrix for the query. The selected submatrix is normalized using the following function:

Definition 9.4

We define the *matrix normalization* function $MN:\mathbb{C}^{*\times n}\rightarrow\mathbb{C}^{*\times n}$. For any $k\in\mathbb{N}^*$, $A\in\mathbb{C}^{k\times n}$, such that $A=[a_{ij}]$, $i\in\{1,2,\ldots,k\}$, $j\in\{1,2,\ldots,n\}$, we define:

$MN(A)=B\in\mathbb{C}^{k\times n}$, where $B=[b_{ij}]$ and for all $i\in\{1,2,\ldots,k\}$, $j\in\{1,2,\ldots,n\}$:

$b_{ij}=Re(a_{ij})-Im(a_{ij})$

In effect, an element reliability function assigns to each element of the matrix its reliability; a line reliability function assigns to each line of the matrix its reliability; whereas a line selection function selects a non-empty subset of the lines of the matrix. The MS function is well defined, and returns a submatrix of A which consists of some of the lines of A, those whose indexes belong to the set S. In the abnormal cases where $S=\varnothing$ or S contains faulty indexes (out of range), the function returns the matrix A. Such cases will not appear in this application, but are included for completeness. Finally, the MN function is used to normalize the result, and the normalized matrix is transformed to a logical proposition using the TTI function.

The informal analysis made in the previous section, can be more formally set forth using the formalization presented above. More specifically, the correct way to transform a matrix $A\in\mathbb{C}^{k\times n}$ (for some $k\in\mathbb{N}^*$) into a propositional expression for the needs of queries (QT function) is composed of the following steps:

1) Apply the user-defined $RE \in \mathcal{F}_{RE}$ function on the matrix A to get $B=RE(A) \in \mathbb{R}^{(+)k\times n}$.
2) Apply the user-defined $RL \in \mathcal{F}_{RL}$ function on the matrix B to get $C=RL(B) \in \mathbb{R}^{(+)k\times 1}$.
3) Apply the user-defined $LS \in \mathcal{F}_{LS}$ function on the matrix C to get $S=LS(C) \subseteq \{1,2,\ldots,k\}$, $S \neq \varnothing$.
4) Select the submatrix: $D=MS(C,S)=MS(C,LS(C)) \in \mathbb{C}^{m\times n}$, for some $m \in \mathbb{N}^*$, $m \leq k$.
5) Apply the MN function on the matrix D to get $E=MN(D) \in \mathbb{C}^{m\times n}$.
6) Apply the TTI function on the result to get $p=TTI(E) \in L^*$.

With the formalization of the query answering process at hand, we have now completed the description of our method. Some properties of the method and its relation with the considerations described in section 4 ("Driving Considerations") can be immediately deducted. Our algorithm keeps one matrix for the base, and not each individual update, but the retention of contradictory lines implies that the individual updates are kept in an encoded fashion. Moreover, iterated revisions are supported, as we keep track of all the previous updates and RFs. Finally, the rejection of the old data when answering queries is made according to the RF and the functions RE and RL, and is minimal with respect to these quantities (where the notion of minimality depends on the definition of the LS function).

The algorithm of revision is clearly defined; however, the flexibility in the definition of the query transformation process allows us to relate the same matrix with several different propositions, depending on the selection of some parameters. This means that we have in fact defined a whole class of algorithms. The optimum algorithm of this class can only be determined after several tests and it may be application-dependent. However, we have reduced the problem of finding the optimum belief revision algorithm of this class in the problem of specifying the following parameters of the above algorithm:

1) The RFs of the atoms of the KB and update(s). This may be done in a fixed manner (for example setting all atoms of the update and/or KB to an RF of 1) or in a free manner (letting the user freely specify the RF of the updates, depending on his confidence on each atom, disjunction or proposition).
2) The RL and RE functions. These functions, along with parameter 3 below, play a very important role in the selection of the lines to be used in the queries. There are in general two types of such functions: those that depend only on the existence of a contradiction and those that depend on the "intensity" of each contradiction. The "intensity" of a contradiction may be defined in different ways and the best way can only be determined by running several tests of updates on the KB and taking into account the specific application that we are interested in.
3) The lines to be selected for the query (function LS). This may be a fixed number of lines, or a function of the number of updates, the number of lines of the matrix or some other parameter. Once again, the best criterion is application-dependent, and can only be determined after extensive tests.

All these parameters are subject to the rationality constraints mentioned above. An additional parameter may be the selection of the logical operators $\wedge$, $\vee$, $\neg$; however, we consider this of lesser importance, as these operators may not be necessary in general for the definition of the revision algorithm (if the DNF of propositions is used). Finally, a fifth parameter, also of lesser importance, will be introduced in section 11 ("Complexity Issues"), referring to the procedure of abruption that we will define.

10. Other Operations

The operation of contraction is dual to revision, and it has been argued ([8, 11, 14]) that it is intuitively simpler to deal with contraction instead of revision. For a certain class of update schemes, namely those that satisfy the AGM postulates, it has been proven that revision and contraction can be defined in terms of each other. Similarly, update is an important operation, whose dual is erasure ([12]).

As already mentioned above, our algorithm deals with all the above operations interchangeably, despite their different nature. More specifically, contraction is the process of removing, instead of adding, knowledge from a KB. This naturally implies that contraction of a matrix K with a matrix M, could simply be the revision of matrix K with matrix –M. Naturally, we could apply the minus sign to all the elements of M, thus eliminating the need for a specific contraction operator.

Normally, knowledge is expressed using elements from the set $\mathbb{C}^{(+)}$. Elements with negative real or imaginary parts actually imply the loss of confidence in a certain proposition, which is the notion behind contraction. It follows that in the same update matrix M, we could do both revision and contraction, by simply setting some of the elements to have negative real or imaginary parts and some of the elements to have positive real and imaginary parts. This is an option not available in other update schemes.

The integration of the update operator into our update scheme is somewhat more complex. Updating with a proposition p, actually means that the world has changed in such a way that p is now true. All previous knowledge is irrelevant, because it refers to a previous state, contrary to revision where previous knowledge is important, because it refers to the same state of the world. These thoughts indirectly imply that the matrix P that represents proposition p should be "enhanced" in such a way as to remove the previous knowledge as well as to add the current one.

We will see how this is possible with an example. We will use the TT function as was informally defined in section 5 ("Table Transformation"), and apply the method to the example we used in section 3 ("Properties of Belief Revision") in order to outline the difference between revision and update. In that example, we had the proposition: (A∧¬B)∨(¬A∧B) for our base and the proposition A for the update/revision. Remember that the proper intuitive result was A∧¬B for the revision and A for the update. Turning into matrices we have the following matrix for the KB:

$$K = \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}$$

Regarding the update, one may notice that both matrices M_1 and M_2 below:

$$M_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 1-i & 0 \end{bmatrix}$$

refer to the proposition A, ie TTI(M_1)=TTI(M_2)=A. However, M_1 contains the information that A is true, whereas M_2 additionally informs the base that ¬A is no longer true. Moreover, the update of K with M_1 gives a different result than the update of K with M_2:

$$K_1 = K \bullet M_1 = \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & i \\ 1+i & 1 \end{bmatrix}$$

$$K_2 = K \bullet M_2 = \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix} \bullet \begin{bmatrix} 1-i & 0 \end{bmatrix} = \begin{bmatrix} 2-i & i \\ 1 & 1 \end{bmatrix}$$

The actual proposition that will be used in queries obviously depends on the selection of the RE, RL and LS functions. Notice however, that K_1 contains only one non-contradictory line, the first one, while both lines of K_2 are non-contradictory. So, most (but not all) of the rational selections of the parameters RE, RL and LS will give the proposition $A\wedge\neg B$ for K_1 (first line only) and the proposition $(A\wedge\neg B)\vee(A\wedge B)\cong A$ for K_2 (both lines selected). Therefore, we have performed a revision by using M_1 and an update by using M_2; both using the same algorithm!

Having defined the update operator, we can now define the operation of erasure, which is dual to update. We can do so in a manner similar to the extension of revision for the integration of contraction. Specifically, erasure with matrix M is equivalent to update with matrix –M.

11. Complexity Issues

The computational complexity of our update algorithm depends on the number of elements per matrix. For a knowledge base with n_0 lines and m columns and an update of n_1 lines and m columns, the resulting matrix will have $n_0 \cdot n_1$ lines and m columns. To calculate each element of the resulting matrix, we must perform 2 primitive additions, so the worst-case operations (additions) required are: $[(n_0 \cdot n_1)\cdot m]\cdot 2$; that implies a complexity of $O(n_0 \cdot n_1 \cdot m)$.

Continuing the above example, for a knowledge base with n_0 lines and m columns and after k consecutive updates, with $n_1, n_2, \ldots, n_k$ lines respectively and m columns per update, the resulting matrix will have $n_0 \cdot n_1 \cdot \ldots \cdot n_k$ lines and m columns. This implies that the number of lines in the matrix representing the knowledge base increases exponentially with the number of updates, and so does the cost for each new update as well as the memory space required to store the base.

This may be unacceptable in most cases, so one could decide to reject some of the lines of the matrix by a procedure based on line reliability, which is called *abruption*. The number of lines to be removed should be an application-dependent, user-defined parameter, representing a trade-off between knowledge integrity and processing speed.

Finally, the complexity of the query transformation procedure on any matrix depends on the size of the matrix, and is analogous to the number of elements of the matrix. The exact complexity cannot be computed unless the functions RE, RL and LS are known, but it is clear that the technique of abruption will speed up things, regardless of the above selection.

12. Positive and Negative Knowledge

We have already stressed the fact that matrices' elements can have negative real or imaginary parts. Such numbers indicate lack of confidence to a given literal and/or its negation. This means that they do not give direct information on the truth or falsity of a literal; instead, they indirectly imply its truth or falsity by specifying distrust in its falsity or truth respectively. Such kind of knowledge will be denoted by the term *negative knowledge* contrary to elements with non-negative parts (real and imaginary), which will be referred to as *positive knowledge*.

The analysis made in section 10 ("Other Operations") has pointed out the necessity of negative knowledge. However, if we restrict ourselves to revisions, it is enough to deal with positive knowledge only. This restriction simplifies things a lot, as many more interesting results can be proven in space $\mathbb{C}^{(+)*\times n}$, which may not be true in the general case ($\mathbb{C}^{*\times n}$). The updating schemes proposed in the literature do not refer to negative knowledge, as there is no way to express such knowledge in propositional logic. Therefore, this restriction (and its consequences) may prove helpful in our attempt to compare our work with existing revision schemes.

First of all, it is obvious that all the results referring to $\mathbb{C}^{*\times n}$, can be successfully applied to $\mathbb{C}^{(+)*\times n}$, as $\mathbb{C}^{(+)}\subseteq\mathbb{C}$. Moreover, all definitions can be rephrased to refer to $\mathbb{C}^{(+)*\times n}$. Additionally, in $\mathbb{C}^{(+)*\times n}$, there is a very simple way to define the operations $\wedge$, $\vee$, $\neg$, which does not work in $\mathbb{C}^{*\times n}$. The properties of $\mathbb{C}^{(+)*\times n}$ are based on the following simple, but important remarks:

- $\mathbb{C}^{(+)}\cap\mathbb{C}_{+}=\{x+yi\in\mathbb{C}: y=0, x\geq 0\}$
- $\mathbb{C}^{(+)}\cap\mathbb{C}_{-}=\{x+yi\in\mathbb{C}: x=0, y\geq 0\}$
- $\mathbb{C}^{(+)}\cap\mathbb{C}_{*}=\{x+yi\in\mathbb{C}: x>0, y>0\}$
- $\mathbb{C}^{(+)}\cap\mathbb{C}_{0}=\mathbb{C}_{0}=\{0\}$

It is trivial to verify that the above equations hold, by the application of the definitions of the above sets. Using these relations, we can prove three important propositions:

Proposition 12.1

Let $A,B\in\mathbb{C}^{*\times n}$. We define the operation $\vee$ as: $A\vee B=A|B$. Then, the operator $\vee$ belongs to the class of disjunction functions $\mathcal{F}_{\vee}$.

Proof

From the definition of satisfiability, it is obvious that $mod(A|B)=mod(A)\cup mod(B)$ for any $A,B\in\mathbb{C}^{*\times n}$, so the operator $\vee$ as defined above belongs to the class of disjunction functions $\mathcal{F}_{\vee}$.

Notice that this proposition holds in $\mathbb{C}^{*\times n}$, and not only in $\mathbb{C}^{(+)*\times n}$. However, the proposition below does not hold in $\mathbb{C}^{*\times n}$:

Proposition 12.2

Let $A,B\in\mathbb{C}^{(+)*\times n}$. We define the operation $\wedge$ as: $A\wedge B=A\bullet B$. Then, the operator $\wedge$ belongs to the class of conjunction functions $\mathcal{F}_{\wedge}$.

Proof

We must prove that $mod(A\bullet B)=mod(A)\cap mod(B)$.

First of all, let us suppose that $A,B\in\mathbb{C}^{(+)1\times n}$. In this case, $A\bullet B=A+B$.

Let us suppose that $A=[w_1\ w_2\ \ldots\ w_n]$, $B=[z_1\ z_2\ \ldots\ z_n]$, $w_j,z_j\in\mathbb{C}^{(+)}$, for all $j\in\{1,2,\ldots,n\}$. Thus:

$C=A\bullet B=[w_1+z_1\ w_2+z_2\ \ldots\ w_n+z_n]$.

Obviously, $w_j+z_j\in\mathbb{C}^{(+)}$, for all $j\in\{1,2,\ldots,n\}$.

By proposition 6.4, we have that:

$mod(A)=I_{A1}\times I_{A2}\times\ldots\times I_{An}$ (1)

$mod(B)=I_{B1}\times I_{B2}\times\ldots\times I_{Bn}$ (2)

$mod(C)=I_{C1}\times I_{C2}\times\ldots\times I_{Cn}$ (3)

where, as usual, for any $j\in\{1,2,\ldots,n\}$:

$0 \in I_{Aj}$ iff $w_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)}$

$1 \in I_{Aj}$ iff $w_j \in \mathbb{C}_+ \cap \mathbb{C}^{(+)}$

$0 \in I_{Bj}$ iff $z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)}$

$1 \in I_{Bj}$ iff $z_j \in \mathbb{C}_+ \cap \mathbb{C}^{(+)}$

$0 \in I_{Cj}$ iff $w_j + z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)}$

$1 \in I_{Cj}$ iff $w_j + z_j \in \mathbb{C}_+ \cap \mathbb{C}^{(+)}$

It follows that $0 \in I_{Cj} \Leftrightarrow w_j + z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)} \Leftrightarrow \mathrm{Re}(w_j + z_j) = 0$ and $\mathrm{Im}(w_j + z_j) \geq 0 \Leftrightarrow$
$\Leftrightarrow \mathrm{Re}(w_j) + \mathrm{Re}(z_j) = 0$ and $\mathrm{Im}(w_j) + \mathrm{Im}(z_j) \geq 0$ (4).

Given that $w_j, z_j \in \mathbb{C}^{(+)}$, we have, by definition, that: $\mathrm{Re}(w_j) \geq 0$, $\mathrm{Im}(w_j) \geq 0$, $\mathrm{Re}(z_j) \geq 0$ and $\mathrm{Im}(z_j) \geq 0$.
Combining (4) with the above relations we have that:

$\mathrm{Re}(w_j) = \mathrm{Re}(z_j) = 0$ and $\mathrm{Im}(w_j) \geq 0$, $\mathrm{Im}(z_j) \geq 0$, so: $w_j, z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)}$, which implies that $0 \in I_{Aj}$ and $0 \in I_{Bj}$.
Thus, $0 \in I_{Cj} \Rightarrow 0 \in I_{Aj}$ and $0 \in I_{Bj}$.

Similarly, if $0 \in I_{Aj}$ and $0 \in I_{Bj}$, then $w_j, z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)}$ so:
$\mathrm{Re}(w_j) = 0$, $\mathrm{Re}(z_j) = 0$, $\mathrm{Im}(w_j) \geq 0$ and $\mathrm{Im}(z_j) \geq 0$, which means that:

$\mathrm{Re}(w_j + z_j) = 0$ and $\mathrm{Im}(w_j + z_j) \geq 0 \Rightarrow w_j + z_j \in \mathbb{C}_- \cap \mathbb{C}^{(+)} \Rightarrow 0 \in I_{Cj}$.
Summarizing we have that: $0 \in I_{Cj} \Leftrightarrow 0 \in I_{Aj}$ and $0 \in I_{Bj}$.
Using the same technique, it can be proven that: $1 \in I_{Cj} \Leftrightarrow 1 \in I_{Aj}$ and $1 \in I_{Bj}$.
The above two equivalences, along with the fact that $I_{Aj} \subseteq \{0,1\}$, $I_{Bj} \subseteq \{0,1\}$ and $I_{Cj} \subseteq \{0,1\}$, imply that $I_{Cj} = I_{Aj} \cap I_{Bj}$, and this holds for all $j \in \{1,2,\ldots,n\}$. Moreover:
$\mathrm{mod}(A) \cap \mathrm{mod}(B) = (I_{A1} \times I_{A2} \times \ldots \times I_{An}) \cap (I_{B1} \times I_{B2} \times \ldots \times I_{Bn}) =$
$= (I_{A1} \cap I_{B1}) \times (I_{A2} \cap I_{B2}) \times \ldots \times (I_{An} \cap I_{Bn}) = I_{C1} \times I_{C2} \times \ldots \times I_{Cn} = \mathrm{mod}(C) = \mathrm{mod}(A \bullet B) = \mathrm{mod}(A \wedge B)$,

and the proof is complete for $A, B \in \mathbb{C}^{(+)1\times n}$.

In the general case, where $A, B \in \mathbb{C}^{(+)*\times n}$, let us suppose that:

$A = A^{(1)}|A^{(2)}|\ldots|A^{(k)}$ and $B = B^{(1)}|B^{(2)}|\ldots|B^{(m)}$, for some $k, m \in \mathbb{N}^*$, where $A^{(j)} \in \mathbb{C}^{(+)1\times n}$, $j \in \{1,2,\ldots,k\}$ and $B^{(j)} \in \mathbb{C}^{(+)1\times n}$, $j \in \{1,2,\ldots,m\}$.
Then:
$\mathrm{mod}(A) \cap \mathrm{mod}(B) = \mathrm{mod}(A^{(1)}|A^{(2)}|\ldots|A^{(k)}) \cap \mathrm{mod}(B^{(1)}|B^{(2)}|\ldots|B^{(m)}) =$
$= [\mathrm{mod}(A^{(1)}) \cup \mathrm{mod}(A^{(2)}) \cup \ldots \cup \mathrm{mod}(A^{(k)})] \cap [\mathrm{mod}(B^{(1)}) \cup \mathrm{mod}(B^{(2)}) \cup \ldots \cup \mathrm{mod}(B^{(m)})] =$

$$= \bigcup_{h=1, j=1}^{h=k, j=m} (\mathrm{mod}(A^{(h)}) \cap \mathrm{mod}(B^{(j)})) .$$

Because of the fact that $A^{(h)}, B^{(j)} \in \mathbb{C}^{(+)1\times n}$, for all $h \in \{1,2,\ldots,k\}$, $j \in \{1,2,\ldots,m\}$, it follows that:
$\mathrm{mod}(A^{(h)}) \cap \mathrm{mod}(B^{(j)}) = \mathrm{mod}(A^{(h)} \wedge B^{(j)}) = \mathrm{mod}(A^{(h)} + B^{(j)})$.
Thus:

$$\mathrm{mod}(A) \cap \mathrm{mod}(B) = \bigcup_{h=1, j=1}^{h=k, j=m} \mathrm{mod}(A^{(h)} + B^{(j)}) = \mathrm{mod}(\Big|_{h=1, j=1}^{h=k, j=m} (A^{(h)} + B^{(j)})) = \mathrm{mod}(A \bullet B),$$

by definition 7.1, and the proof is complete.

Proposition 12.3

Let $A \in \mathbb{C}^{(+)*\times n}$. We define the operation $\neg$ recursively as follows:

- If $A=A_j(z)$ for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap(\mathbb{C}_+\setminus\mathbb{C}_-)$, then $\neg A=A_j(z\cdot i)$
- If $A=A_j(z)$ for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap(\mathbb{C}_-\setminus\mathbb{C}_+)$, then $\neg A=A_j(-z\cdot i)$
- If $A=A_j(z)$ for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap\mathbb{C}_0$, then $\neg A=F_n$
- If $A=A_j(z)$ for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap\mathbb{C}_*$, then $\neg A=A_j(\mathrm{Re}(z)\cdot i)\vee A_j(\mathrm{Im}(z))$
- In the general case, let $A\in\mathbb{C}^{(+)m\times n}$, for some $m\in\mathbb{N}^*$ and:

  $A=(\sum_{j=1}^{n}A_j(z_{1j}))\,|\,(\sum_{j=1}^{n}A_j(z_{2j}))\,|\dots|\,(\sum_{j=1}^{n}A_j(z_{mj}))$ be its ENF. Then:

  $\neg A=\bigwedge_{h=1}^{m}\bigvee_{j=1}^{n}(\neg A_j(z_{hj}))$.

Under this definition, the operator $\neg$ belongs to the class of negation functions $\mathcal{F}_{\neg}$.

Proof

If $A=A_j(z)$, for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap(\mathbb{C}_+\setminus\mathbb{C}_-)$, then:
$\mathrm{mod}(A)=I_1\times I_2\times\dots\times I_n$, where $I_k=\{0,1\}$, $k\in\{1,2,\dots,n\}\setminus\{j\}$ and $I_j=\{1\}$.

On the other hand, $z\cdot i\in\mathbb{C}^{(+)}\cap(\mathbb{C}_-\setminus\mathbb{C}_+)$, therefore:
$\mathrm{mod}(\neg A)=\mathrm{mod}(A_j(z\cdot i))=I^{(1)}\times I^{(2)}\times\dots\times I^{(n)}$, where $I^{(k)}=\{0,1\}$, $k\in\{1,2,\dots,n\}\setminus\{j\}$ and $I^{(j)}=\{0\}$.
Thus: $\mathrm{mod}(\neg A)=I(n)\setminus\mathrm{mod}(A)$ in this case.

Similarly, if $A=A_j(z)$, for some $j\in\{1,2,\dots,n\}$, $z\in\mathbb{C}^{(+)}\cap(\mathbb{C}_-\setminus\mathbb{C}_+)$, then:
$\mathrm{mod}(A)=I_1\times I_2\times\dots\times I_n$, where $I_k=\{0,1\}$, $k\in\{1,2,\dots,n\}\setminus\{j\}$ and $I_j=\{0\}$.

On the other hand, $-z\cdot i\in\mathbb{C}^{(+)}\cap(\mathbb{C}_+\setminus\mathbb{C}_-)$, therefore:
$\mathrm{mod}(\neg A)=\mathrm{mod}(A_j(-z\cdot i))=I^{(1)}\times I^{(2)}\times\dots\times I^{(n)}$, where $I^{(k)}=\{0,1\}$, $k\in\{1,2,\dots,n\}\setminus\{j\}$ and $I^{(j)}=\{1\}$.
Thus: $\mathrm{mod}(\neg A)=I(n)\setminus\mathrm{mod}(A)$ in this case.

If $z\in\mathbb{C}^{(+)}\cap\mathbb{C}_0$, then $A_j(z)=T_n\Rightarrow\mathrm{mod}(A)=\mathrm{mod}(T_n)=I(n)$. On the other hand:
$\mathrm{mod}(\neg A)=\mathrm{mod}(F_n)=\varnothing\Rightarrow\mathrm{mod}(\neg A)=I(n)\setminus\mathrm{mod}(A)$.

Finally, if $z\in\mathbb{C}^{(+)}\cap\mathbb{C}_*$, then $\mathrm{mod}(A)=\mathrm{mod}(A_j(z))=\varnothing$. On the other hand:

$\mathrm{Re}(z)\cdot i\in\mathbb{C}^{(+)}\cap(\mathbb{C}_-\setminus\mathbb{C}_+)$ and $\mathrm{Im}(z)\in\mathbb{C}^{(+)}\cap(\mathbb{C}_+\setminus\mathbb{C}_-)$. Thus:
$\mathrm{mod}(A_j(\mathrm{Re}(z)\cdot i))=I_1\times I_2\times\dots\times I_n$ and $\mathrm{mod}(A_j(\mathrm{Im}(z)))=I^{(1)}\times I^{(2)}\times\dots\times I^{(n)}$, where:

- $I_h=I^{(h)}=\{0,1\}$ for $h\in\{1,2,\dots,n\}\setminus\{j\}$
- $I_j=\{0\}$
- $I^{(j)}=\{1\}$

Therefore:
$\mathrm{mod}(\neg A)=\mathrm{mod}(A_j(\mathrm{Re}(z)\cdot i)\vee A_j(\mathrm{Im}(z)))=\mathrm{mod}(A_j(\mathrm{Re}(z)\cdot i))\cup\mathrm{mod}(A_j(\mathrm{Im}(z)))=I(n)\Rightarrow$
$\Rightarrow\mathrm{mod}(\neg A)=I(n)\setminus\mathrm{mod}(A)$.
In the general case, we have:

$$\mathrm{mod}(\neg A)=\mathrm{mod}(\bigwedge_{h=1}^{m}\bigvee_{j=1}^{n}(\neg A_j(z_{hj})))=\bigcap_{h=1}^{m}\bigcup_{j=1}^{n}\mathrm{mod}(\neg A_j(z_{hj}))=$$

$$=\bigcap_{h=1}^{m}\bigcup_{j=1}^{n}(I(n)\setminus\mathrm{mod}(A_j(z_{hj})))=I(n)\setminus\bigcup_{h=1}^{m}\bigcap_{j=1}^{n}\mathrm{mod}(A_j(z_{hj}))=I(n)\setminus\mathrm{mod}(A),$$ and the proof is complete.

With these three operations at hand, we have defined a logically complete matrix space of dimension n. We also have the operation of revision, as well as the obvious result that all the above operations (disjunction, conjunction, negation, revision) give results in $\mathbb{C}^{(+)*\times n}$ when applied to matrices in $\mathbb{C}^{(+)*\times n}$. Therefore, our framework is complete. Let us now see how we can apply this framework to emulate Dalal's algorithm using the table transformation.

13. Comparison with Dalal's Algorithm

In this section, we will try to find the required parameters for the emulation of Dalal's algorithm. For a detailed description of this algorithm, see [2, 3]. In short, for a KB K and an update U, the updated base K•U has as its models the interpretation(s) of mod(U) which are closest to the interpretations of mod(K). Closeness between two interpretations is defined in terms of the number of atoms in which the values of these two interpretations differ.

In all of the following, we will assume that the underlying propositional language is $L=\{\alpha_1,\alpha_2,\ldots,\alpha_n,\wedge,\vee,\neg,T,F,(,)\}$ and that the operations of disjunction, conjunction and negation upon matrices are as defined in the previous section. The underlying matrix space is $\mathbb{C}^{(+)*\times n}$. A min-term in propositional logic is a proposition in DNF, which is a conjunction of atoms or their negation. So, if $p\in L^*$ and it is a min-term, then there exist b_j, $j=1,2,\ldots,m$ such that: $p=b_1\wedge b_2\wedge\ldots\wedge b_m$, and for all $j\in\{1,2,\ldots,m\}$, $b_j=\alpha$ or $b_j=\neg\alpha$ for some atom $\alpha\in L^*$. Moreover, for all $j,k\in\{1,2,\ldots,m\}$, $j\neq k$ $b_j\neq b_k$ and $b_j\neq\neg b_k$, ie there is no pair of b_js that refer to the same atom. When referring to a min-term, we will implicitly assume that it is a satisfiable proposition (not equivalent to the constant F).

Lemma 13.1

We define the function $RE_0:\mathbb{C}\to\mathbb{R}^+$, such that for any $z\in\mathbb{C}$, with $x=Re(z)$, $y=Im(z)$, we have:
$RE_0(z)=RE_0(x+yi)=1-\min\{1,|x|,|y|\}$.

Moreover, we define the function $RE:\mathbb{C}^{*\times n}\to\mathbb{R}^{(+)*\times n}$, such that for any matrix $A=[z_{ij}]$:
$RE(A)=[RE_0(z_{ij})]$.

Then: $RE_0\in\mathcal{F}_{RE0}$ and $RE\in\mathcal{F}_{RE}$.

Proof

Let $z\in\mathbb{C}$, such that $Re(z)=x$, $Im(z)=y$.

It is obvious that for all $z\in\mathbb{C}$, $0\leq\min\{1,|x|,|y|\}\leq1\Rightarrow$

$\Rightarrow0\leq1-\min\{1,|x|,|y|\}\leq1\Rightarrow RE_0(z)\geq0\Rightarrow RE_0(z)\in\mathbb{R}^+$ for all $z\in\mathbb{C}$.

Furthermore:
$RE_0(x+yi)=1-\min\{1,|x|,|y|\}=1-\min\{1,|y|,|x|\}=RE_0(y+xi)$ and
$RE_0(x+yi)=1-\min\{1,|x|,|y|\}=1-\min\{1,|-x|,|-y|\}=RE_0(-x-yi)$.

This means that $RE_0\in\mathcal{F}_{RE0}$. By this fact, and the definition of RE, it follows that $RE\in\mathcal{F}_{RE}$, and the proof is complete.

Lemma 13.2

For all $z\in\mathbb{C}^{(+)}$, $Re(z)=x$, $Im(z)=y$, such that $x\leq1$ or $y\leq1$, it holds that:
$RE_0(z)=1-\min\{x,y\}$, where RE_0 the function of the above lemma.

Proof

It is obvious that $|x|=x$, $|y|=y$ and $\min\{1,|x|,|y|\}=\min\{1,x,y\}=\min\{x,y\}$. Thus, the lemma holds.

Lemma 13.3

We define the function $RL_0:\mathbb{R}^{(+)1\times n}\to\mathbb{R}^{+}$, such that for any $x_1,x_2,\ldots,x_n\in\mathbb{R}^{(+)}$: $RL_0(x_1,x_2,\ldots,x_n)=x_1+x_2+\ldots+x_n$.

Moreover, we define the function $RL:\mathbb{R}^{(+)*\times n}\to\mathbb{R}^{(+)*\times 1}$, such that for any $k\in\mathbb{N}^*$ and $A\in\mathbb{R}^{(+)k\times n}$, with $A=A^{(1)}|A^{(2)}|\ldots|A^{(k)}$, $A^{(j)}\in\mathbb{R}^{(+)1\times n}$, $j\in\{1,2,\ldots,k\}$:

$$RL(A)=\begin{bmatrix} RL_0(A^{(1)}) \\ RL_0(A^{(2)}) \\ \ldots \\ RL_0(A^{(n)}) \end{bmatrix}.$$

Then: $RL_0\in\mathcal{F}_{RL0}$ and $RL\in\mathcal{F}_{RL}$.

Proof

For any $j,k\in\{1,2,\ldots,n\}$, $j<k$ and $x_1,x_2,\ldots,x_n\in\mathbb{R}^{(+)}$ it holds trivially that: $RL_0(x_1,x_2,\ldots,x_j,\ldots,x_k,\ldots,x_n)=RL_0(x_1,x_2,\ldots,x_k,\ldots,x_j,\ldots,x_n)$.

Moreover, it is obvious that for any $x,y,x_2,\ldots,x_n\in\mathbb{R}^{(+)}$, such that $y\geq x$, it holds that $RL_0(y,x_2,\ldots,x_n)\geq RL_0(x,x_2,\ldots,x_n)$.

Therefore $RL_0\in\mathcal{F}_{RL0}$. By this fact, and the definition of RL, it follows that $RL\in\mathcal{F}_{RL}$, and the proof is complete.

Lemma 13.4

Let $LS:\mathbb{R}^{(+)*\times 1}\to P(\mathbb{N}^*)$ be the function such that for any $k\in\mathbb{N}^*$, $A\in\mathbb{R}^{(+)k\times 1}$ such that:

$$A=\begin{bmatrix} x_1 \\ x_2 \\ \ldots \\ x_k \end{bmatrix},$$

we have that:

$LS(A)=\{m\in\{1,2,\ldots,k\}:\forall j\in\{1,2,\ldots,k\}\ x_m\geq x_j\}$. Then $LS\in\mathcal{F}_{LS}$.

Proof

Obvious by the definition of $\mathcal{F}_{LS}$.

Lemma 13.5

Let G be the transformation defined in Dalal's papers ([2, 3]). Then for all propositions $p,q\in L^*$ and all $k\in\mathbb{N}^*$ the following equation holds: $G^k(p\vee q)\cong G^k(p)\vee G^k(q)$.

Proof

For k=0, $G^0(p)=p$ for all $p\in L^*$, so the equation trivially holds.

For k=1, by the definition of G in [2], we have that:

$mod(G^1(p\vee q))=mod(G(p\vee q))=g(mod(p\vee q))=g(mod(p)\cup mod(q))=$

$$=\bigcup_{I\in \mathrm{mod}(p)\cup \mathrm{mod}(q)} g(I)=(\bigcup_{I\in \mathrm{mod}(p)} g(I))\cup(\bigcup_{I\in \mathrm{mod}(q)} g(I))=g(\mathrm{mod}(p))\cup g(\mathrm{mod}(q))=$$

$=mod(G(p))\cup mod(G(q))=mod(G(p)\vee G(q))=mod(G^1(p)\vee G^1(q))$, which means that $G^1(p\vee q)\cong G^1(p)\vee G^1(q)$.

Let us suppose that the equation holds for all k=0,1,2,…,m. All we have to do is prove it for k=m+1.

Indeed:
$G^{m+1}(p\vee q)\cong G^{m}(G(p\vee q))\cong G^{m}(G(p)\vee G(q))\cong G^{m}(G(p))\vee G^{m}(G(q))\cong G^{m+1}(p)\vee G^{m+1}(q)$, and the proof is complete.

Lemma 13.6
Let $p\in L^{*}$, such that p is a min-term. Then, there exist unique $P^{0},P^{+},P^{-}\subseteq\{1,2,\ldots,n\}$, such that $P^{+}\cap P^{-}=\varnothing$, $P^{+}\cap P^{0}=\varnothing$, $P^{-}\cap P^{0}=\varnothing$, $P^{0}\cup P^{+}\cup P^{-}=\{1,2,\ldots,n\}$ and:

$$TT(p)=\sum_{j\in P^{+}}A_j+\sum_{j\in P^{-}}A_j(i).$$

Proof
Given that p is a min-term, there exists a set $S\subseteq\{1,2,\ldots,n\}$ such that:
$p=\wedge_{j\in S}b_j$, where $b_j=\alpha_j$ or $b_j=\neg\alpha_j$ for all $j\in S$.
We select:
$P^{+}=\{j\in S:b_j=\alpha_j\}$, $P^{-}=\{j\in S:b_j=\neg\alpha_j\}$, $P^{0}=\{1,2,\ldots,n\}\backslash(P^{+}\cup P^{-})$.
It follows that:
$P^{0},P^{+},P^{-}\subseteq\{1,2,\ldots,n\}$, $P^{+}\cap P^{-}=\varnothing$, $P^{+}\cap P^{0}=\varnothing$, $P^{-}\cap P^{0}=\varnothing$, $P^{0}\cup P^{+}\cup P^{-}=\{1,2,\ldots,n\}$.
Moreover, $P^{+}\cup P^{-}=S$.
Finally, by the definition of TT and the operator $\wedge$, if follows that:

$$TT(p)=TT(\wedge_{j\in S}b_j)=\wedge_{j\in S}TT(b_j)=\sum_{j\in S}TT(b_j)=\sum_{j\in\ {}^{+}}TT(a_j)+\sum_{j\in\ {}^{-}}TT(\neg a_j)=$$

$$=\sum_{j\in P^{+}}A_j+\sum_{j\in P^{-}}\neg A_j=\sum_{j\in P^{+}}A_j+\sum_{j\in P^{-}}A_j(i).$$

Now, let us suppose that there is a different triple of sets Q^{+},Q^{-},Q^{0} with the above properties. Then we have that:

$$TT(p)=\sum_{j\in P^{+}}A_j+\sum_{j\in P^{-}}A_j(i)=\sum_{j\in Q^{+}}A_j+\sum_{j\in Q^{-}}A_j(i).$$

By the definition of the $A_j(z)$, we have that for any $j\in\{1,2,\ldots,n\}$, the above equation will give:

- $j\in P^{+}\Leftrightarrow j\in Q^{+}$ and
- $j\in P^{-}\Leftrightarrow j\in Q^{-}$

which actually means that $P^{+}=Q^{+}$ and $P^{-}=Q^{-}$. Given the properties of the sets, we have that $P^{0}=Q^{0}$ as well, so the above sets are unique.

The above lemma actually says that there is a direct correspondence between a min-term $p\in L^{*}$, a matrix $P\in\mathbb{C}^{(+)1\times n}$ and three sets P^{+},P^{-},P^{0} with the above properties. This means that all three of the above structures can be defined in terms of each other. We will heavily use this property in the following lemmas.

Lemma 13.7
Let $p,q\in L^{*}$, be two min-terms and $P^{0},P^{+},P^{-},Q^{0},Q^{+},Q^{-}$, sets of integers such that:
$P^{0},P^{+},P^{-}\subseteq\{1,2,\ldots,n\}$, $P^{+}\cap P^{-}=\varnothing$, $P^{+}\cap P^{0}=\varnothing$, $P^{-}\cap P^{0}=\varnothing$, $P^{0}\cup P^{+}\cup P^{-}=\{1,2,\ldots,n\}$,
$Q^{0},Q^{+},Q^{-}\subseteq\{1,2,\ldots,n\}$, $Q^{+}\cap Q^{-}=\varnothing$, $Q^{+}\cap Q^{0}=\varnothing$, $Q^{-}\cap Q^{0}=\varnothing$, $Q^{0}\cup Q^{+}\cup Q^{-}=\{1,2,\ldots,n\}$
and $P=\sum_{j\in P^{+}}A_j+\sum_{j\in P^{-}}A_j(i)=TT(p)$, $Q=\sum_{j\in Q^{+}}A_j+\sum_{j\in Q^{-}}A_j(i)=TT(q)$.
Moreover, we define the set of atoms: $S=\{\alpha_j|j\in(P^{+}\cap Q^{-})\cup(P^{-}\cap Q^{+})\}$.
Then the least $k\in\mathbb{N}$ for which $mod(G^{k}(p)\wedge q)\neq\varnothing$ is $k=|S|$.
Moreover, $mod(res_S(p)\wedge q)\neq\varnothing$ and S is the only set of size k with this property.

Finally, $mod(G^k(p)\wedge q)=K_1\times K_2\times\ldots\times K_n$ where for all $j\in\{1,2,\ldots,n\}$:

- $K_j=\{1\}$, iff $j\in Q^+\cup(Q^0\cap P^+)$
- $K_j=\{0\}$, iff $j\in Q^-\cup(Q^0\cap P^-)$
- $K_j=\{0,1\}$, iff $j\in Q^0\cap P^0$

Proof

Let us suppose that there is $k<|S|\leq n$ such that $mod(G^k(p)\wedge q)\neq\varnothing$.

By the definition of sets P^+,P^- (lemma 13.6), and by Dalal's Theorem 5.6 ([3]) it follows that for this k:

$$G^k(p)=\bigvee_{U\subseteq\{a_1,a_2,\ldots,a_n\},|U|=k} res_U(p)\Rightarrow G^k(p)\wedge q=\bigvee_{U\subseteq\{a_1,a_2,\ldots,a_n\},|U|=k}(res_U(p)\wedge q)\Rightarrow$$

$$\Rightarrow\bigcup_{U\subseteq\{a_1,a_2,\ldots,a_n\},|U|=k}\text{mod}(res_U(p)\wedge q)\neq\varnothing.$$

So, there is at least one set U, with $|U|<|S|$, such that $mod(res_U(p)\wedge q)\neq\varnothing$.

However, for any such set, there exists an atom $\alpha_j\in S$ such that $\alpha_j\notin U$.

At first, let us suppose that $j\in P^+\cap Q^-$. Then, the atom α_j appears as a positive atom in p and as a negative atom in q (lemma 13.6). Moreover, by theorem 5.4 ([3]), it follows that the models of $res_U(p)$ differ from the models of p in the truth-values of the atoms in U, at most. Therefore, for all interpretations $I=(b_1,b_2,\ldots,b_n)\in mod(res_U(p))$, it must hold that $b_j=1$, because α_j appears as a positive atom in p ($\alpha_j\in P^+$) and $\alpha_j\notin U$. On the other hand, for all interpretations $I'=(c_1,c_2,\ldots,c_n)\in mod(q)$, it must hold that $c_j=0$, as α_j appears as a negative atom in q ($\alpha_j\in Q^-$). So, there is no interpretation $I\in mod(res_U(p))$ such that $I\in mod(q)$, thus $mod(res_U(p)\wedge q)=\varnothing$, contradiction. Therefore, for all $j\in P^+\cap Q^-$ we must have $\alpha_j\in U$.

In a similar fashion, we can prove that for all $j\in P^-\cap Q^+$ we must have $\alpha_j\in U$.

Combining the above, we have that $\alpha_j\in S\Rightarrow\alpha_j\in U$, so for any U such that $mod(res_U(p)\wedge q)\neq\varnothing$ it must hold that $U\supseteq S$ and, consequently, $|U|\geq|S|$. This means that for all $k\in\mathbb{N}$ for which $mod(G^k(p)\wedge q)\neq\varnothing$ it holds that $k\geq|S|$.

We will now prove that $mod(res_S(p)\wedge q)\neq\varnothing$.

It is known that $mod(p)=I_1\times I_2\times\ldots\times I_n$, where for all $j\in\{1,2,\ldots,n\}$:

- $I_j=\{0\}$ iff α_j appears as a negative atom in p, ie iff $j\in P^-$
- $I_j=\{1\}$ iff α_j appears as a positive atom in p, ie iff $j\in P^+$
- $I_j=\{0,1\}$ iff α_j does not appear in p, ie iff $j\in P^0$

Moreover, $mod(res_S(p))$ contains all the interpretations of mod(p) plus all the interpretations of I(n) which differ from some interpretation of mod(p) in the truth-value(s) of one or more atom(s) from S.

Therefore $mod(res_S(p))=J_1\times J_2\times\ldots\times J_n$, where for all $j\in\{1,2,\ldots,n\}$:

- $J_j=\{0,1\}$ iff $j\in(P^+\cap Q^-)\cup(P^-\cap Q^+)\Leftrightarrow\alpha_j\in S$
- $J_j=I_j$ iff $j\notin(P^+\cap Q^-)\cup(P^-\cap Q^+)\Leftrightarrow\alpha_j\notin S$

Similarly, $mod(q)=L_1\times L_2\times\ldots\times L_n$, where:

- $L_j=\{0\}$ iff α_j appears as a negative atom in q, ie iff $j\in Q^-$
- $L_j=\{1\}$ iff α_j appears as a positive atom in q, ie iff $j\in Q^+$
- $L_j=\{0,1\}$ iff α_j does not appear in q, ie iff $j\in Q^0$

It follows that: $mod(res_S(p)\wedge q)=mod(res_S(p))\cap mod(q)=$

$=(J_1\times J_2\times\ldots\times J_n)\cap(L_1\times L_2\times\ldots\times L_n)=(J_1\cap L_1)\times(J_2\cap L_2)\times\ldots\times(J_n\cap L_n)$.

We set $K_j=J_j\cap L_j$, for all $j\in\{1,2,\ldots,n\}$. Then, for all $j\in\{1,2,\ldots,n\}$ we have:

- if $j\in Q^+$, then $L_j=\{1\}$ and:
    - if $j\in P^+$, then $J_j=\{1\}\Rightarrow K_j=J_j\cap L_j=\{1\}$.
    - if $j\in P^-$, then $J_j=\{0,1\}\Rightarrow K_j=J_j\cap L_j=\{1\}$.

- if $j\in P^0$, then $J_j=\{0,1\}\Rightarrow K_j=J_j\cap L_j=\{1\}$.

- if $j\in Q^-$, then $L_j=\{0\}$ and:
  - if $j\in P^+$, then $J_j=\{0,1\}\Rightarrow K_j=J_j\cap L_j=\{0\}$.
  - if $j\in P^-$, then $J_j=\{0\}\Rightarrow K_j=J_j\cap L_j=\{0\}$.
  - if $j\in P^0$, then $J_j=\{0,1\}\Rightarrow K_j=J_j\cap L_j=\{0\}$.
- if $j\in Q^0$, then $L_j=\{0,1\}$ and:
  - if $j\in P^+$, then $J_j=\{1\}\Rightarrow K_j=J_j\cap L_j=\{1\}$.
  - if $j\in P^-$, then $J_j=\{0\}\Rightarrow K_j=J_j\cap L_j=\{0\}$.
  - if $j\in P^0$, then $J_j=\{0,1\}\Rightarrow K_j=J_j\cap L_j=\{0,1\}$.

By summarizing the above 9 cases, and for all $j\in\{1,2,\ldots,n\}$, we have that:

- $K_j=\{1\}$, iff $j\in Q^+\cup(Q^0\cap P^+)$
- $K_j=\{0\}$, iff $j\in Q^-\cup(Q^0\cap P^-)$
- $K_j=\{0,1\}$, iff $j\in Q^0\cap P^0$

Thus: $\mathrm{mod}(res_S(p)\wedge q)=K_1\times K_2\times\ldots\times K_n\neq\varnothing$.

In conjunction with the fact that the least $k\in\mathbb{N}$ for which $\mathrm{mod}(G^k(p)\wedge q)\neq\varnothing$ has the property $k\geq|S|$, the above equation gives that for this k it holds that: $k=|S|$. Moreover, since for any set U of atoms such that $\mathrm{mod}(res_U(p)\wedge q)\neq\varnothing$ it must hold that $U\supseteq S$, and given that $\mathrm{mod}(res_S(p)\wedge q)\neq\varnothing$, we have as a consequence that the only set U of atoms of size $k=|S|$ with the property that $\mathrm{mod}(res_U(p)\wedge q)\neq\varnothing$ is S.

Finally, by the above results and the equation:

$G^k(p)\wedge q=\bigcup_{U\subseteq\{a_1,a_2,\ldots,a_n\},|U|=k}\mathrm{mod}(res_U(p)\wedge q)$, it follows that:

$\mathrm{mod}(G^k(p)\wedge q)=\mathrm{mod}(res_S(p)\wedge q)=K_1\times K_2\times\ldots\times K_n$, where K_j as above, and this completes the proof.

Lemma 13.8

Let P^0,P^+,P^-,Q^0,Q^+,Q^-, sets of integers such that:

$P^0,P^+,P^-\subseteq\{1,2,\ldots,n\}$, $P^+\cap P^-=\varnothing$, $P^+\cap P^0=\varnothing$, $P^-\cap P^0=\varnothing$, $P^0\cup P^+\cup P^-=\{1,2,\ldots,n\}$ and:

$Q^0,Q^+,Q^-\subseteq\{1,2,\ldots,n\}$, $Q^+\cap Q^-=\varnothing$, $Q^+\cap Q^0=\varnothing$, $Q^-\cap Q^0=\varnothing$, $Q^0\cup Q^+\cup Q^-=\{1,2,\ldots,n\}$.

Let $P,Q\in\mathbb{C}^{(+)1\times n}$, such that:

$P=\sum_{j\in P^+}A_j+\sum_{j\in P^-}A_j(i)$ and:

$Q=2\cdot(\sum_{j\in\ ^{+}}A_j+\sum_{j\in\ ^{-}}A_j(i))$ and $R=P\bullet Q\in\mathbb{C}^{(+)1\times n}$.

Moreover, let the RE and RL functions be as defined above.

Then $RL(RE(R))=[k]$ where $k=n-|(P^+\cap Q^-)\cup(P^-\cap Q^+)|$.

Proof

It is obvious from the definition of P^+,P^-,P^0 and P that the elements of P are from the set: $\{0,1,i\}$ and that the element of the j-th column p_j is equal to:

- 0 iff $j\in P^0$
- 1 iff $j\in P^+$
- i iff $j\in P^-$

Similarly, the elements of Q are from the set: $\{0,2,2i\}$ and the element of the j-th column q_j is equal to:

- 0 iff $j\in Q^0$
- 2 iff $j\in Q^+$
- 2i iff $j\in Q^-$

By the definition of revision, we have that: R=P•Q=P+Q, thus the elements of R are taken from the set:
S={0,2,2i,1,3,1+2i,i,2+i,3i}.
Let $R=[r_1\ r_2\ \dots\ r_n]=[p_1+q_1\ p_2+q_2\ \dots\ p_n+q_n]$.
By the definition of RL,RE, we have that RL(RE(R))=[k] for some $k\in\mathbb{R}^{(+)}$.
For all j∈{1,2,…,n}, we have that $r_j\in S$, thus $r_j\in\mathbb{C}^{(+)}$, and $Re(r_j)\leq 1$ or $Im(r_j)\leq 1$ (easily verified). Thus by lemma 13.2, for the only line of R, we have that:
$k=RL_0(RE_0(r_1),RE_0(r_2),\dots,RE_0(r_n))=$
$=1-\min\{Re(r_1),Im(r_1)\}+1-\min\{Re(r_2),Im(r_2)\}+\dots+1-\min\{Re(r_n),Im(r_n)\}=$
$=n-(\min\{Re(r_1),Im(r_1)\}+\min\{Re(r_2),Im(r_2)\}+\dots+\min\{Re(r_n),Im(r_n)\})$ (1).
By looking at the set S, we can easily deduct that for each j∈{1,2,…,n} we have that:
$\min\{Re(r_j),Im(r_j)\}=1$ iff $r_j=1+2i$ or $r_j=2+i$ and
$\min\{Re(r_j),Im(r_j)\}=0$ otherwise.
Equivalently:
$\min\{Re(r_j),Im(r_j)\}=1$ iff $p_j=1$, $q_j=2i$ or $p_j=i$, $q_j=2$ which is true iff:
$j\in P^+$ and $j\in Q^-$ or $j\in P^-$ and $j\in Q^+\Leftrightarrow j\in(P^+\cap Q^-)\cup(P^-\cap Q^+)$.
Thus:
$\min\{Re(r_1),Im(r_1)\}+\min\{Re(r_2),Im(r_2)\}+\dots+\min\{Re(r_n),Im(r_n)\}=|(P^+\cap Q^-)\cup(P^-\cap Q^+)|$.
Applying the above result to (1) we get:
$k=n-|(P^+\cap Q^-)\cup(P^-\cap Q^+)|$, and the proof is complete.

The following proposition offers a way to emulate Dalal's algorithm under our framework. The Element Reliability function used is different from the one used in [7] for the respective proposition, but lemma 13.2 implies that the selection of the RF factors makes the difference unimportant. Specifically:

Proposition 13.1

Let $p,q\in L^*$ be two satisfiable propositional expressions in DNF and let r be the revision of p with q under Dalal's algorithm ($r=p\bullet^D q$). Moreover, let $P\in\mathbb{C}^{(+)*\times n}$ the matrix related to p via the TT function, using an RF of 1 for all atoms (P=TT(p)), $Q\in\mathbb{C}^{(+)*\times n}$ the matrix related to q via the TT function, using an RF of 2 for all atoms ($Q=2\cdot TT(q)$) and $R\in\mathbb{C}^{(+)*\times n}$ the matrix resulting by the update of P with Q under our framework (R=P•Q). Finally, we define the functions RE, RL and LS as in the above lemmas (13.1, 13.3, 13.4).
Under these parameters, the resulting propositional expression (to be used in queries) is logically equivalent to the expression r as defined above, that is: QT(R)≅r.

Proof

At first, we will assume that both p and q are min-terms. Then $P,Q,R\in\mathbb{C}^{(+)1\times n}$.
We also assume the sets P^+,P^-,P^0,Q^+,Q^-,Q^0 as usual.
Let $S=(P^+\cap Q^-)\cup(P^-\cap Q^+)$.
By lemma 13.7, the least $k\in\mathbb{N}^*$ for which $mod(G^k(p)\wedge q)\neq\emptyset$ is $k=|S|$ and that for this k it holds that:
$mod(G^k(p)\wedge q)=I_1\times I_2\times\dots\times I_n$ where for all j∈{1,2,…,n}:

- $I_j=\{1\}$, iff $j\in Q^+\cup(Q^0\cap P^+)$
- $I_j=\{0\}$, iff $j\in Q^-\cup(Q^0\cap P^-)$
- $I_j=\{0,1\}$, iff $j\in Q^0\cap P^0$

Therefore, by the definition of the operator $\bullet^D$, it follows that:
$mod(r)=I_1\times I_2\times\dots\times I_n$, where I_j, j∈{1,2,...,n} as above.

Returning to our method, we have that $R\in\mathbb{C}^{(+)1\times n}$, so $B=RL(RE(R))\in\mathbb{R}^{(+)1\times 1}$.
Therefore: $\varnothing\subset LS(B)\subseteq\{1\}\Rightarrow LS(B)=\{1\}$.
So, $MS(R,LS(B))=MS(R,\{1\})=R$.
Let $P=[p_j]$, $Q=[q_j]$, $j=1,2,\ldots,n$.
Then: $R=[r_j]=[p_j+q_j]$.
Moreover $MN(R)=C=[c_j]$, such that $c_j=Re(r_j)–Im(r_j)$.
For all $j\in\{1,2,\ldots,n\}$ there exist the following possibilities:

- if $j\in Q^+$, then $q_j=2$ and:
  - if $j\in P^+$, then $p_j=1\Rightarrow r_j=3\Rightarrow c_j=3$.
  - if $j\in P^-$, then $p_j=i\Rightarrow r_j=2+i\Rightarrow c_j=1$.
  - if $j\in P^0$, then $p_j=0\Rightarrow r_j=2\Rightarrow c_j=2$.
- if $j\in Q^-$, then $q_j=2i$ and:
  - if $j\in P^+$, then $p_j=1\Rightarrow r_j=1+2i\Rightarrow c_j=–1$.
  - if $j\in P^-$, then $p_j=i\Rightarrow r_j=3i\Rightarrow c_j=–3$.
  - if $j\in P^0$, then $p_j=0\Rightarrow r_j=2i\Rightarrow c_j=–2$.
- if $j\in Q^0$, then $q_j=0$ and:
  - if $j\in P^+$, then $p_j=1\Rightarrow r_j=1\Rightarrow c_j=1$.
  - if $j\in P^-$, then $p_j=i\Rightarrow r_j=i\Rightarrow c_j=–1$.
  - if $j\in P^0$, then $p_j=0\Rightarrow r_j=0\Rightarrow c_j=0$.

In short:

- $c_j>0$ iff $j\in Q^+\cup(Q^0\cap P^+)$
- $c_j<0$ iff $j\in Q^-\cup(Q^0\cap P^-)$
- $c_j=0$ iff $j\in Q^0\cap P^0$

Thus, $mod(C)=J_1\times J_2\times\ldots\times J_n$, where for all $j\in\{1,2,\ldots,n\}$:

- $J_j=\{1\}$ iff $c_j=1\Leftrightarrow j\in Q^+\cup(Q^0\cap P^+)$
- $J_j=\{0\}$ iff $c_j=i\Leftrightarrow j\in Q^-\cup(Q^0\cap P^-)$
- $J_j=\{0,1\}$ iff $c_j=0\Leftrightarrow j\in Q^0\cap P^0$

By comparing J_j and I_j, it follows that mod(r)=mod(C).
However, $mod(QT(R))=mod(TTI(C))=mod(C)=mod(r)\Rightarrow QT(R)\cong r$.
In the general case, where p is the disjunction of m min-terms and q is the disjunction of l min-terms, we have that:
$p=p_1\vee p_2\vee\ldots\vee p_m$, $q=q_1\vee q_2\vee\ldots\vee q_l$, where each p_j, $j\in\{1,2,\ldots,m\}$ and each q_j, $j\in\{1,2,\ldots,l\}$ are min-terms.
Let k be the minimum integer for which $mod(G^k(p)\wedge q)\neq\varnothing$.
Then we have that:
$mod(r)=mod(p\bullet^D q)=mod(G^k(p)\wedge q)=mod(G^k(p_1\vee p_2\vee\ldots\vee p_m)\wedge(q_1\vee q_2\vee\ldots\vee q_l))=$
$= mod(G^k(p_1)\vee G^k(p_2)\vee\ldots\vee G^k(p_m))\wedge(q_1\vee q_2\vee\ldots\vee q_l))=$

$$= \bigcup_{i=1,j=1}^{i=m,j=l} \mathrm{mod}(G^k(p_i)\wedge q_j) \quad (1).$$

In the above relation, we notice that all p_i and q_j are min-terms. For each arbitrary selection of $i\in\{1,2,\ldots,m\}$,$j\in\{1,2,\ldots,l\}$, we can define the sets P^+,P^-,P^0,Q^+,Q^-,Q^0 as usual for the min-terms p_i, q_j respectively, as well as the quantity:
$t=|(P^+\cap Q^-)\cup(P^-\cap Q^+)|\in\mathbb{N}^*$.
There are three possibilities regarding t:

- t<k: In this case, we have that mod($G^t(p_i) \wedge q_j$)≠∅, by lemma 13.7, so the union: $\bigcup_{i=1,j=1}^{i=m,j=l} \text{mod}(G^t(p_i) \wedge q_j) \neq \varnothing$, which means that mod($G^t(p) \wedge q$)≠∅ for some t<k, which is a contradiction by definition of k.
- t>k: In this case, by lemma 13.7, t is the least integer for which the relation mod($G^t(p_i) \wedge q_j$)≠∅ holds, and since t>k we have that mod($G^k(p_i) \wedge q_j$)=∅.
- t=k: In this case, by lemma 13.7 again, mod($G^k(p_i) \wedge q_j$)=$I_1 \times I_2 \times \ldots \times I_n$ where for all d∈{1,2,…,n}:
    - I_d={1}, iff d∈$Q^+ \cup (Q^0 \cap P^+)$
    - I_d={0}, iff d∈$Q^- \cup (Q^0 \cap P^-)$
    - I_d={0,1}, iff d∈$Q^0 \cap P^0$

We will call T the set of (i,j) pairs for which the equation t=k holds. There exists at least one such pair, because if it didn't exist then we would have that: mod($G^k(p) \wedge q$)=∅, which is a contradiction by the definition of k; therefore T≠∅.
It follows by (1) that:

$$\text{mod}(r) = \bigcup_{(i,j)\in T} \text{mod}(G^k(p_i) \wedge q_j) \;\; (2).$$

For each (i,j) pair, we can define t_{ij} as above, and for all i,j it holds that: $t_{ij} \geq k$. Since the set T contains the (i,j) pairs for which $t_{ij}=k$, we can deduct that:
T={(i,j)|i∈{1,2,…,m},j∈{1,2,…,l},$t_{ij} \leq t_{i'j'}$, for all i′∈{1,2,…,m},j′∈{1,2,…,l}}.
On the other hand, it is obvious that:

$$P = TT(p) = \bigvee_{i=1}^{m} TT(p_i) = \Big|_{i=1}^{m} TT(p_i) \text{ and:}$$

$$Q = 2\cdot TT(q) = 2\cdot \bigvee_{j=1}^{l} TT(q_j) = 2\cdot \Big|_{j=1}^{l} TT(q_j) = \Big|_{j=1}^{l} (2\cdot TT(q_j)).$$

By the definition of the operator of revision, it follows that:

$$R = P \bullet Q = (\Big|_{i=1}^{m} TT(p_i)) \bullet (\Big|_{j=1}^{l} (2\cdot TT(q_j))) = \Big|_{i=1,j=1}^{i=m,j=l} (TT(p_i) + 2\cdot TT(q_j)) \;\; (3).$$

Each of the $TT(p_i)$, $TT(q_j)$ are matrices with only one line, so the above relation is the expression of R as the juxtaposition of its lines.
Let us suppose that R=[r_{dh}], d∈{1,2,…,m·l}, h∈{1,2,…,n}. Then:

$$RL(RE(R)) = RL(RE(\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{m\cdot l1} & r_{m\cdot l2} & \cdots & r_{m\cdot \text{ln}} \end{bmatrix})) =$$

$$= RL(\begin{bmatrix} RE_0(r_{11}) & RE_0(r_{12}) & \ldots & RE_0(r_{1n}) \\ RE_0(r_{21}) & RE_0(r_{22}) & \ldots & RE_0(r_{2n}) \\ \ldots & \ldots & \ldots & \ldots \\ RE_0(r_{m\cdot l1}) & RE_0(r_{m\cdot l2}) & \ldots & RE_0(r_{m\cdot \text{ln}}) \end{bmatrix}) =$$

$$= \Big|_{d=1}^{m\cdot l} RL_0([RE_0(r_{d1}) \quad RE_0(r_{d2}) \quad \ldots \quad RE_0(r_{dn})]).$$

By this relation we can deduct that RL(RE(R))∈$\mathbb{R}^{(+)m\cdot l\times 1}$, and each element of this matrix can be computed using lemma 13.8.
Let us arbitrarily select a d_0∈{1,2,…,m·l}. We also select:
i_0=((d_0–1) div m)+1 and j_0=((d_0–1) mod m)+1 (4).

It is trivial to verify using relation (3) that line d_0 of matrix R (denoted by $R^{(d_0)}$) is equal to:
$R^{(d_0)} = TT(p_{i_0}) + 2 \cdot TT(q_{j_0})$.
Moreover, by lemma 13.8: RL(RE($R^{(d_0)}$))=[n–$t_{i_0 j_0}$], where the t_{ij} are as defined above.
Thus:
$RL(RE(R)) = \mathop{|}\limits_{d=1}^{m \cdot l} [n - t_{i(d)j(d)}]$, where i(d),j(d) the indexes related to d, as specified by relations (4). Let B′=RL(RE(R)).
By the definition of LS it follows that:
LS(B′)={h∈{1,2,…,m·l}:∀h′∈{1,2,…,m·l} n–$t_{i(h)j(h)}$≥n–$t_{i(h')j(h')}$}=
={h∈{1,2,…,m·l}:∀h′∈{1,2,…,m·l} $t_{i(h)j(h)} \le t_{i(h')j(h')}$}.
By comparing the set T above:
T={(i,j)|i∈{1,2,…,m},j∈{1,2,…,l},$t_{ij} \le t_{i'j'}$, for all i′∈{1,2,…,m},j′∈{1,2,…,l}},
with the set LS(B′) we can deduct that:
h∈LS(B′)⇔(i(h),j(h))∈T.
It follows that:
$C' = MS(R, LS(B')) = \mathop{|}\limits_{h \in LS(B')} R^{(h)} = \mathop{|}\limits_{(i,j)\in T} (TT(p_i) + 2 \cdot TT(q_j))$ and that:
$D = MN(C') = MN(\mathop{|}\limits_{(i,j)\in T} (TT(p_i) + 2 \cdot TT(q_j))) = \mathop{|}\limits_{(i,j)\in T} MN(TT(p_i) + 2 \cdot TT(q_j))$.
Finally:
$\mathrm{mod}(QT(R)) = \mathrm{mod}(TTI(D)) = \mathrm{mod}(D) = \mathrm{mod}(\mathop{|}\limits_{(i,j)\in T} MN(TT(p_i) + 2 \cdot TT(q_j))) =$
$= \mathrm{mod}(\mathop{\vee}\limits_{(i,j)\in T} MN(TT(p_i) + 2 \cdot TT(q_j))) = \bigcup\limits_{(i,j)\in T} \mathrm{mod}(MN(TT(p_i) + 2 \cdot TT(q_j)))$ (5).
Let us arbitrarily select a pair (i,j)∈T, and let A_P=TT(p_i), A_Q=2·TT(q_j).
Both p_i and q_j are min-terms, and let s=$p_i \bullet^D q_j$. Then:
mod(s)=mod($G^{k'}(p_i) \wedge q_j$), for the least $k' \in \mathbb{N}^*$ for which mod($G^{k'}(p_i) \wedge q_j$)≠∅. By the definition of T, it follows that k′=t_{ij}=k.
Given that $A_P, A_Q \in \mathbb{C}^{(+)1\times n}$, we have that: QT($A_P \bullet A_Q$)=TTI(MN($A_P \bullet A_Q$)).
Moreover, the proposition has already been proven for min-terms, so:
mod(s)=mod(QT($A_P \bullet A_Q$))=mod(TTI(MN($A_P \bullet A_Q$)))=mod(MN(TT(p_i)+2·TT(q_j)))⇒
⇒mod($G^k(p_i) \wedge q_j$)=mod(MN(TT(p_i)+2·TT(q_j))).
The above equation holds for all (i,j)∈T, thus:
(2)⇒ $\mathrm{mod}(r) = \bigcup\limits_{(i,j)\in T} \mathrm{mod}(G^k(p_i) \wedge q_j) =$
$= \bigcup\limits_{(i,j)\in T} \mathrm{mod}(MN(TT(p_i) + 2 \cdot TT(q_j))) = \mathrm{mod}(QT(R))$ by (5).
Therefore: r≅QT(R), and the proof is complete.

Regarding the above proposition, some comments are in order. First of all, it must be stressed that Dalal’s algorithm is based on different semantics, therefore emulation cannot be perfect. In effect, the above proposition offers a way to interpret in logical terms (logical expressions) the result of the update under our semantics in order to be equivalent with the result of the same update under Dalal’s algorithm. Dalal’s algorithm does not take into account iterated revisions and neither do we. If we perform a second revision on the resulting matrix, the result may no longer be the same as Dalal's. In order to deal with iterated revisions we have to readjust the matrix representing the KB before each new revision. The method does not hold for propositions not in DNF, or for unsatisfiable propositions, at least under these parameters. The revelation of parameters under which the emulation covers those cases is an ongoing work.

14. Conclusions and Future Work

In this report, we presented a novel representation of propositional expressions and its theoretical foundation. We successfully applied this representation to the problem of belief revision. The introduction of the Reliability Factor (RF) and the quantitative nature of the table representation introduce an increased expressiveness in propositional logic, allowing the use of features not normally available. This could possibly speed up existing algorithms or provide solutions to existing problems in propositional logic or its applications. We believe that much more work needs to be done in order to fully exploit its capabilities.

Regarding belief revision, we provided a framework for addressing the problem based on the table transformation. The operations of contraction ([8, 14]), update and erasure ([12]) are included in our model in a very direct manner, without the need of additional operators. The result of the revision is a matrix and not a logical proposition; this fact opens up interesting possibilities regarding the interpretation of the result. Despite the fact that revision is performed in a clearly defined way, the result's interpretation can be heavily parameterized, in effect providing a different revision algorithm per parameter selection. This allows the definition of a whole class of algorithms, whose optimum member(s) may be application-dependent. Performing some tests with different parameters is an interesting area for future work.

Some theoretical work on these parameters may also be in line. For example, the search for the general conditions (parameters) under which the algorithm satisfies the AGM postulates is an ongoing effort, as well as the simulation of other algorithms, like those proposed in [16, 17] by Williams, in [15] by Nebel etc.

References

[1] Alchourron, Gärdenfors, Makinson, "On the Logic of Theory Change: Partial Meet Contraction and Revision Functions", 1985, The Journal of Symbolic Logic 50, pp.510-530.

[2] Dalal M., "Investigations Into a Theory of Knowledge Base Revision: Preliminary Report", In Proceedings of the Seventh National Conference on Artificial Intelligence, pp. 475-479, 1988.

[3] Dalal M., "Updates in Propositional Databases", Technical Report, DCS-TR-222, Department of Computer Science, Rutgers University, February 1988.

[4] Darwiche A., Pearl J., "On the Logic of Iterated Belief Revision", UCLA Cognitive Systems Laboratory, Technical Report (R-202) In R. Fagin (Ed.), Proceedings of the 1994 Conference on Theoretical Aspects of Reasoning about Knowledge (TARK '94), Pacific Grove, CA, 5-23, March 13-16, 1994. In Artificial Intelligence, 89(1-2):1--29, 1997.

[5] Delgrande J., "Considerations on a Similarity-Based Approach to Belief Change", 16th International Joint Conference on Artificial Intelligence (IJCAI'99), Stockholm, Sweden, August 1999, pp. 180-185.

[6] Fagin, Ullman, Vardi, "On the Semantics of Updates in Databases", 1983, Proceedings of Second ACM SIGACT-SIGMOD, Atlanta, pp. 352-365.
[7] Flouris G., Plexousakis D., "Belief Revision in Propositional Knowledge Bases", to appear in the 8th Panhellenic Conference on Informatics, November 2001, Nicosia, Cyprus.
[8] Gärdenfors P., "Belief Revision: An introduction", pp. 1-20 in Belief Revision, ed. by P. Gärdenfors, Cambridge University Press, 1992.
[9] Gärdenfors P., "The dynamics of knowledge and belief: Foundational vs. coherence theories", Revue Internationale de Philosophie 44, pp. 24-46. Reprinted in Knowledge, Belief and Strategic Interaction, ed. by Bicchieri and M. L. dalla Chiara, Cambridge University Press, Cambridge, 1992, pp. 377-396.
[10] Hansson Sven Ove, "In Defense of Base Contraction", Synthese 91, pp. 239-245, 1992.
[11] Katsuno, Mendelzon, "Propositional Knowledge Base Revision and Minimal Change", KRR-TR-90-3, March 1990, Technical Reports on Knowledge Representation and Reasoning, University of Toronto.
[12] Katsuno, Mendelzon, "On the Difference Between Updating a Logical Database and Revising it", in Belief Revision, P. Gardenfors, Cambridge Press, 1992.
[13] Liberatore P., "The Complexity of Iterated Belief Revision", In Proceedings of the Sixth International Conference on Database Theory (ICDT'97), pages 276-290, 1997.
[14] Makinson, "How to Give it up: A Survey of some Formal Aspects of the Logic of Theory Change", 1985, Synthese 62, pp. 347-363.
[15] Nebel B., "A Knowledge Level Analysis of Belief Revision", In Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning, pp. 301-311, 1989.
[16] Williams, "Anytime Belief Revision", 1997, International Joint Conference on Artificial Intelligence, 1997.
[17] Williams Mary-Anne, Williams David, "A Belief Revision System for the World Wide Web", Web: http://u2.newcastle.edu.au/webworld/ai-internet.html.