DSM relational structures extended with fuzzy sets of higher types

Patrik Eklund[1], Magnus Löfstrand[2], Satyam Paul[2], Mehdi Goodarzi[2]

[1]Umeå University, Department of Computing Science, Umeå, Sweden
[2]Örebro University, School of Science and Technology, Örebro, Sweden

Abstract: In this paper we show how relational representations of design structure matrices (DSM), on the one hand, enables to describe domain dependencies and connections as relational composition, and, on the other hand, invites to using a variety of algebraic structures for the sets of qualifications attached with non-binary matrices. Particularly, we use fuzzy sets of higher types to model qualifications in many-valued DSMs where compositional techniques allow for extending the use of fuzzy sets of higher types also in the setting of multidomain matrices (MDM). We further show how clustered domains can be embedded as modelled within powersets of domains, thus providing a further justification for adopting the relational view of DSMs, particularly as the qualification space needs to support folding and unfolding across hierarchies in clustered domains. Our case study is drawn from scenarios involving maintenance of equipment in mineral mining.

Keywords: many-valued relation, powerset, relational composition, fuzzy set of higher type, mineral mining

1 Introduction

The design structure matrix (DSM), originally presented by Steward (Steward, 1967 and 1981), is a many-valued relation in the sense that positions in the matrix are qualified sometimes simply in a two-valued fashion, sometimes using a dedicated set of qualifications (Eppinger and Browning, 2012). DSMs have been used in a variety of applications (Browning, 2016).

Whereas in probability theory a notion of 'probability of probability' makes less sense, or even no sense at all, in fuzzy set theory the notion of 'uncertainty of uncertainty' not just makes sense but can be shown to be useful in applications. The notion was formalized by Lotfi Zadeh in his 1973 technical report (Zadeh, 1973), providing the notion of *fuzzy sets of type 2*, and later in 1975 published in the trilogy on his papers titled *The Concept of a Linguistic Variable and its Application to Approximate Reasoning - I, II, III* (Zadeh, 1975a, 1975b, 1975c).

In unravelling the formal content of the notion of fuzzy sets of higher types, we need to understand some subtleties related with the notion of *sets of sets*, i.e., the notion of powerset in set theory. More generally, we need to make use of set theoretic constructions creating new sets from old ones, where the powerset technique is one such construction. Set operations, such as union, intersection and complement create new sets from old ones, and are frequently used as basic set-theoretic tools, where the set of functions between sets is more rarely viewed as a constructor of new sets. Sets of functions viewed as generalized powersets are frequently used in our constructions.

Structure modelling in this paper is inspired by project work done in cooperation with mining companies and their equipment suppliers. One focus has been predictive maintenance for drill rigs based on operation and failure data analytics. Information on configuration of maintenance teams has also been included, and such information is often delivered by both mine operator and equipment supplier, with respective commitments further embraced and specified by and within service contracts.

2 Design structure matrices as relational structures

A *binary* DSM is based on a finite set X of labels, X=$\{e_1, \dots, e_n\}$, where some pairs (e_i, e_j) of labels are defined as related, saying that e_i and e_j *interact*, so that the binary DSM essentially is a relation $R \subseteq X \times X$. Interaction is directed, i.e., DSMs are usually not symmetric. The relation for a binary DSM can equivalently be written as a function $f_R : X \times X \to \{0,1\}$, defined by $f_R(e_i, e_j) = 1$ if and only if e_i and e_j are related, $e_i R e_j$, which in DSM context is understood as e_j "receiving information" from e_i, or, as often said, "e_i provides *output* that is *input* to e_j".

A non-binary DSM is then similarly representable by a function $f_R : X \times X \to Y$, where Y is a set of attributes, features or qualifications of various type, the set Y often including algebraic properties, e.g., like being an ordered set of qualifications and enabling operation with attributes. The algebraic properties of Y are ideally selected for the purpose not just of modelling attributes, features and qualifications but also to support compliance when combining and moving them within and between domains.

A DSM is often graphically shown as a table with rows and columns, where the elements of X are the *names*, not the *numbers*, of the rows and the columns. A DSM therefore invites to be viewed as a *square matrix* as it is understood in linear algebra, formally representing a square array of numbers or other types of values, where a value with notation a_{ij} is the value in jth column of the ith row. In this view, a binary or non-binary DSM, with n elements in its domain X, becomes a n-by-n matrix, which then invites to making use of calculations and properties within linear algebra for matrix manipulation. Building upon the relational view of a DSM will similarly invite to making use of operations within relational algebra for manipulation of two-valued or many-valued relations $f : X \times X \to Y$, where a DSM indeed is called *binary* whenever the set Y is two-pointed.

When graphically depicted as a table of cells, a binary DSM may use a symbol like the check mark "✓", or similar, to indicate that two elements are related. A cell remaining empty is symbolized e.g. using "∅" or "0", or similar, whenever two elements are not related. In such cases, Y may appear as $\{0,1\}$, $\{\emptyset, \checkmark\}$, or $\{0, \checkmark\}$, or similar, to formally provide a symbolic notation for the elements in Y. The understanding here is that Y can basically be any type of a set, structured or unstructured, and it is important not to create difficulties e.g. for MDMs if the choice and understanding of Y is different for different DSM domains. An MDM in such a case needs to relate, or "map", elements to elements between different domains, and may also need to combine different types of values from different Ys. If elements in Y are numbers, or symbols viewed as numbers, it invites to computing with these symbols as numbers. If elements in Y are treated as non-numerical symbols, and Y possesses algebraic structure, rather than possessing *only linear* algebraic structure, e.g., with elements in Y being "comparable", or as appearing in form of "membership grades", or similar, it invites to using algebraic operations to compute with these non-numerical symbols.

In this paper we focus on the interpretation of values in Y being membership values, numeric or symbolic, and thereby view a DSM as a "fuzzy relation" $f : X \times X \to Y$, discussed in more detail in Section 3. Transformations across domains, and as involving subsystems of various kind, which requires to deal with powerset structures, are discussed in Section 4.

In this paper we do not dwell on sequencing and as supported by techniques where e.g. *powers of adjacency* are used, but we may note that powers of binary matrices using Boolean operations meet ("AND") and join ("OR"), instead of addition and multiplication, is for relations exactly mirrored by *transitive closure*, also used when dealing with powers of graphs.

3 Fuzzy sets of higher types

Given sets X and Y, the power Y^X is the set of all functions $f : X \to Y$. The powerset of X, PX, i.e., the set of all subsets of X, is a special case of the power Y^X with Y being a two-pointed set, say $Y = \{0,1\}$. A subset $A \subseteq X$ thus corresponds to a function $f_A : X \to \{0,1\}$, called the *characteristic function*, where

$$f_A(x) = \begin{cases} 1, if\ x \in A \\ 0, if\ x \notin A \end{cases}$$

If $Y = [0,1]$, the unit interval of real numbers, then the power Y^X is the set of all fuzzy sets on X, and these are in (Zadeh, 1973) called *fuzzy sets of type 1*. Zadeh's famous notation $\mu_A : X \to [0,1]$ for fuzzy sets (of type 1) may sometimes appear confusing, as A is not a set, but merely an index or a name for the fuzzy set. Instead of saying "A is a fuzzy subset" and providing the notion $\mu_A : X \to [0,1]$ for it, we may say "μ is a fuzzy set", with the notation $\mu : X \to [0,1]$.

The situation with fuzzy sets of higher types, given Zadeh's introduction of the notion, gives room for different interpretations, and thereby also using different notations and definitions. Before discussing various options, let us first observe that the unit interval $[0,1]$, the 2-pointed set $\{0,1\}$, as well as many-valued sets of membership or uncertainty values, are all viewed as ordered sets. More precisely, in fuzzy set theory, the minimum requirement is that these sets are *complete lattices*. Goguen (Goguen, 1967) generalized fuzzy sets over the unit interval to so called L-fuzzy sets, where L is a *complete distributive lattice*. We should note that distributivity is an additional algebraic condition on lattices, where the meet and join operations distribute over each other. It is a fairly strong condition. The unit interval with the minimum and maximum operations for numbers makes it a complete distributive lattice and the 2-pointed set as a conventional Boolean algebra is also a complete distributive lattice. Applications may use a 3-valued logic, as seen e.g. in (Reed and Löfstrand, 2016), or the underlying logic make use 5-6 or even more values in its logic.

We may weaken the distributivity condition to look at meet distributivity only and from there we usually extend to *quantales* (Eklund et al., 2018a) which are complete lattices with a semigroup operation that distributes over join. Quantales are very flexible and appealing in practical applications, as we have seen in the use in many-valued extensions within industrial applications (Eklund and Löfstrand, 2016) and similarly in extensions of DSM (Eklund et al., 2017b and 2019). A quantale enables to manage computations with unknown values, where *unital quantales* are suitable for such applications. Logical representations, e.g. as in (Eklund et al., 2014), come into play, involving a wide variety of approaches using many-valued logic.

We may write $\mathbf{2}$ for the 2-pointed set $\{0,1\}$, and then view $\mathbf{2}$ as a complete lattice. In this case, the $\mathbf{2}$-fuzzy sets over a set X are precisely the ordinary subsets of X, represented by their characteristic functions. Then $\mathbf{2}^X$ is the same as the powerset PX. For the set of L-fuzzy sets over X, L^X, we may use the notation LX for the set of all L-fuzzy sets over X, similarly to using PX, or $\mathbf{2}X$, for the two-valued situation.

The fundamental, yet simple, idea of Zadeh's *fuzzy sets of type 2*, for uncertainties of graded memberships, is that $\mu(x)$ in type 2 context is not a value in the unit interval but a fuzzy set. Zadeh (Zadeh, 1973) didn't formally define fuzzy sets of type 2 but wrote that $\mu_A(x)$, for a fuzzy set μ_A of type 2, is "allowed to be a fuzzy set of [0,1]". In the vocabulary and notation of (Zadeh, 1973), a "fuzzy sets of X" is a function $\mu : X \to [0,1]$, so a "fuzzy set of [0,1]" is then a function $\mu : [0,1] \to [0,1]$, i.e., the definition of a fuzzy set of type 2 in (Zadeh, 1973) is clearly

$$\mu : X \to [0,1]^{[0,1]}$$

without using the "A" index. In (Mizumoto and Tanaka, 1976), the recursive nature of the definition, indicated in (Zadeh, 1973), is formally presented as *fuzzy sets of type n* being of the form

$$\mu_A : X \to [0,1]^{J_1^{\cdot^{\cdot^{J_{n-1}}}}}$$

where it is assumed that J_i, $i = 1, \ldots, n-1$ are finite sets, with the remark that "the algebraic properties of fuzzy grades in J discussed later [in the paper] are satisfied in the case where J is continuous".

In the case of sets and functions, e.g. for sets X, Y and Z, it is well-known, referred to as the property of *cartesian closedness*, that

$$(X^Y)^Z \cong X^{Y \times Z}$$

When $[0,1], J_1, \ldots, J_{n-1}$ are viewed as sets, and

$$[0,1]^{J_1^{\cdot^{\cdot^{J_{n-1}}}}}$$

is understood as

$$(\ldots([0,1]^{J_1})^{\cdot^{\cdot}})^{J_{n-1}}$$

and not as

$$[0,1]^{\left(J_1^{\cdot^{\cdot^{J_{n-1}}}}\right)}$$

we will, by cartesian closedness, then arrive at

$$[0,1]^{J_1^{\cdot^{\cdot^{J_{n-1}}}}} \cong [0,1]^{J_1 \times \ldots \times J_{n-1}}$$

This means that, given $x \in X$, $\mu(x)$, in the sense of (Zadeh, 1973), is a function of the form

$$\mu(x) : [0,1]^{n-1} \to [0,1]$$

It may, given sets X and Y, be remarked that $X \times Y$ logically relates to the independency "$X\ and\ Y$" whereas X^Y logically relates to the directed dependency "$from\ Y\ to\ X$". Whenever X and Y are equipped with lattice structures and even extended to be quantales, which may be commutative or non-commutative, $X \times Y$ and X^Y as resulting lattices and quantales will obviously embrace quite different properties (Eklund et al., 2017a). In (Zadeh, 1973) and (Mizumoto and Tanaka, 1976), on their announcement of "type n", the discussion were intuitive and somewhat less formal, in these directions.

From application point of view, we see how the uncertainty $\mu(x)$ of a point $x \in X$, represented by a fuzzy set of type 2, is algebraically a unary, or 1-ary, operator

$$\mu(x) : [0,1] \to [0,1]$$

From the unary situation we can extend e.g. to fuzzy sets of type 3 as binary, or 2-ary, operators

$$\mu(x) : [0,1]^2 \to [0,1]$$

Generalizing the unit interval as an algebra of truth values to other suitable algebras, like to a quantale Q, provides fuzzy sets of type 3 in the form

$$\mu(x) : Q^2 \to Q$$

From fuzzy sets of type 2 and type 3 we may then extend to fuzzy sets of type n as

$$\mu(x) : [0,1]^{n-1} \to [0,1]$$

or, using quantales $(Q,*)$, or other algebraic structures, similarly so that

$$\mu_*(x) : Q^{n-1} \to Q$$

is given, e.g., by

$$\mu_*(x)(q_1, \ldots, q_{n-1}) = q_1 * \ldots * q_{n-1}$$

Here we note, generally speaking, that the number of quantales available for defining such a μ using the operator is quite large as the number of elements in the lattice grows from just a few to many. If Q is 3-pointed, there is just the 3-chain as the underlying lattice, and there are 12 quantales over that 3-chain, many of those 12 quantales being trivial and not fit for use in practice. For 4 points, there are two lattices, the chain and the "diamond", with a total of 129 quantales, many of which fit well for chosen application scenarios. For Q being a 5-point set, there are 5 different lattices, and a total of 1852 quantales. For a 6-point set there are 15 lattices and totally 33 391 quantales (Shamsgovara et al., 2019), and so on. All quantales up to 9 points have been listed and constructed (Shamsgovara, 2023).

Quantales enable to use finite sets of uncertainty values yet having a rich set of algebraic properties and alternatives available for modelling situations where uncertainty calculations cannot rely on arithmetic and unit intervals only. Important also to note is that some quantale properties allow for creation of new operators, like when a * as a "conjunction" is able to define implication relations, i.e., a causality relation is instantly available.

From DSM point of view, and in the relational situation

$$\delta : X \times X \to Q$$

representing a fuzzy relation of type 1, we then extend DSMs to using fuzzy relations of type n with

$$\delta(x_1, x_2) : Q^{n-1} \to Q$$

The general extension of DSM with uncertainty modelling, making use of fuzzy sets of higher types, is now as follows. With a number of domains, like X_{Co}, X_{Pe} and X_{Ac} for *components*, *people* and *activities*, for each domain X_d we may have fuzzy sets of various types, ranging from using fuzzy sets of type 1, $\mu : X_d \to [0,1]$, to using higher types with attributes residing in various algebraic structures. We may have uncertain relations either within one and the same domain (DSM), or across two or more domains (MDM), and even combine domain crossover involving external domains of interest. In such cases we are dealing with multidimensional uncertain relations, i.e., multidomain uncertainties, of the form

$$\delta : X_{d_1} \times \ldots \times X_{d_m} \to [0,1]^{J_1^{\cdot^{\cdot^{\cdot J_{n-1}}}}}$$

using traditional fuzzy sets of higher types in the sense of (Zadeh, 1973) and (Mizumoto et al., 1976), or, more generally,

$$\delta : X_{d_1} \times \ldots \times X_{d_m} \to Q^{Q^{n-1}}$$

using a wider range of algebraic structures. Combining single domain uncertainties

$$\mu : X_d \to Q^{Q^{n-1}}$$

with multidomain uncertainties using compositional techniques, provided by (Hisdal, 1981), enables to use a wide range of techniques for application development based on DSMs and MDMs using uncertainty modelling as provided by fuzzy sets of higher types, either in settings with the unit interval, or, more generally, using algebraic structures.

In (Eklund et al., 2019) we described a fuzzy set model for the "documentation of interaction between elements" as described in (Pimmler, 1994) and (Pimmler and Eppinger, 1994) using the ordered set $\{Detrimental, Undesired, Indifferent, Desired, Required\}$ as a scale set $\{-2, -1, 0, 1, 2\}$ of scores, respectively, for

Spatial, *Energy*, *Information* and *Materials*. For instance, in the case of *Spatial*, with scale value at *Desired* (+1), the understanding of the score is "physical adjacency is beneficial, but not absolutely necessary for functionality". "Documentation" in the sense of (Pimmler, 1994) involves *function* descriptions for two related elements in a domain. That relationship is then scored by a quadruple like $(+2,0,0,+2)$, i.e., the *Spatial* and *Materials* scores are $+2$, whereas the *Energy* and *Information* scores are 0.

We may now have $J_i = \{-2,-1,0,1,2\}$, $i = Spatial, Energy, Information, Materials$, so that a DSM is

$$f_R : X \times X \to J_{Spatial} \times J_{Energy} \times J_{Information} \times J_{Materials}$$

whenever we are *certain* that a relation between elements e_i and e_j can be provided as the quadruple value $f_R(e_i, e_j)$ in $J_{Spatial} \times J_{Energy} \times J_{Information} \times J_{Materials}$. Whenever we need to represent the relation between e_i and e_j to be *uncertain*, the DSM will then be provided in form of a fuzzy set of type 5

$$g_R : X \times X \to [0,1]^{J_{Spatial} \times J_{Energy} \times J_{Information} \times J_{Materials}}$$

More background and detail on fuzzy sets of higher types, as used in this paper for the purpose of DSM relational structures extended with fuzzy sets of higher types, and as related to their algebraic and logical nature, is found in (Eklund et al., 2025).

Fuzzy sets of type 2 have been used also in engineering applications with fuzzy control techniques (Karnik et al., 1999) and (Karnik and Mendel, 2001a), where "type 1 fuzzy control" is extended to enabling the use of fuzzy sets of type 2. In this case, the type 2 involvement in the end needs to be reduced to type 1, and type reduction techniques (Karnik and Mendel, 2001b) have been developed. Type 2 fuzzy controllers have also been used with dynamic modelling (Paul et al., 2022).

4 Many-valued powersets in multidomains

A binary DSM $f_R : X \times X \to \{0,1\}$ can equivalently be represented as $f_{R,pow} : X \to PX$, where PX is the powerset of X, i.e., the set of all subsets of X. Each $e \in X$ gives a subset $f_{R,pow}(e) \subseteq X$, so that $e' \in f_{R,pow}(e)$ if and only if eRe'. DSM representation by powersets is useful when dealing with partitioning of domains.

A DSM partitioning analysis, often based on sequencing, leads to a partitioning of a domain into subdomains. Subdomains are often called *clusters* in situations where "near-by" elements are seen becoming grouped. In such situations, where each subdomain X_i is a non-empty subset of X, the subsets $X_1, \dots, X_n$ may or may not form a strict partition of X, i.e., where distinct sets X_i and X_j, $i \neq j$, are disjoint and $\bigcup_{i=1}^{n} X_i = X$. A DSM partitioning indeed often will not fulfill the disjointness condition, but rather possess overlapping subdomains. A clustered DSM then also needs to provide valuations of the interactions between clusters, and these valuations are, in one way or another, inherited from the valuations of the interactions within the original DSM. Interactions between clusters need to "comply with" interactions between the elements in the original DSM, and interaction valuations can be computed in various ways, often determined by the application.

Suppose that the subdomains $X_1, \dots, X_n$ provide a strict or non-strict partitioning of the hosting domain X with its DSM relational function being $f : X \times X \to Y$. Each product $X_i \times X_i$ then forms the basis of a "cluster-DSM relation" with the corresponding cluster-DSM relational function being $f_i : X_i \times X_i \to Y$, where f_i is just the restriction of f. Further, the set of sets $\boldsymbol{X} = \{X_1, \dots, X_n\}$ establishes a base for forming a new "domain of sub-domains", with a new DSM $g_{\boldsymbol{X}} : \boldsymbol{X} \times \boldsymbol{X} \to Y$, which, since $\boldsymbol{X} \subseteq PX$, is a restriction of a relational function $g : PX \times PX \to Y$. Clearly, the definition of $g_{\boldsymbol{X}}$ should derive from the cluster-DSM functions f_i, so that the relational valuations defined by $g_{\boldsymbol{X}}$ are "inherited" from the cluster-DSMs. Inheritance will involve aggregation of values internal within cluster-DSMs, and such aggregations can again be provided in different ways.

Using these notations we also realize how we must not confuse $PX \times PX$ with $P(X \times X)$, even if subdomains as subsets of $X \times X$ might be useful in practice also when they appear in "non-square" form. Note indeed how the pair of sub-domains $(X_i, X_i) \in PX \times PX$ provides the "square" base product $X_i \times X_i \in P(X \times X)$ for the sub-domain $f_i : X_i \times X_i \to Y$. This shows how DSM squares and rectangles are constrained by a "rigidity of squares and rectangles" as the base sets of sub-domains must be drawn from $PX \times PX$ and not more "liberally" from $P(X \times X)$. The relational notations and formalism in this paper, however, shows that future development of formalities concerning DSM structures may seek to liberalize that rigidity. Clustering as restricted to $PX \times PX$ should also be understood as separated from clustering to produce subset in $P(X \times X)$ or fuzzy clusters in $L(X \times X)$. Many-valued clustering, c-means clustering, as a technique for "soft decomposition" seems not to have been considered within the DSM community, but may be worthwhile to consider, particularly as c-means techniques have been extended also to be used with fuzzy sets of higher types (Yang et al., 2021).

The reverse of clustering is unfolding of domain elements to become labels for sets of elements on a lower level in the domain structure. A label, or element e, in a set X of labels in a DSM, e.g. for components or subsystems in a system of systems, may in turn represent sets of labels e_i, i.e., elements in a lower level domains, so that e unfolds to a set $e = \{e_1, e_2, \dots\}$, where each e_i in turn may be a set of labels of elements in a unfolded level further down in the domain. Subdomains are in fashion broken down, i.e., labels in e_i can represent labels of sets, so that we arrive at a decomposition or hierarchy of "domains of domains" or "systems of systems", or "systems of systems of systems of …" (Eklund et al., 2018), and so on. System availability for functional products is a typical problem area where subdomain problems come into play (Löfstrand et al., 2011).

MDMs involving clustering within and across domains obviously requires a flexible yet algebraically strict relational modelling to understand inheritance of relational qualifications. Some basic techniques are required such as relational composition leading to relational equations and their solutions, for which much work has been done on *fuzzy relational equations*, with techniques to be found e.g. in (Sanchez, 1974), (Di Nola et al., 1989) and (Bartl, 2012).

On composition, in Appendix C of (Sosa, 2000), we note how "Algebraic Model" is a matrix based model, rather than a model based on algebra, so "algebraic" is used since the transformation technique is based on matrix algebra. Sosa's thesis suggests using a "Design Contribution Matrix" which relates elements in different domains. The domains considered are those for *Components* and *People*, i.e., for element in X_{Co} and X_{Pe}, where X_{Pe} is viewed as a set of *design teams*, with m elements, and X_{Co} represents a product decomposed into n *components*. Suppose an n-by-n DSM components matrix M_{Co} is given corresponding to a 2-valued relation $R_{Co} \subseteq X_{Co} \times X_{Co}$, together with a *domain mapping* as a n-by-m design contribution matrix C, where each team in the set X_{Pe} of m teams is contributing to the design of one or more components in X_{Co}. The (i,j)-element in the DSM m-by-m people matrix $M_{Pe} = C^T M_{Co} C$ is non-zero when and only when teams i and j contribute to the design of the same components. If R_{Pe} is the corresponding binarized *people* relation for the matrix $C^T M_{Co} C$, and $R_C \subseteq X_{Pe} \times X_{Co}$ is the corresponding relation for the contribution matrix C, with ${R_C}^{-1} \subseteq X_{Co} \times X_{Pe}$ being the inverse relation to R_C, then the relational composition

$$R_{Pe} = R_C \circ R_{Co} \circ {R_C}^{-1}$$

is the corresponding relational view of the equation $M_{Pe} = C^T M_{Co} C$. Matrix multiplication relies on arithmetic multiplication of numerical values, whereas the relational view provides "composition of connections", i.e., it computes with membership values. Given the purpose of our paper we will argue that the relational notation supports MDM extension quite well in our approach to using fuzzy sets of higher types for modelling of uncertainty with DSM and MDM.

Note that we do not have a function from X_{Co} to X_{Pe} nor a function from X_{Pe} to X_{Co}, but, given R_C, we have the functions $f_{R_C} : X_{Pe} \to PX_{Co}$ and $f_{{R_C}^{-1}} : X_{Co} \to PX_{Pe}$. Thus, the *domain mapping matrices* (DMM) are not mappings in the sense of point-to-point functions, but rather as point-to-set *mappings*. These notations further show how the technique suggested in (Sosa, 2000) can immediately be extended to the many-valued context using $f_{R_C} : X_{Pe} \to QX_{Co}$ and $f_{{R_C}^{-1}} : X_{Co} \to QX_{Pe}$.

The relational notation of DSM and DMM matrices further invites to recall that the powerset of cartesian products of sets is not the same as the cartesian product of powersets, i.e., there is not an equality between $P(X \times X)$ and $PX \times PX$. This is reflected in DSM modelling in general as point-to-point functions between elements in different domains are not available and not even desired, whereas point-to-set functions as DMMs represent what the DSM matrices need in practice when providing applications in an MDM context.

Domain mappings as rectangular matrices are used also in (Danilovic and Browning, 2007), where "possible matrix multiplications" are mentioned but 'multiplications' are not presented explicitly, compared to suggestions provided in (Sosa, 2000). A "periodic table" of DSMs and DMMs matrices is informally depicted including the sizes of matrices indicating the order in which they can be multiplied. Example 3 with Figure 11 in (Danilovic and Browning, 2007), based on (Danilovic and Sandkull, 2005) with corresponding Figure 4, combines two DSMs, respectively, for 'product' and 'organization', connected with a dual-domain DMM, based on a study of interdependencies between organization departments, product subsystems and a multi-project domain. Interdependencies in all three matrices are quantified by values in $Y = \{0,1,2,3\}$, with "0" not written in cells to indicate "no interdependency".

5 Applications

In this section we use experience from failure analytics for particular drill rigs in open pit mining, and we briefly focus on product and organization domains, i.e., on drill rig subsystems and maintenance people and teams. Our listing of elements, not disclosing classified company information, is general but specific enough to demonstrate the capabilities of our structure modelling when using DSMs in contexts involving uncertain information.

We may first note that there are many types of drill rigs, including type “face”, “production” and “surface”. In the case e.g. of surface drill rigs, they are designed for various purposes, including open pit mining. Open pit mining drill rigs can be further classified based e.g. on drilling method and mineral types. Our example is drawn from operation of a drill rig for open pit mining of iron ore, including experience from cooperation with industrial partners.

The “Component” or “Product” domain is obviously a hierarchy or taxonomy no just for various types of drill rigs but indeed for mining products in general, including e.g. loaders, haulers and dumpers (LHD), where a mining company may use its fleet in several mines at different locations, and therefore faces a complex task of operation and maintenance in order to manage a fleet of such products and vehicles, possibly as supplied by different providers. Maintenance is often provided by the product supplier, but not always, whenever third parties are engaged for maintenance, and even for operation.

Similarly, the “People” or “Organization” domain is a hierarchy including regional and site managers, group and team leaders, with engineers and technicians of various types and with various levels of competences and experiences, sometimes all residing and being employed within one and the same producer and provider, sometimes being configured by a mix including employment in different companies.

Given the purpose of this paper, we simplify this situation by having our “domain of components” being a general description of subsystems defining a drill rig, involving just a few “components” rather than listing several hundreds of subsystems and components appearing within their respective indenture levels. We further simply the “domain of people” by having a domain consisting of a number of “teams”.

We may at this point remark that “re-composition” from lower or deeper levels of indentures to higher or shallow levels also in itself pre-defines a clustered structure, which in many case will not correlate with the clustered indenture resulting from a DSM sequencing, often being based on powers of adjacencies, when leaning on linear algebra, or transitive closures when viewing DSMs as relational structures, and thus leaning on relational algebra.

In demonstrating the use of our relational framework and the utility of fuzzy sets of higher types, we will deploy domains for *Component* and *People*, and we will use domain mappings between component and people domains. Our component domain is

$$X_{Co} = \{PowerTrain, \dots, DrillUnit, \dots, HydraulicSystem, \dots\}$$

The peoples’ domain is simply

$$X_{Pe} = \{Team_1, Team_2, \dots\}$$

The most simple situation is where we know relational data for one of the DSMs, say for *People*, $f : X_{Pe} \times X_{Pe} \to Y$, and we further have a DMM $k : X_{Co} \times X_{Pe} \to Y$, and where we want to compute the DSM $g : X_{Co} \times X_{Co} \to Y$ for *Component*. Let us further assume that Y possesses a type of many-valued structure with algebraic operations making it possible to compute composition of many-valued relations. For instance, Y might be a quantale Q, or a fuzzy set of higher type. In the case of a quantale Q, and a set X, we have QX as the Q-powerset of X, i.e., the set Q^X of all Q-fuzzy sets over X, as explained in Section 3.

Similar to functions $h : A \to B$, mapping points to points, being extendable to functions $Ph : PA \to PB$ on the powerset level, mapping sets to sets, the same technique can be applied for many-valued powersets. Details in the case of complete lattices are found in (Eklund et al., 2014).

The many-valued powerset representation of the DSMs and DMMs in a MDM is, using the *cartesian closedness* property explained in Section 2, then $k_Q : X_{Pe} \to QX_{Co}$ and $k^{[-1]}{}_Q : X_{Co} \to QX_{Pe}$ and $f_Q : X_{Pe} \to QX_{Pe}$, where $k^{[-1]}$ is the function connected with the inverse relation of the relation connected with k. The relational version and many-valued extension of the matrix algebraic technique in (Sosa, 2000) now gives $g_Q : X_{Co} \to QX_{Co}$ through the so called “Kleisli composition”

$$g_Q = \mu_Q \circ Q\mu_Q \circ QQk_Q \circ Qf_Q \circ k^{[-1]}{}_Q$$

where $\mu_Q : QQX \to QX$ is the *flattening function*. In the case of powers of sets, i.e., sets of sets of elements, the flattening simply picks out all elements in all sets in the sets of sets, and places these elements in a new set. In other words, and intuitively speaking, element residing in sets, where the sets in turn reside in a bag of all these sets, are shifted from residing in respective sets to become placed in one and the same (new) set, where all the elements mingle together. The flattening function in the many-valued case is a bit more complicated as membership values need to be considered, transferred and re-computed, but the intuition and principle is basically the same. See (Eklund et al., 2014) for detail.

We may also make use of intermediate DMMs in order to transfer relational knowledge from one domain to another. In our example case, we may replace the index set $\{Spatial, Energy, Information, Materials\}$ used in Section 3 e.g. by the index set

$$F = \{Mechanics, \dots, Hydraulics, \dots, Lubrication\}$$

with elements representing some general groups of function aspects relevant for components and subsystems of drill rigs, and for which dedicated teams provide maintenance especially for one or a few selected elements of the domain F. We then have the relations $t : X_{Pe} \to QF$, and $u : F \to QX_{Co}$, together with $k_Q : X_{Pe} \to QX_{Co}$, composed as

$$k_Q = \mu_Q \circ Qu \circ t$$

Here we immediately see how we arrive at problems related with *relational equations* involving many-valued information and relational qualifications. Note how we in this case indeed have a triad of domains and relational functions between all three pairs of domains, where the content of one domain may or may not be derived from the content of two other domains. The derived content, if derivable, may not be unique.

We may add a further intermediate DMM, e.g., for maintenance,

$$M = \{Preventive, \dots, Predictive, \dots, Improvement, \dots, Corrective, \dots, Scheduled, \dots, Operator\}$$

based on (EN 13306, 2017), with the corresponding relational function $v : M \to QX$, together with its inverse $v^{-1} : X \to QM$, relating maintenance elements to elements in one or more of the domains X_{Pe}, X_{Co}, and F. We then have four domains in a quad or tetrad, and we will need at least three out of six possible DMMs in order to formulate relational equations.

The EN 13306 standard for maintenance also defines five levels of maintenance, going from "simple actions carried out with minimal training" at Level 1 to "actions which imply a knowledge held by the manufacturer or a specialized company with industrial logistic support equipment" on Level 5. A domain relation $t : X_{Pe} \to QM$ will obviously need to consider such levels, which in turn requires that the domain X_{Pe} need to be enriched with competence levels and types attached with elements in X_{Pe}.

We may further add a domain $X_{Failure}$ for failures, where failure types and qualifications are provided either generally by standards or specifically by a manufacturer. Failures may be attached with failure modes, like e.g. in FMEA (AIAG & VDA, 2019), *Failure Mode and Effect Analysis, for risk analysis*, involving modes *Severity*, *Occurrence*, and *Detection*. FMEA's original approach to rating has been debated, as the risk priority number (RPN) simply multiplies ratings in a set of scores $\{0,1,\dots,10\}$, respectively attached to each failure mode, so that multiplying scores from the three failure modes results in a score in $\{0,1,\dots,1000\}$, being the RPN.

FMEA is in fact yet another highly potential area for development of many-valued scoring based on sets of qualification levels that possess algebraic structure. In such cases, the use of unital quantales is advantageous as the unit in a particular quantale can be interpreted as representing "unknown" or "missing" data. Uncertainty modelling with quantales (Eklund, 2020) thus enables to compute with missing data, which is not possible when computing with numerical data. A useful quantale for reducing FMEA's $\{0,1,\dots,10\}$ is a 6-point quantale which for the underlying lattice, one of 15 lattices, includes a 5-chain with one additional "sideline" element, which can be located in various ways in the lattice with respect to the 5-chain (Eklund, 2021). Doing so means that the RPN is computed using the semigroup operation of the quantale. In comparison, (Danilovic and Sandkull, 2005) uses $\{0,1,2,3\}$ as the set of qualifications of relations between elements in DSMs and MDMs, where qualifications are aggregated heuristically rather than subjected to the compositional machinery. We also remark that modes, like those in FMEA, may independent, as kind of a conjunction ('and'), or dependent, as kind of a directed causality. All modes may share one and the same quantale, or their respective quantales may be joined together to form a mode overarching quantale. Joined as product, $Q \times Q$, corresponding to independency of modes, as compared to joined as a directed causality, Q^Q, will produce different underlying lattices (Eklund et el., 2017a), as we noted in Section 3.

Clearly, we may add even more domains, which will require more and more domain relations, providing a wide spectrum of possibilities for solving relational equations. With five (5) domains we potentially have 10 domain mappings as relations, with 6 domains we have up to 15 relations, and so on.

The literature shows some contributions on work viewing DSM and FMEA within one and the same overall framework. In (Maurer and Kesper, 2011), a general MDM view is presented, involving DSMs for *Components*, *Functions* and *Failures*, and saying that "potential dependencies between failures can also be derived by matrix multiplication in the MDM", but DMMs are not provided. In (Neumann et al., 2012), FMEA is used in a form of *Uncertainty Mode and Effect Analysis* (UMEA) where risk evaluation is based on Monte Carlo simulation.

6 Conclusion

DSMs, often viewed just as matrices to be manipulated as in linear algebra, can favorably be represented as relations, in particular for the purpose of modelling multidomain interactions using techniques involving relational composition. Further, relations written as functions with powersets enables immediately to realize how element membership can be graded, so that uncertainties can be introduced into relations. Solving many-valued relational equations then provide a technique for transferring knowledge between domains in multidomain frameworks. A further advantage of techniques presented in this paper is that the enrichment of algebraic structure for modelling uncertainty will not prescribe the way powerset techniques are adopted for clustering and decomposition of domains but enables structure preservation across and within domains and decompositions.

Acknowledgement

This work has been supported by the projects *A general digital twin driving mining innovation through statistical and logical modelling* (DT-MINN), funded by VINNOVA, the Swedish Governmental Agency for Innovation Systems. We also gratefully acknowledge suggestions provided by Jari Kortelainen on algebraic techniques related to many-valued relational equations.

References

Bartl, E., 2012. Fuzzy relational equations, Ph.D. Thesis, Faculty of Science, Palacky University Olomouc, 2012.
Browning, T.R., 2016. Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities, IEEE Transactions on Engineering Management, 63(1), pp 27-52. doi:10.1109/TEM.2015.2491283
Danilovic, M., Browning. T.R., 2007. Managing complex product development projects with design structure matrices and domain mapping matrices, International Journal of Project Management, Volume 25, Issue 3, pp 300-314. doi:10.1016/j.ijproman.2006.11.003
Danilovic, M., Sandkull, B., 2005. The use of dependence structure matrix and domain mapping matrix in managing uncertainty in multiple project situations, International Journal of Project Management, Volume 23, Issue 3, pp 193-203. doi:10.1016/j.ijproman.2004.11.001
Di Nola, A., Sanchez, E., Pedrycz, W., Sessa, S., 1989. Fuzzy Relation Equations and Their Applications to Knowledge Engineering, Kluwer. doi:10.1007/978-94-017-1650-5
Eklund, P., 2020. Quantales for uncertainty in design engineering, In: Yang, XS., Sherratt, S., Dey, N., Joshi, A. (eds) Proceedings of Fifth International Congress on Information and Communication Technology, Advances in Intelligent Systems and Computing, vol 1184, Springer, pp 368-376. doi:10.1007/978-981-15-5859-7_36
Eklund, P., 2021. Quantales in circuit design, 2021 IEEE 51st International Symposium on Multiple-Valued Logic (ISMVL), pp. 39-42. doi:10.1109/ISMVL51352.2021.00016
Eklund, P., Galán, M.A., Helgesson, R., Kortelainen, J., 2014. Fuzzy terms, Fuzzy Sets and Systems 256, 211-235. doi:10.1016/j.fss.2013.02.012
Eklund, P., Gutiérrez García, J., Höhle, U., Kortelainen, J., 2018a. Semigroups in complete lattices: Quantales, modules and related topics, Developments in Mathematics 54, Springer. doi:10.1007/978-3-319-78948-4
Eklund, P., Höhle, U., Kortelainen, J., 2017a. Modules in health classifications, 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Naples, Italy, 2017, pp. 1-6. doi:10.1109/FUZZ-IEEE.2017.8015430
Eklund, P., Johansson, M., Kortelainen, J., 2018b. The logic of information and processes in system-of-systems applications, Soft Computing Applications for Group Decision-making and Consensus Modeling (Eds. M. Collan and J. Kacprzyk), Studies in Fuzziness 357, Springer, 89-102. doi:10.1007/978-3-319-60207-3_6
Eklund, P., Johansson, M., Kortelainen, J., V. Salminen, 2017b. The logic of DSM, Proceedings of the 19th International Dependency and Structure Modeling (DSM) Conference (Eds. K. Hölttä-Otto; T. R. Browning; S. D. Eppinger; L. Becerril), Espoo (Finland), 11-13 September 2017, pp. 25-31.
Eklund, P., Johansson, M., Kortelainen, J., Winter, M., 2019. Many-valued and many-sorted structure relations, Proceedings of the 21st International Dependency and Structure Modeling (DSM) Conference (Eds. H. Stowe, T. R. Browning, S. D. Eppinger, J. Tran, P. Montijo), Monterey (USA), 23-25 September 2019, pp. 141-150. doi:10.35199/dsm2019.8
Eklund, P., Kortelainen, J., Löfstrand, M. 2025. Quantales for fuzzy sets and relations of higher types, Mathematics, 13(13), 2159. doi: 10.3390/math13132159
Eklund, P., Löfstrand, M., 2016. Many-valued logic in manufacturing, Position Papers of the 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), ACSIS 9 (2016), 11–17. doi:10.15439/2016f73
Eppinger, S.D., Browning. T.R., 2012. Engineering Systems: Design Matrix Methods and Applications, MIT Press.
Goguen, J.A., L-fuzzy sets, Journal of Mathematical Analysis and Applications, Volume 18, Issue 1, pp 145-174. doi:10.1016/0022-247X(67)90189-8
Hisdal, E., 1981. The IF THEN ELSE statement and interval-valued fuzzy sets of higher type, International Journal of Man-Machine Studies, Volume 15, Issue 4, pp 385-455. doi: 10.1016/S0020-7373(81)80051-X
Karnik, N.N., Mendel, J.M., 2001a. Operations on type-2 fuzzy sets, Fuzzy Sets and Systems, Volume 122, Issue 2, pp 327-348. doi:10.1016/S0165-0114(00)00079-8
Karnik, N.N., Mendel, J.M., 2001b. Centroid of a type-2 fuzzy set, Inform. Sci., vol. 132, no. 1-4, pp. 195–220. doi:10.1016/S0020-0255(01)00069-X
Karnik, N.N., Mendel, J.M., Liang, Q., 1999. Type-2 fuzzy logic systems, IEEE Transactions on Fuzzy Systems, vol. 7, no. 6, pp. 643-658. doi:10.1109/91.811231

Löfstrand, M., Andrews, J., Karlberg, M., Karlsson, L., 2011. Functional product system availability: simulation-driven design and operation through coupled multi-objective optimisation, International Journal of Product Development, 13, issue 2, pp 119-131.

Maurer, M., Kesper, H., 2011. eFMEA - Raising efficiency of FMEA by matrix-based function and failure networks, Proceedings of the 13th International Dependency and Structure Modeling (DSM) Conference (Ed. A. Chakrabarti), Bangalore (India), pp. 179-186.

Mizumoto, M., Tanaka, K., 1976. Some properties of fuzzy sets of type 2, Information and Control, Volume 31, Issue 4, pp 312-340. doi:10.1016/S0019-9958(76)80011-3

Neumann, M., Sadek, T., Labenda P., Herzog, M., 2012. Integrating structure and uncertainty modeling using multiple-domain matrices, International Design Conference - Design 2012, Dubrovnic (Croatia), pp. 1771-1780.

Pimmler, T.U., 1994. A development methodology for product decomposition and integration, Massachusetts Insitute of Technology Master's Thesis.

Pimmler, T.U., Eppinger, S.D., 1994. Integration analysis of product decomposition, ASME Design Theory and Methodology Conference, Minneapolis, MN, USA. doi:10.1115/DETC1994-0034

Paul, S., Turnbull, R., Khodadad, D., Löfstrand, M., 2022. A Vibration Based Automatic Fault Detection Scheme for Drilling Process Using Type-2 Fuzzy Logic, Algorithms 15(8), 284. doi:10.3390/a15080284

Reed, S., Löfstrand, M., 2016. Logical Representation of Maintenance Procedures for Verification and Analysis. SoSE and Cyber Physical Systems, from academia to application and back, Kongsberg, Norway, June 12-15, 2016.

Sanchez, E., 1974. Equations de Relations Floues, Ph.D. Thesis in Human Biology, Faculty of Medicine of Marseille, France.

Shamsgovara, A., 2023. Enumerating, Cataloguing and Classifying All Quantales on up to Nine Elements. In: Glück, R., Santocanale, L., Winter, M. (eds) Relational and Algebraic Methods in Computer Science. RAMiCS 2023. Lecture Notes in Computer Science, vol 13896. Springer, Cham., pp 224-240. doi:10.1007/978-3-031-28083-2_14

Shamsgovara, A., Eklund, P., Winter, M., 2019. A Catalogue of Finite Quantales, GLIOC Notes. urn:nbn:se:umu:diva-239055

Steward, D., 1967. The design structure systems, General Electric report no. 67APE6, San Jose, CA; 1967.

Steward, D.V., 1981. The design structure system: A method for managing the design of complex systems," in IEEE Transactions on Engineering Management, vol. EM-28, no. 3, pp. 71-74. doi:10.1109/TEM.1981.6448589

Yang, X., Yu, F., Pedrycz, W., 2021, Typical Characteristic-Based Type-2 Fuzzy C-Means Algorithm, IEEE Transactions on Fuzzy Systems, vol. 29, no. 5, pp. 1173-1187. doi:10.1109/TFUZZ.2020.2969907

Zadeh, L.A., 1973. The Concept of a Linguistic Variable and its Application to Approximate Reasoning, Memorandum No. ERL-M411, Electronics Research Laboratory, College of Engineering, University of California, Berkeley

Zadeh, L.A., 1975a. The concept of a linguistic variable and its application to approximate reasoning-I, Information Sciences, Volume 8, Issue 3, pp 199-249. doi:10.1016/0020-0255(75)90036-5

Zadeh, L.A., 1975b. The concept of a linguistic variable and its application to approximate reasoning-II, Information Sciences, Volume 8, Issue 4, pp 301-357. doi:10.1016/0020-0255(75)90046-8

Zadeh, L.A., 1975c. The concept of a linguistic variable and its application to approximate reasoning-III, Information Sciences, Volume 9, Issue 1, pp 43-80. doi:10.1016/0020-0255(75)90017-1

AIAG & VDA, 2019. FMEA Handbook.

EN 13306, 2017. Maintenance - Maintenance terminology, European Committee for Standardization.

Contact: M. Löfstrand, Örebro University, School of Science and Technology, Örebro, Sweden, +46 707 959 429, magnus.lofstrand@oru.se, https://www.oru.se/english/employee/magnus_lofstrand