A Generic Approach to Diagrammatic Representation: The Case of Single Feature Indicator Systems

Atsushi Shimojima[1(✉)] and Dave Barker-Plummer[2]

[1] Faculty of Culture and Information Science, Doshisha University, 1-3 Tatara-Miyakodani, Kyotanabe 610-0394, Japan
ashimoji@me.com

[2] CSLI/Stanford University, Cordura Hall, 210 Panama Street, Stanford, CA 94305, USA
dbp@stanford.edu

Abstract. In this paper we take a *generic* approach to developing a theory of representation systems. Our approach involves giving an abstract formal characterization of a class of representation systems, and proving formal results based on this characterization.

We illustrate this approach by defining and investigating two closely related classes of representations that we call *Single Feature Indicator Systems* (SFIS), with and without *neutrality*. Many common representations including *tables*, such as timetables and work schedules; *connectivity graphs*, including route maps and circuit diagrams; and *statistical charts* such as bar graphs, either are SFIS or contain one as a component.

By describing SFIS abstractly, we are able to prove some properties of all of these representation systems by virtue of the fact that the properties can be proved on the basis of the abstract definition only. In particular we show that certain abstract inference rules are sound, and that each instance admits concrete inference rules obtained by instantiating the abstract counterparts.

1 Introduction

In this paper we adopt a *generic* approach to developing a theory of representation systems in general, with diagrammatic systems as a special case. Our approach involves giving an abstract formal characterization of a class of representation systems, and then proving results about the properties of all members of the class in the abstract setting. By adopting this approach, we are able to short-circuit investigation of individual representation systems, and also to assign the responsibility for the possession of various properties of an individual representation system to its membership in the class. Specifically, we do three things in this paper:

1. **Describe and formalize our view of a *representation system*.** Our formalization uses channel theory, a formal framework for modeling information

M. Jamnik et al. (Eds.): Diagrams 2016, LNAI 9781, pp. 83–97, 2016.
DOI: 10.1007/978-3-319-42333-3_7

flow, of which representation is a special case [3]. This task occupies Sects. 2 and 3 of this paper.

2. **Show how to model particular types of representation systems within the channel theory framework**. We focus on two closely related classes of representation systems: *Single Feature Indicator Systems with and without neutrality* (SFIS). This is the content of Sects. 4 and 5. SFISs are among the simplest representation systems that we can think of, and are built into a number of important, familiar diagrammatic representation systems. Each of the diagrams presented in Fig. 1 illustrates a system that either is an SFIS itself or has an SFIS as its main component. We will refer to these examples throughout this paper.
3. **Demonstrate properties held in common between all instances of the class of SFIS**. Our formalization of SFIS allows us to prove that they all share important properties. One important goal of the diagrams research community has been the development of diagrammatic proof editors, with the ability to verify the application of inference rules to diagrammatic representations. Hyperproof [2], Diamond [4], and CDEG [5] are examples of such proof editor/checkers. MixR and Openbox are frameworks for constructing heterogeneous proof systems for arbitrary representations [1,7]. For the purpose of developing such systems, it is useful to have a generic view of a set of inference rules that are guaranteed to be valid in any member of a class of diagrammatic representation systems. We discuss this in Sects. 6 and 7.

Sections 2 and 4 describe the intuitions guiding this paper, while Sects. 3 and 5 describe the corresponding formalization of these ideas. We recommend reading the informal sections first, before delving into their formalization.

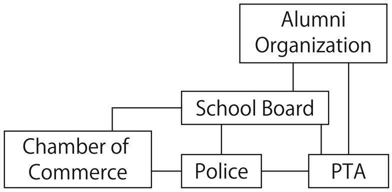
2 The General Picture

We begin by sketching the general picture of representation systems that forms the basis of the development of the theory that we present here. Our notion of representation system is designed to capture important semantic properties of a *representational practice* followed by a group of people. A representational practice is a recurrent pattern in which people express information by creating a (typically proximal) object and extract the information from it. In many cases, the information thus expressed is about a particular (distal) object or situation. We call a proximal object created on a particular occasion a *representation*. When a representation s is created to express information about a particular distal object or situation t, we say s *represents* t.

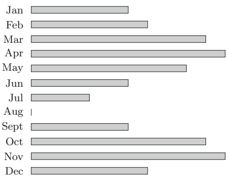
For example, a project leader may create the table in Fig. 1a to express information about the work schedules of four workers at a research project. Many people know how to extract the information expressed in this table and they do extract information from it. Here we see a representational practice followed by the project leader and these people. We will refer to this representational practice and its formalization as $\mathcal{R}_t$.

	Mon	Tues	Wed	Thurs	Fri
Atsushi	✓	✓	✓	✓	✓
Dave	✗	✗	✗	✗	✗
John	✓	✗	✓	✗	✗
Mike	✓	✗	✓	✓	✗

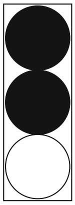
(a) Scheduling Table.



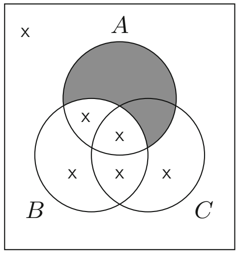
(b) Organization Chart.



(c) Bar Chart.



(d) Traffic Light.



(e) Venn Diagram.

Fig. 1. Diagrams illustrating representation systems that either are SFISs themselves or amplifications of SFISs

Typically, a representational practice is governed by various constraints of different origins, and the effectiveness of the practice deeply depends on these constraints. They consist of *source constraints* concerning what kinds of symbols appear in a representation and how they are arranged, *semantic constraints* concerning what arrangement of symbols indicate what information, and *target constraints* holding among the pieces of information expressible in the representational practice in question.

The source constraints in $\mathcal{R}_t$ include the fact that each cell of the table contains one, and only one, of the symbols ✓ and ✗. The target constraints include the facts that every worker either does, or does not, work on a particular day. The semantic constraints include the fact that a cell has the symbol ✓ only if the relevant worker works on that particular day. The project manager, his workers, and other users know these constraints and respect them to make their communication based on scheduling tables reliable and efficient.

The source constraints, semantic constraints, and target constraints governing a representational practice can be considered to make up a system, which we call a *representation system*. Thus, we can think of the system $\mathcal{R}_t$ of scheduling tables for this research project, as well as the systems underlying the other diagrams listed in Fig. 1a—the systems of connectivity maps between these organizations, of bar charts representing sales of a particular book, of diagrams of a particular traffic light, and of Venn diagrams concerning these particular sets.

In the next section we present a formalization of representation systems using channel theory [3]. In Sect. 4 we will define the notion of Single Feature Indicator System using this general theory.

3 Channel Theory and Representation Systems

Channel theory provides us with a formal framework for describing information flow, and it can be used to describe representation systems as a paradigmatic case [3]. This section is a free-standing presentation, simplified to fit our purposes, of Barwise and Seligman's discussion of representation systems within the channel theory framework (see Chap. 20 of [3]).

3.1 Types

As we discussed above, a representation system can be decomposed into three systems of constraints. Each system of constraints is modeled as a *theory*, while a theory is built on top of a set of *types*. The relevant set of types depends on the system of constraints that we are considering. When we consider source constraints of the system $\mathcal{R}_t$, for example, the set of types include the following types:

(σ_1) The intersection of a row labeled "Atsushi" and a column labeled "Tues" has a ✓.
(σ_2) The intersection of a row labeled "Atsushi" and a column labeled "Tues" has an ✗.
(σ_3) A column labeled "Tues" has at least one ✓.

while when we consider that system's target constraints, the set of types include:

(θ_1) Atsushi works on Tuesday.
(θ_2) Atsushi does not work on Tuesday.
(θ_3) At least one person is working on Tuesday.

3.2 Constraints

We represent constraints using Gentzen sequents, which are pairs of sets of types. When we write $\Gamma \vdash \Delta$, we refer to the pair of sets Γ and Δ, and indicate that this pair is a member of the set $\vdash$, or that the relation $\vdash$ holds between them.

We use lowercase Greek letters to refer to types, and uppercase Greek letters to refer to sets of types and adopt a common abuse of notation and use types and set of types interchangeably in sequents.

Types on the left hand side of the $\vdash$ are interpreted conjunctively, and on the right hand side, disjunctively. A sequent of the form $\alpha_1, \alpha_2, \alpha_3 \vdash \beta_1, \beta_2$ represents the constraint that any object which is of type α_1, α_2 and α_3, is also of one of the types β_1 or β_2 (or both). As a consequence:

1. $\alpha \vdash \beta$, represents the constraint that everything of type α is also of type β,
2. $\emptyset \vdash \alpha$, represents the constraint that everything is of type α,
3. $\alpha, \beta \vdash \emptyset$, represents the constraint that types α and β do not hold together.

For example, the Gentzen sequents $\sigma_1, \sigma_2 \vdash \emptyset$ and $\sigma_1 \vdash \sigma_3$ capture plausible source constraints in the system $\mathcal{R}_t$. Source constraints often originate in syntactic conventions combined with natural, spatial constraints on the arrangement of symbols. For example, neither of these example constraints would hold without syntactic conventions saying that there can be only one column labeled "Tues" and that a cross or a check appearing in a cell has a certain minimal size and may not overlap other marks. Both example constraints do hold in the presence of such syntactic conventions.

3.3 Theories

A theory captures a set of constraints holding in a domain by modeling them as a set of Gentzen sequents defined over a fixed set of types.

Definition 1 (Theory). *A* theory *is a pair $T = \langle \Upsilon, \vdash \rangle$, where $\vdash$ is a set of Gentzen sequents over Υ. A* constraint *of the theory T is a sequent $\langle \Gamma, \Delta \rangle$ in $\vdash$.*

When the set of constraints of a theory is logically closed, it is called a "regular theory".

Definition 2 (Regular Theory). *A theory $T = \langle \Upsilon, \vdash \rangle$ is* regular *if it satisfies the following closure conditions:*

- **Identity:** $\alpha \vdash \alpha$*, for all types* α
- **Weakening:** *If $\Gamma \vdash \Delta$, then $\Psi_1, \Gamma \vdash \Delta, \Psi_2$ for any sets of types Ψ_1, Ψ_2,*
- **Global Cut:** *If $\Psi_1, \Gamma \vdash \Delta, \Psi_2$ for any partition of any set Ψ into Ψ_1, Ψ_2, then $\Gamma \vdash \Delta$.*

The following proposition from [3] shows that any set $\vdash$ of Gentzen sequents has a unique regular theory that minimally extends it.

Proposition 1. *For every theory $T = \langle \Upsilon, \vdash \rangle$, there is a smallest regular theory on Υ containing the sequents in Σ as constraints. This is called* the regular closure *of T.*

Proof: See [3], Proposition 9.7.

3.4 Representation Systems

As we described in Sect. 2, a representation system consists of three parts, a system of source constraints (pertaining to representations), a system of target constraints (pertaining to the represented situations), and a system of semantic constraints linking representations to the represented situations. Each of these components is represented as its own theory.

Definition 3 (Representation System). *A* representation system *is a triple* $\langle T_s, T_c, T_t \rangle$, *where*

1. T_s *is a theory* $\langle \Upsilon_s, \vdash_s \rangle$, *this is the* source *theory,*
2. T_t *is a theory* $\langle \Upsilon_t, \vdash_t \rangle$, *this is the* target *theory,*
3. T_c *is a theory* $\langle \Upsilon_c, \vdash_c \rangle$ *where* $\Upsilon_c = \Upsilon_s \uplus \Upsilon_t$, *this is the* semantic *theory.*

Among the three theories posited in this definition, T_c requires further explanation.[1] As we have explained above, the semantic conventions observed in a representational practice can be considered as constraints on what arrangement of symbols indicate what information. Take the previous example of the semantic convention in $\mathcal{R}_t$, according to which a check mark in the intersection of a row labeled "Atsushi" and a column labeled "Tuesday" indicates that Atsushi works on Tuesday. Since the participants of this practice generally follow this convention, it gives rise to a constraint in their local environment, according to which σ_1 holds of the scheduling table only if θ_1 holds in the work place. The theory $T_c = \langle \Upsilon_c, \vdash_c \rangle$ captures constraints of this sort. Since the relevant constraints to capture are ones from subsets of Υ_s to subsets of Υ_t, we define the set Υ_c to be the disjoint union of these two sets: $\Upsilon_s \uplus \Upsilon_t$. The theory T_c then lists the relevant constraint from σ_1 to θ_1 as a sequent in $\vdash_c$ (i.e., $\sigma_1 \vdash_c \theta_1$). When a type σ in Υ_s and a type θ in Υ_t are connected in this way, we say that σ *indicates* θ in the system $\mathcal{R}_t$.

The above definition of representation systems significantly simplifies the one proposed by [3] (Definition 20.1) while preserving the idea that three systems of constraints make up a representation system with one system providing a semantic connection between the other two.

4 Observations Underlying the Concept of SFIS

In this section we make some observations about similarities among many familiar representation systems. We will abstract these observations into a definition of Single Feature Indicator Systems, the class of representation systems sharing these similarities. In Sect. 5 we present a formalization of this definition.

4.1 First Observation: Roles

Many diagrammatic representations consist of basic components playing certain common *roles*. These roles are *common* in the sense that they are played not only by components of particular diagrams, but by components of all diagrams used in the given representational practice. For example, each scheduling table in the system $\mathcal{R}_t$ has a basic component that plays the role of [the intersection of the row labeled "Atsushi" and the column labeled "Mon"]. The existence of such a component is mandated by syntactic stipulations and spatial constraints on scheduling tables in the system $\mathcal{R}_t$. In this way, we can think of $4 \times 5 = 20$

[1] We sometime call this the *core* theory, hence the subscript "c".

common roles for the source domain of $\mathcal{R}_t$. Such roles appear in every diagram in this representational practice, for example, if one schedule is made for each week for a year.

Similarly, consider a Venn diagram representation system, $\mathcal{R}_v$, with circles labelled A, B and C, such as depicted in Fig. 1e. Every diagram used in the system $\mathcal{R}_v$ has a component that plays the role of [the set of points inside all of the circles labeled "A," "B," and "C"]. Another role is that of [the set of points inside the circles labeled "A" and "B," but outside the circle labeled "C"]. In this way, we can think of $2^3 = 8$ roles in the source domain of $\mathcal{R}_v$. When a symbol or a place in a diagram plays a common role in this sense, we call it a *basic element* of that diagram.

4.2 Second Observation: Values

In many diagrammatic representation systems there is a fixed range of possible values that a basic element can take. Further, each basic element must take at least one value (value existence condition) but not more than one (value uniqueness condition). For example, a basic element of a scheduling table must have either a ✓ or ✗ (existence) but cannot have both (uniqueness). The basic elements of the bar chart in Fig. 1c are individual bars, and each has a certain height (existence) but not more than one height (uniqueness).

4.3 Third Observation: Features in the Source Domain

The combination of roles and values that satisfy the value existence condition and the value uniqueness condition give rise to a structure that we call a *feature*.

For example, the source domain of our scheduling tables involves a feature consisting of 20 common roles (played by cells) and 2 values (having a ✓ or ✗). The source domain of our bar charts involves a feature consisting of 12 roles (played by bars) and an infinite number of values (heights). The source domain of our organization charts involves a feature consisting of ${}_5C_2 = 10$ roles (played by pairs of organization names) and 2 values (directly connected or not).

Notice that in each of the example representation systems, the values taken by the various roles in the source domain are *independent*. That is, as far as the syntactic stipulations and spatial constraints are concerned, the basic element playing a role can take any value without consideration of the values of other basic elements. We call this the *independence condition*.

4.4 Fourth Observation: Feature in the Target Domain

Often the target domain in a diagrammatic system makes up a feature too. For example, in any given week represented by a scheduling table, Atsushi either works or does not work on Monday, but he cannot do both. We can restate this as the fact that the element of the described situation playing the role of $\langle \text{Atsushi}, \text{Monday} \rangle$ must have either the property [working_on] or the property [not_working_on] (value existence) but cannot have both (value uniqueness). Here, $\langle \text{Atsushi}, \text{Monday} \rangle$ is a role, and the properties [working_on] and [not_working_on] make up the value range. The other roles are $\langle \text{Atsushi}, \text{Tuesday} \rangle$, $\langle \text{Mike}, \text{Friday} \rangle$, and so on, counting up to $4 \times 5 = 20$ pairs.

Similarly, in any particular group of people represented by an $\mathcal{R}_v$-diagram, the set of $A \cap B \cap C$ must be either empty or non-empty (existence) but cannot be both (uniqueness). In this case $2^3 = 8$ sets, including $A \cap B \cap C$, $A \cap B \cap \overline{C}$ and $A \cap \overline{B} \cap C$, constitute the set of roles, and the value range is $\{\text{empty}, \text{non-empty}\}$.

4.5 Fifth Observation: Semantic Correspondence of Features

We have just seen that the source and the target domain of a diagrammatic system often make up features. Our final observation is that these features typically stand in a close correspondence through the system's semantic conventions.

Take the case of bar charts. The source role of [the bar labeled "Jan"] corresponds to the target role [January]. In this way, a natural one-one correspondence holds between the set of source roles and the set of target roles in this system. A natural correspondence holds between the sets of values too. Each possible height taken by a bar corresponds to a possible number of books sold in the corresponding month. In the case of scheduling tables, the source role of [the intersection of the row labeled "Atsushi" and the column labeled "Mon"] corresponds to the target role of (Atsushi, Monday) and similarly for the other roles. The two source values, having a ✓ or ✗, each corresponds to a unique target value, [working_on] or [not_working_on]. The reader can easily check a similar two-fold correspondence holds between the source feature and the target feature involved in each of the other systems illustrated in Fig. 1.

These correspondences underlie semantic conventions in these systems. In the system of scheduling tables $\mathcal{R}_t$, if the intersection labeled "John" and the column labeled "Mon" has a ✓ in a scheduling table, it indicates that John works on Monday. In the system of bar charts, that the bar labelled January having a height of 10 mm indicates that the number of book sales in the month of January being 100 units. In this way, many diagrammatic representation systems employ semantic conventions with the form that, if a basic element playing role r has the value v, it indicates that the element playing the role corresponding to r has the value corresponding to v. We call representation systems having this form of semantic convention "Single Feature Indicator Systems" or "SFISs" for short.

5 Single Feature Indicator Systems – Formalized

One of the contributions that we make in this paper is a demonstration of how channel theory can be used to formalize classes of representation systems. This section of the paper, where we formalize the notion of Single Feature Indicator System, is focussed on this task. Our approach is to specialize Barwise and Seligman's definition of representation system that we presented in Definition 3, so that the types and constraints in the component theories capture the observations about roles and values that we outlined in Sect. 4.

5.1 Features

In Sect. 4, we observed that the target and source domains of many diagrammatic representation systems can be characterized as consisting of roles which take on specific values. We formalize this idea by using an ordered pair $\langle r, v\rangle$ to model the type that holds on a representation d if and only if the element of d playing the role r has the value v. For example, a diagram in the traffic light representation system illustrated in Fig. 1d is of type $\langle \mathit{uppermost\_circle}, \mathit{white}\rangle$ if the uppermost circle of d is white. A *feature* is a specialized theory over these types:

Definition 4 (Feature). *A* feature *is a regular theory $T = \langle \Upsilon, \vdash\rangle$ for which there are sets R and V such that:*

1. $R \times V = \Upsilon$,
2. *For every $r \in R$,*
   (a) $\vdash \{\langle r, v\rangle : v \in V\}$,
   (b) $\langle r, v\rangle, \langle r, v'\rangle \vdash \emptyset$ *for all distinct* $v, v' \in V$.

When these conditions hold, R and V are called the set of roles *and the* set of values *of the feature T, respectively.*

Clause 1 declares that this theory is concerned with pairs of the form $\langle r, v\rangle$ with a role $r \in R$ and a value $v \in V$. Clauses 2a and 2b capture the value existence condition and the value uniqueness conditions respectively.

5.2 Single Feature Indicator Systems

A Single Feature Indicator System is a representation system whose source and target theories are features with appropriate connections provided by the semantic theory.

Definition 5 (Single Feature Indicator System (SFIS)). *A representation system $\langle S, C, T\rangle$ is a* Single Feature Indicator System (SFIS) *iff:*

1. *S is a feature with the set R_s of roles and the set V_s of values*
2. *T is a feature with the set R_t of roles and the set V_t of values*
3. *Every assignment $f : R_s \to V_s$ of values in V_s to roles in R_s is consistent, i.e., $f \nvdash_S \emptyset$*
4. *C is the theory $\langle \Upsilon_c, \vdash_c\rangle$ where there are bijections p_r from R_s to R_t and p_v from V_s to V_t such that $\vdash_c$ is the regular closure of the set of all sequents $\langle \{\langle r, v\rangle\}, \{\langle p_r(r), p_v(v)\rangle\}\rangle$ where $\langle r, v\rangle \in \Sigma_S$.*

Conditions 1 and 2 state that the source and target theories are features. The additional condition on the source feature expressed by Clause 3 is the *independence condition*, which implies that basic elements of a representation can take any value no matter what values are taken by other basic elements.

The target theory T does not necessarily satisfy such a condition, which is to say that the constraints on T may result in some assignments of values not being permitted. In our model of a representation system, the source theory captures only those constraints originating in spatial constraints and syntactic stipulations associated with a representational practice. We have seen that as far as these constraints are concerned, the assignments of values to roles are independent of one another. On the other hand, T is intended to capture any constraint that holds on the types Σ_T in the target domain. So for example, there are no spatial or syntactic constraints preventing the drawing of a traffic light diagram with both the uppermost and lowermost circles being white, but there are additional constraints in the target domain which should prevent such a combination (on the assumption that a white circle indicates that the corresponding lamp is illuminated).

Finally, condition 4 tells us about the connections between the source and target theories. The projection functions p_r and p_v respectively associate roles in the source with roles in the target, and values in the source with values in the target. This clause requires that the system's semantic theory respects the correspondence between source types and target types established by these projection functions.

5.3 Single Feature Indicator System with Neutrality

Before we discuss the logical properties of Single Feature Indicator Systems we will introduce a closely related, and more interesting, class of representation systems that we call Single Feature Indicator System with Neutrality.

Consider a situation where you are observing the author of the scheduling table in Fig. 1a as it is being constructed. Perhaps all of the row and column labels are present, but the author has not yet filled in all of the cells. Such a representation carries *partial* information about the target that it describes. We can see, perhaps, that Dave will not be working on Monday, but whether or not Atsushi is working that day is not represented in the diagram.

We can model this situation by introducing a third kind of source value, a blank, in addition to ✓ and ✗. But we do not want to assign a target value to this source value. The function p_v defined to map source properties to target properties must be partial with respect to this blank property value.

The definition of a Single Feature Indicator System with Neutrality is similar to that of an SFIS.

Definition 6 (SFIS with Neutrality (SFIS$^\perp$)). *A representation system* $\langle S, C, T\rangle$ *is a* Single Feature Indicator System with Neutrality (SFIS$^\perp$) *iff there are sets* R_s, V_s, R_t, V_t *such that*

1. *S is a feature with the set R_s of roles and the set V_s of values*
2. *V_s contains a distinguished value $\perp$*
3. *Every assignment $f : R_s \to V_s$ of roles in R_s to values in V_s is consistent, i.e., $f \not\vdash_S \emptyset$*
4. *T is a feature with the set R_t of roles and the set V_t of values*
5. *There is a bijection p_r from R_s to R_t and a bijection p_v from $V_s - \{\perp\}$ to V_t such that $f_s(\langle r, v\rangle) \vdash_C f_t(\langle p_r(r), p_v(v)\rangle)$ for every $\langle r, v\rangle \in (R_s \times V_s - \{\perp\})$.*

The critical difference between this definition and the definition of Single Feature Indicator System *simpliciter* is that the source domain contains a neutral value $\perp$, and that this value is not in the domain of the bijection p_v, and therefore carries no information about the target.

6 Specifications and Semantic Consequence

We now turn out attention to some inference rules supported by every SFIS.

Before we can define these inference rules, we need to have a clear notion of consequence between diagrams. That is, we must define what it means for one diagram to follow from another. But before we can do this we need a way to describe complete diagrams.

Let $\langle S, C, T\rangle$ be an SFIS (with or without neutrality). We call any function $\Sigma : R_s \to V_s$ a *complete specification of the source.* Σ is a set of types assigning a unique value to every role in R_s. Such a function completely describes a source representation by associating a value with each role. Indeed, we can think of the complete specification *as* the representation with the visual appearance of the roles and values abstracted away. As the range of diagrams in Fig. 1 attest, the values associated with roles can be represented in a variety of ways, but semantically, only the particular association of roles to values matters.

If Σ is a complete specification, we can define

$$Ind(\Sigma) = \{\langle p_r(r), p_v(v)\rangle : \langle r, v\rangle \in \Sigma \text{ and } v \neq \perp\}$$

$Ind(\Sigma)$ is the set of target types *indicated* by the source types in Σ.

The critical definition is of what it means for a specification be a consequence of other specifications. If S is a set of representations (or more precisely, their complete specifications) and Σ_0 another representation, then Σ_0 is a logical consequence of S if everything indicated by Σ_0 is entailed by the union of types indicated by the members of S, or formally:

Definition 7 (Semantic Consequence). *Given a collection S of complete specifications of source (the premises), and a complete specification of source Σ_0 (the conclusion), we say that Σ_0 is a* semantic consequence *of S and write $S \Longrightarrow \Sigma_0$ iff $\bigcup\{Ind(\Sigma) : \Sigma \in S\} \vdash_t \sigma$, for all $\sigma \in Ind(\Sigma_0)$.*

7 Inference Rules

We now have everything that we need in order to define inference rules for SFISs, and to demonstrate their soundness in our theory.

7.1 Contradiction

Definition 8 (Value Conflict). *Two complete specifications of source Σ_1 and Σ_2 have an* value conflict *iff there is some role $r \in R_s$ and values v_1 and v_2 in V_s such that $\langle r, v_1\rangle \in \Sigma_1$ and $\langle r, v_2\rangle \in \Sigma_2$ and $v_1 \neq v_2$, $v_1 \neq \perp$ and $v_2 \neq \perp$.*

Theorem 1 (Contradiction). *Given complete specifications of source Σ_1 and Σ_2, if Σ_1 and Σ_2 have a value conflict, then $\Sigma_1, \Sigma_2 \Longrightarrow \Sigma$ for any complete specification of source Σ.*
Proof: *$Ind(\Sigma_1) \cup Ind(\Sigma_2)$ has both $\langle p_r(r), p_v(v_1)\rangle$ and $\langle p_r(r), p_v(v_2)\rangle$ for some $r \in R_s$ and distinct v_1, v_2. $\langle p_r(r), p_v(v_1)\rangle, \langle p_r(r), p_v(v_2)\rangle \vdash_t \emptyset$ because T is a feature and $p_v(v_1) \neq p_v(v_2)$. By weakening, $\langle p_r(r), p_v(v_1)\rangle, \langle p_r(r), p_v(v_2)\rangle \vdash_t \sigma$ for any target type σ, so certainly for any $\sigma \in Ind(\Sigma)$.*

We therefore obtain a generic inference rule which we will call SFIS-Contradiction. This a *generic* rule since the rule may be specialized to a particular concrete version of the rule in any particular SFIS, whether it is in the system of scheduling tables, that of Venn diagrams, or that of connectivity maps. For example, if one organization chart shows a connection between [Police] and [PTA], and another shows no such connection, the rule lets us derive any organization chart from the two.

We now turn our attention to additional generic inference rules which are available only within SFIS$^\perp$ since they require the existence of a neutral value for their specification.

7.2 Erasure

In what follows, we need some definitions which will help us to describe manipulations of complete specifications of the source (manipulations of the diagrams that they describe).

Definition 9 (Point Substitution). *Suppose that Σ is complete specification of source. Let $\Sigma^{\langle r,v\rangle}$ be defined in the following way:*

$$\Sigma^{\langle r,v\rangle} = (\Sigma - \{\langle r, v'\rangle\}) \cup \{\langle r, v\rangle\}$$

where v' is the value taken by r in Σ. We call $\Sigma^{\langle r,v\rangle}$ the $\langle r, v\rangle$-substitution *of Σ.*

The $\langle r, v\rangle$-substitution of Σ is just like Σ, except that $\langle r, v\rangle$ is a member of $\Sigma^{\langle r,v\rangle}$, instead of $\langle r, v'\rangle$. Note that, since Σ is a complete specification of source, there is some v' such that $\langle r, v'\rangle \in \Sigma$, and that $\Sigma^{\langle r,v\rangle}$ is also a complete specification of source.

As special cases of point substitution, we call the $\langle r, \perp\rangle$-substitution of Σ, the *r-erasure* of Σ, and, if $\langle r, \perp\rangle \in \Sigma$ and $v \neq \perp$, we call $\Sigma^{\langle r,v\rangle}$ the *$\langle r, v\rangle$-extension* of Σ.

Theorem 2. *If Σ_1 is a complete specification of source in an SFIS$^\perp$ and Σ_2 is the r-erasure of Σ_1, then $\Sigma_1 \Longrightarrow \Sigma_2$.*
Proof: *The result follows from Identity (Definition 2), since $Ind(\Sigma_2) \subset Ind(\Sigma_1)$.*

In this case, we say that Σ_2 may be obtained from Σ_1 by SFIS$^\perp$-Erasure. As an example of its use: if an organization chart showing a connection between [Police] and [PTA] represents the world accurately, then a chart which is otherwise identical but is non-committal about the existence of a connection between these organizations is also accurate, though less informative.

If Σ_2 may be obtained from Σ_1 by repeated use of this rule, then we say that Σ_2 is *an erasure of* Σ_1.

Corollary 1. *If Σ_2 is an erasure of Σ_1, then $\Sigma_1 \Longrightarrow \Sigma_2$.*
Proof: *Trivial using Theorem 2.*

7.3 Proof by Cases

Any SFIS$^\perp$ supports an inference rule allowing proof by cases. We know that in the target theory, there is some property enjoyed by the target role, which corresponds to a source role having one or other of the definite properties. If some source role in a diagram token has the neutral property $\perp$ then there is a collection of new representations, which differ from the original, and from each other, only in their assignment of a source property to this role. One of these representations is a faithful representation of the target.

Suppose that $\langle r, \perp \rangle \in \Sigma$ for some role r. Then we define the set of r-extensions of Σ, denoted $Ext_r(\Sigma)$ to be the set containing the $\langle r, v \rangle$-extensions of Σ for $v \in V_s$. If the set V_s is finite, then so is $Ext_r(\Sigma)$.

Theorem 3. *If $\Sigma' \Longrightarrow \Sigma^*$ for every $\Sigma' \in Ext_r(\Sigma)$, then $\Sigma \Longrightarrow \Sigma^*$.*
Proof: *Assume the antecedent, and let σ be an arbitrary member of $Ind(\Sigma^*)$. It suffices to show $Ind(\Sigma) \vdash_t \sigma$. Let $A = \{\langle p_r(r), p_v(v) \rangle : v \in V_s\}$. To apply Global Cut, we show $A_1, Ind(\Sigma) \vdash_t \sigma, A_2$ for every partition $\langle A_1, A_2 \rangle$ of A. Suppose $A_1 = \emptyset$. Then $A_2 = A$. By the definition of a feature, $\vdash_t A$. Thus, by Weakening, $A_1, Ind(\Sigma) \vdash_t \sigma, A_2$. Suppose on the other hand that $A_1 \neq \emptyset$. Then $\langle p_r(r), p_v(v') \rangle \in A_1$ for some $v' \in V_s$. Let Σ' be the r-extension of Σ for v'. Then $Ind(\Sigma') = Ind(\{\langle r, v' \rangle\} \cup \Sigma)$, and so $Ind(\Sigma') \subseteq A_1 \cup Ind(\Sigma)$. But $Ind(\Sigma') \vdash_t \sigma$ by assumption. So, by Weakening, $A_1, Ind(\Sigma) \vdash_t \sigma, A_2$.*

This permits the definition of a generic inference rule SFIS$^\perp$-Cases, which lets us derive any representation Σ' from Σ if Σ' is derivable from every r-extension of Σ for some role r.

7.4 Disjunction

Every SFIS$^\perp$ supports a rule which allows the weakening of information from a collection of representations into a single representation.

Definition 10 ($\Sigma_{\bigvee S}$). *Let S be a set of complete specifications of source, define $\Sigma_{\bigvee S}$ as follows:*

For each $r \in R_s$:

(a) $\Sigma_{\bigvee S}(r) = v$ *iff* $\Sigma(r) = v$ *for all* $\Sigma \in S$
(b) $\Sigma_{\bigvee S}(r) = \bot$ *otherwise*

We call $\Sigma_{\bigvee S}$ the disjunction of S.

Theorem 4. *Suppose that* $\langle r, \bot \rangle \in \Sigma$ *for some role* r. *If* S *is a set of complete specifications of source, and for each* $\Sigma' \in Ext_r(\Sigma)$ *there is* $\Sigma^* \in S$ *such that* $\Sigma' \Longrightarrow \Sigma^*$, *then* $\Sigma \Longrightarrow \Sigma_{\bigvee S}$.
Proof: *Observe* $\Sigma_{\bigvee S}$ *is a consequence of every* r*-extension of* Σ, *since* $\Sigma_{\bigvee S}$ *is a consequence of every member of* S *(Corollary 1) while every* r*-extension of* Σ *has some member of* S *as a consequence (assumption). Then apply Theorem 3.*

This theorem immediately permits the definition of an abstract inference rule, SFIS$^\bot$-Merge, which is a more useful version of proof by cases. Rather than insisting that each subproof of the proof by cases derive the same specification, possibly involving uses of erasure, we can allow the different cases to derive different specifications, but the disjunction of those specifications is exported into the main proof. An instance of SFIS$^\bot$-Merge is implemented in the Hyperproof program [2] under the name of MERGE.

7.5 Conjunction

Definition 11. **($\Sigma_{\bigwedge S}$).** *Let* S *be a set of complete specifications of source. Define* $\Sigma_{\bigwedge S}$ *as follows:*

1. $\Sigma_{\bigwedge S}$ *is undefined if* S *contains a value conflict,*
2. *for each* $r \in R_s$*:*
   (a) $\Sigma_{\bigwedge S}(r) = v$ *iff* $\Sigma(r) = v$ *for some* $\Sigma \in S$, *and* $v \neq \bot$
   (b) $\Sigma_{\bigwedge S}(r) = \bot$ *otherwise*

We call $\Sigma_{\bigwedge S}$ the conjunction of S.

Theorem 5. *If* S *is a set of complete specifications of source with no value conflict, then* $S \Longrightarrow \Sigma_{\bigwedge S}$.
Proof: $\bigcup\{Ind(\Sigma) : \Sigma \in S\} = Ind(\Sigma_{\bigwedge S})$.

8 Conclusion

In this paper, we have launched what may be called *generic approach* to the formalization of diagrammatic proof systems. The strategy is to characterize a class of diagrammatic systems using a formal property they commonly have. On the basis of this characterization we can investigate other properties that hold of this class of systems.

In this particular paper we define and analyze two related classes of representation systems, namely, SFIS and SFIS$^\bot$. Although these systems are rather straightforward to characterize, they include a surprisingly large category of diagrammatic representation systems. In this paper, we have developed a set of generic inference rules whose soundness is provable on the basis of membership in the class of these systems. SFIS and SFIS$^{\perp}$ support a substantial set of generic inference rules, including Contradiction, Proof by Cases, Conjunction and Disjunction, which can serve the foundation for more complex inference rules that are applicable in more sophisticated diagrammatic representation systems.

In future work, we propose to investigate other properties of SFIS, for example their ability to support free rides and diagrammatic consistency-check [6]. Again, we will investigate these ideas in the abstract and demonstrate the conditions under which SFIS have such properties.

In our view, many more sophisticated systems are derivatives of SFIS. We have already discussed one derivative of SFIS, namely, SFIS$^{\perp}$. We can also think of derivatives whose indication relation is amplified through the meaning derivation mechanism [6] or through the introduction of concrete symbols that indicate abstract information. Each of these different kinds of variants seem to define its own class of diagrammatic representation systems (just as SFIS does) and allow the generic approach to the set of inferences rules applicable in all representation systems in the category. This will open a way to a generic diagrammatic proof editor and checker that incorporates a wide range of diagrammatic systems in an incremental, but systematic manner.

References

1. Barker-Plummer, D., Etchemendy, J., Murray, M., Pease, E., Swoboda, N.: Learning to use the openbox: a framework for the implementation of heterogeneous reasoning. In: Cox, P., Plimmer, B., Rodgers, P. (eds.) Diagrams 2012. LNCS, vol. 7352, pp. 3–3. Springer, Heidelberg (2012)
2. Barwise, J., Etchemendy, J.: Hyperproof. CSLI Publications, Stanford (1994)
3. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge University Press, Cambridge (1997)
4. Jamnik, M.: Mathematical Reasoning with Diagrams. CSLI Publications, Stanford (2001)
5. Miller, N.: Euclid and His Twentieth Century Rivals: Diagrams in the Logic of Euclidean Geometry. CSLI Publications, Stanford (2007)
6. Shimojima, A.: Semantic Properties of Diagrams and Their Cognitive Potentials. Studies in the Theory and Applications of Diagrams. CSLI Publications, Stanford (2015)
7. Urbas, M., Jamnik, M.: A framework for heterogeneous reasoning in formal and informal domains. In: Dwyer, T., Purchase, H., Delaney, A. (eds.) Diagrams 2014. LNCS, vol. 8578, pp. 277–292. Springer, Heidelberg (2014)