On the Completeness of Spider Diagrams Augmented with Constants

Gem Stapleton, John Howse, Simon Thompson, John Taylor, and Peter Chapman

Abstract Diagrammatic reasoning can be described formally by a number of diagrammatic logics; spider diagrams are one of these, and are used for expressing logical statements about set membership and containment. Here, existing work on spider diagrams is extended to include constant spiders that represent specific individuals. We give a formal syntax and semantics for the extended diagram language before introducing a collection of reasoning rules encapsulating logical equivalence and logical consequence. We prove that the resulting logic is sound, complete and decidable.

Keywords Spider diagrams · Constants · Soundness · Completeness · Monadic first-order logic · Diagrammatic reasoning

Mathematics Subject Classification (2010) Primary 68R02; Secondary 03B02

1 Introduction

Diagrams have been used for centuries in the visualization of mathematical concepts and to aid the exploration and formalization of ideas. This is not the place to survey that history; however, we give a brief overview of the background to the development of spider diagrams now.

One of the most successful visual notations is the Venn diagram for sets and their relationships; indeed, it is taught in the elementary school curriculum in many countries. While Venn diagrams contain all possible intersection regions between the sets, Euler diagrams [4] allow set intersection, disjointness and containment to be represented visually. The Euler diagram d_1 in Fig. 1 asserts that A and B are disjoint and C is a subset of A. The relative placement of the curves gives, for free, that C is disjoint from B. This 'free ride' is one of the areas where diagrams are thought to be superior to symbolic languages [20]. This example also illustrates the concept of 'well-matchedness' [8] since the visual representation of assertions mirrors those at the semantic level: for example, the containment of one curve by another mirrors the interpretation that the enclosed curve, C, represents a subset of the set represented by the enclosing curve, A. Moreover, this has the added benefit that the subset relation is mirrored by the transitive property of syntactic containment.

Various extensions to Euler diagrams have been proposed, such as including syntax to represent named individuals [27], or assert the existence of arbitrary finite numbers of elements [12]. The Euler diagram d_2 in Fig. 1 is augmented with shading, which asserts

A. Moktefi, S.-J. Shin (eds.), *Visual Reasoning with Diagrams*, Studies in Universal Logic, DOI 10.1007/978-3-0348-0600-8_7, © Springer Basel 2013

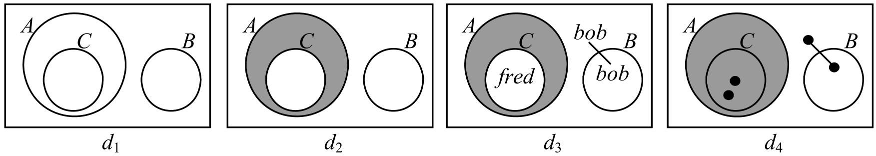


Fig. 1 Extended Euler diagrams

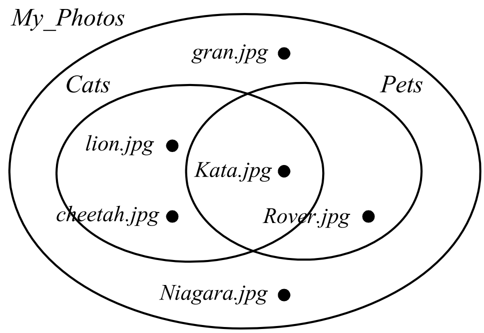


Fig. 2 Non-hierarchical file systems

the emptiness of the set $A - C$ and the Euler/Venn diagram d_3 tells us, in addition, that *fred* is in the set C and *bob* is not in the set A.

Spider diagrams [12] are also based on Euler diagrams. The spider diagram d_4 in Fig. 1 asserts the existence of two elements in the set C and at least one element outside of the set A; this is accomplished through the use of *existential spiders*. A spider is a tree which denotes a single element that can occupy one of the positions given by the nodes of the tree. The shading in d_4 is used to place an upper bound on the cardinality of A, limiting it to two: in a set represented by a shaded region, all elements must be denoted by spiders. Using a model-theoretic argument, it has been shown that spider diagrams are equivalent to Monadic First-Order Logic with equality [23].

Constant spiders [21, 25], corresponding to given spiders in [11], were introduced to provide users of spider diagrams with an explicit way to write constraints involving named individuals. There are a number of examples of spider diagrams being used in practice, such as assisting with the task of identifying component failures in safety critical hardware designs [2]. Equivalent notations have been used for representing non-hierarchical computer file systems [3], in a visual semantic web editing environment [16, 28] and for viewing clusters which contain concepts from multiple ontologies [9]. Each of these applications uses constants to represent specific objects, thus motivating the utility of augmenting spider diagrams with constants. To take a particular example, the VennFS system [3], is used to represent visually non-hierarchical files systems. The example in Fig. 2 provides information about the folder location of certain files stored on a computer: the labeled dots are files—or constant spiders—and the curves represent folders.

In [25], it was established that constants in spider diagrams could be simulated by a shaded contour containing a single (non-constant) spider. This translation gave a diagram that was expressively equivalent to the original, in the sense that it had the same model set as the spider diagram with a constant. As with many notations—both symbolic and diagrammatic—it is worthwhile adding a notation even though it might be dismissed as mere 'syntactic sugar'. The additional notation makes clear the *intention* of the user, and allows that intention to be preserved in reasoning, for instance. In a visual notation it makes it much easier to preserve the 'free ride' and 'well matchedness' properties; in the particular case of constants there is a direct naming of a constant, rather than an indirect naming through the name of the representing contour, for instance. Further discussion and motivation can be found in [21, 25].

Earlier work formalized the syntax and semantics of spider diagrams and specified a logic for the diagrams which was proved to be sound, complete and decidable; in this paper we do the same for spider diagrams with constants. Specifically, in Sect. 2, we give the syntax of spider diagrams extended to include constant spiders and, in Sect. 3, present formal semantics. In Sect. 4, we provide a collection of reasoning rules for spider diagrams with constants and, in Sect. 5, we present sketches of soundness, completeness and decidability results.

2 Syntax

In diagrammatic systems, we can distinguish two levels of syntax: concrete (or token) syntax and abstract (or type) syntax [10]. Concrete syntax captures the physical representation of a diagram. Abstract syntax is independent of the semantically unimportant spatial relations between syntactic elements in a concrete diagram. We do not include the concrete syntax in this discussion since we work at the abstract level here.

The closed curves in a spider diagram are called *contours* and each contour is identified by a label chosen from a countably infinite set, $\mathcal{CL}$. A *zone*[1] is defined to be a pair (in, out) of disjoint finite subsets of $\mathcal{CL}$. The set *in* contains the labels of the contours that include the zone (in, out) whereas *out* is the set of labels of the contours that do not include (in, out). So, in a unitary diagram, *in* and *out* form a partition of the contour label set. In diagram d_1 in Fig. 3 the zone that is inside contour A but outside B and C has abstract representation $(\{A\}, \{B, C\})$. A *region* is a set of zones. We define $\mathcal{Z}$ and $\mathcal{R} = \mathbb{P}\mathcal{Z}$ to be the sets of all zones and regions respectively. As noted earlier, in a Venn diagram, d, every possible zone—that is every element of $\mathbb{P}L$ for the set L of contour labels in d—is represented in d. This is not the case for spider diagram, and a zone is said to be *missing* if it is not a member of the possible zone set for the diagram.

A spider without a label is called an *existential spider*. A spider with a label is called a *constant spider*. A spider *touches* a zone if that zone is in its habitat, and a spider is said to *inhabit* the region in which it is placed, which is termed its *habitat*. To describe the existential spiders in a particular diagram, it is sufficient to say how many existential spiders there are in each region. We will use a bag of regions, called *existential spider descriptors*, with the number of occurrences of each region in the bag giving the number of existential spiders in the region. For example, the region

$$\big\{\big(\{A, C\}, \{B\}\big), \big(\emptyset, \{A, B, C\}\big), \big(\{B\}, \{A, C\}\big), \big(\{B, C\}, \{A\}\big)\big\}$$

in diagram d_2 in Fig. 3 contains two existential spiders. We must also specify which constant spider labels appear and, for each spider label, the habitat of the spider with that

[1] Since all constructs discussed here are abstract, we will use the terminology 'zone' rather than 'abstract zone' throughout.

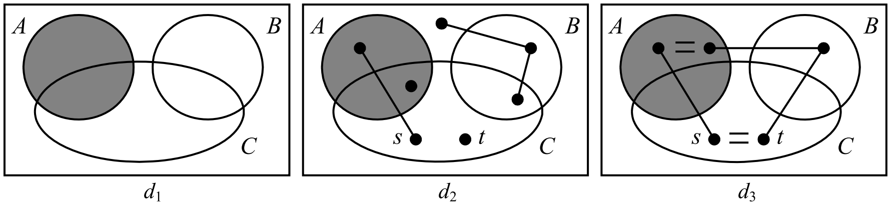


Fig. 3 Examples of unitary spider diagrams

label. At the abstract level, a unitary diagram will contain a finite set of constant spider labels together with a habitat function, mapping each constant spider label to a region in the diagram. The habitat of the constant spider labeled s in diagram d_2 in Fig. 3 is $\{(\{A\}, \{B, C\}), (\{C\}, \{A, B\})\}$.

We will assume that all of the constant spider labels come from a finite set $\mathcal{CS}$. An alternative choice would be to have a countably infinite set of constant spider labels. With this alternative choice, the work below on reasoning rules, soundness and completeness remains identical. However, the approach taken in [23] to prove that augmenting the spider diagram language with constants does not increase expressiveness would need to be modified.

Given two distinct constant spiders, each with a habitat sharing some zone z, a *tie*, represented by an 'equals' sign, can be placed between them in z. The *web* of a pair of constant spiders is the set of zones that contain a tie between those two spiders. The diagram d_3 in Fig. 3 contains two constant spiders, labeled s and t, connected by two ties. The web of s and t is the region made up of the zone inside contour A but outside B and C and the zone inside C but outside A and B.

General spider diagrams are a logical combination of diagrams; a single diagram is called *unitary*. The formal definition of an abstract unitary spider diagram with constants extends that given in [12] for unitary spider diagrams without constants. We assume that the sets $\mathcal{CS}$, $\mathcal{CL}$, $\mathcal{Z}$ and $\mathcal{R}$ are all pairwise disjoint.

Definition 2.1 An **abstract unitary spider diagram with constants**, d (with contour labels in $\mathcal{CL}$ and constant spider labels in $\mathcal{CS}$), is a 7-tuple

$$\langle L, Z, Z^*, ESD, CS, \theta, \omega \rangle$$

whose components are defined as follows.

1. $L = L(d) \subset \mathcal{CL}$ is a finite set of contour labels.
2. $Z = Z(d) \subseteq \{(in, L - in) : in \subseteq L\}$ is a set of zones such that
   (i) for each label $l \in L$ there is a zone $(in, L - in) \in Z(d)$ such that $l \in in$ and
   (ii) the zone $(\emptyset, L)$ is in $Z(d)$.
   We define $R(d) = \mathbb{P}Z - \{\emptyset\}$ to be the set of regions in d. We further define $MZ(d) = \{(in, L - in) : in \subseteq L\} - Z(d)$ to be the **missing zones** of d.
3. $Z^* = Z^*(d) \subseteq Z$ is a set of **shaded zones** and we define $R^*(d) = \mathbb{P}Z^*(d)$ to be the set of shaded regions in d. A region, $r \in R(d) - R^*(d)$, is **completely non-shaded** if and only if $r \cap Z^*(d) = \emptyset$.
4. $ESD = ESD(d) \subset \mathbb{Z}^+ \times R(d)$ is a finite set of **existential spider descriptors** such that

$$\forall (n_1, r_1), (n_2, r_2) \in ESD \ (r_1 = r_2 \Rightarrow n_1 = n_2).$$

If $(n, r) \in ESD$ we say there are n existential spiders with **habitat** r.

5. $CS = CS(d) \subseteq \mathcal{CS}$ is a finite set of constant spider labels.
6. $\theta = \theta_d : CS \to R(d)$, is a function which maps each constant spider label to a region in d. If $\theta_d(s_i) = r$ we say s_i has **habitat** r in d.
7. $\omega = \omega : CS(d) \times CS(d) \to \mathbb{P}Z$ is a function which returns the **web** of each pair of constant spiders where $z \in \omega(s_i, s_j)$ means that there is a **tie** between s_i and s_j in the zone z. Further, ω must ensure that the following hold for all s_i, s_j, s_k in $CS(d)$:
   (a) given two constant spiders there can only be ties in zones common to their habitat: $\omega(s_i, s_j) \subseteq \theta_d(s_i) \cap \theta_d(s_j)$,
   (b) each constant spider is joined by ties to itself (this simplifies the formalization of the semantics below): $\omega(s_i, s_i) = \theta_d(s_i)$,
   (c) if there is a tie between constant spiders s_i and s_j in zone z, then there is a tie between s_j and s_i in z: $\omega(s_i, s_j) = \omega(s_j, s_i)$, and
   (d) given any zone z, if s_i and s_j are joined by a tie in z and so are s_j and s_k, then s_i and s_k are joined by a tie in z: $z \in \omega(s_i, s_j) \cap \omega(s_j, s_k) \Rightarrow z \in \omega(s_i, s_k)$.

Some remarks about the above definition are in order, before we illustrate it with an example.

- Every contour in a concrete diagram contains at least one zone as captured by condition 2 (i).
- In any concrete diagram, the zone inside the boundary rectangle but outside all the contours is present and this is captured by condition 2 (ii).
- Being joined by a tie is interpreted transitively. In fact, ties give rise to an equivalence relation on the spiders in each zone, as specified by conditions 7 (b), (c) and (d).
- Therefore, in a zone z, taking the constant spiders in z as a set of vertices and the ties in that zone as a set of edges, we would have a graph whose components formed complete graphs with loops at each vertex. However, in our concrete syntax we will only draw a spanning forest in each zone so as to avoid unnecessary clutter in diagrams.
- We note that ties could also be used to connect existential spiders. Indeed, they could also be used to connect an existential spiders to constant spiders.[2]

Example The diagram d_1 in Fig. 4 has the following abstract description.

1. Contour label set $L(d_1) = \{A, B\}$.
2. Zone set

$$Z(d_1) = \{(\emptyset, \{A, B\}), (\{A\}, \{B\}), (\{B\}, \{A\}), (\{A, B\}, \emptyset)\}.$$

3. Shaded zone set $Z^*(d_1) = \{(\{B\}, \{A\})\}$.
4. The set of existential spider descriptors

$$ESD(d_1) = \{(1, \{(\{B\}, \{A\})\}), (1, \{(\{A\}, \{B\}), (\{B\}, \{A\})\})\}.$$

5. Constant spider label set $CS(d_1) = \{s, t\}$.

[2]However, for any diagram that incorporated such ties it is possible to define a semantically equivalent diagram that does not contain such ties. This is not the case for ties between constant spiders. It is straightforward to extend the work in this paper to the case where these additional types of tie are permitted.

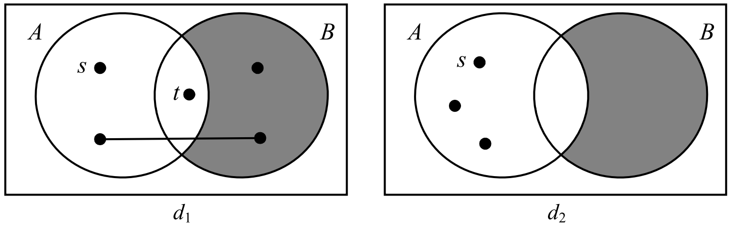


Fig. 4 Two spider diagrams with constants

6. The function $\theta_{d_1} : \{s, t\} \to R(d_1)$ where $\theta_{d_1}(s) = \{(\{A\}, \{B\})\}$ and $\theta_{d_1}(t) = \{(\{A, B\}, \emptyset)\}$.
7. The function $\omega_{d_1} : CS(d_1) \times CS(d_1) \to \mathbb{P}Z(d_1)$ where $\omega_{d_1}(s, s) = \theta_{d_1}(s)$, $\omega_{d_1}(t, t) = \theta_{d_1}(t)$ and $\omega_{d_1}(s, t) = \omega_{d_1}(t, s) = \emptyset$.

Now we introduce some terminology and notation on top of the concepts formalized in the definition. An existential spider descriptor (n, r) is intended to mean that there are precisely n existential spiders placed in the zones in the region r, and we can think of these being numbered from 1 to n. A typical such spider will be spider i, which we denote by $e_i(r)$, to avoid confusion with the notation (i, r) used for existential spider descriptors. The set of **existential spiders** in a unitary diagram d is given by

$$ES(d) = \left\{e_i(r) : \exists(n, r) \in ESD(d) \wedge 1 \le i \le n\right\}.$$

We also define $S(d) = ES(d) \cup CS(d)$ to be the set of **spiders** in d. We assume that the sets $ES(d)$ and $\mathcal{CS} \cup \mathcal{CL} \cup \mathcal{Z} \cup \mathcal{R}$ are disjoint. We also define a function

$$\eta : ES(d) \to R(d)$$

which returns the **habitat** of each existential spider, so that $\eta(e_i(r)) = r$.

Spiders represent the existence of elements and regions represent sets—thus we need to know how many elements are represented in each region. Note here that, in a unitary diagram, a constant spider and an existential spider represent the existence of distinct elements. For example, in Fig. 4, the diagram d_2 asserts that the set represented by the zone $(\{A\}, \{B\})$ contains at least three elements, including the individual represented by s. The set of existential spiders contained by region r in d is denoted by $ES(r, d)$. More formally,

$$ES(r, d) = \left\{e \in ES(d) : \eta(e) \subseteq r\right\}.$$

Similarly, the set of constant spiders contained by region r in d is

$$CS(r, d) = \left\{s \in CS(d) : \theta_d(s) \subseteq r\right\}$$

and we also define

$$S(r, d) = ES(r, d) \cup CS(r, d).$$

So, any spider in d whose habitat is a subset of r is in the set $S(r, d)$. The set of existential spiders touching r in d is denoted by $ET(r, d)$. More formally,

$$ET(r, d) = \left\{s \in ES(d) : \eta(s) \cap r \neq \emptyset\right\}.$$

Moreover, in a shaded region there is an upper bound on the cardinality of the represented set. For example, d_1 in Fig. 4 tells us that there are at most two elements in $B - A$, because exactly two spiders touch $B - A$. The set of constant spiders touching a region, $CT(r, d)$, and the set of spiders touching a region, $T(r, d)$, are defined similarly. In d_1, Fig. 4,

$$\left|S\big(\big\{(\{B\}, \{A\})\big\}, d_1\big)\right| = 1$$

and

$$\left|T\big(\big\{(\{B\}, \{A\})\big\}, d_1\big)\right| = 2.$$

In d_2,

$$\left|S\big(\big\{(\{A\}, \{B\})\big\}, d_2\big)\right| = \left|T\big(\big\{(\{A\}, \{B\})\big\}, d_2\big)\right| = 3.$$

Unitary diagrams form the building blocks of *compound diagrams*, formed by using logical connectives.

Definition 2.2 An **abstract spider diagram with constants** is defined as follows.

1. Any unitary diagram with constants is a spider diagram with constants.
2. If D_1 and D_2 are spider diagrams with constants then $(D_1 \vee D_2)$ and $(D_1 \wedge D_2)$ are spider diagrams with constants.

Our convention will be to denote unitary diagrams by d and arbitrary diagrams by D. Some compound diagrams are not satisfiable (defined later). For convenience later, we introduce the symbol $\bot$, defined to be a unitary diagram that is not satisfiable.

3 Semantics

We now sketch, informally, the semantics of unitary spider diagrams. Regions represent sets. Missing zones represent the empty set. For example, in diagram d_1 in Fig. 3, the zones $(\{A, C\}, \{B\})$ and $(\{A\}, \{B, C\})$ are missing and so represent the empty set; from this we can deduce that sets represented by A and B are disjoint.

Now, for simplicity, suppose a unitary diagram d does not contain any ties. If region r is inhabited by n spiders in d then d expresses that the set represented by r contains at least n elements. If r is shaded and touched by m spiders in d then d expresses that the set represented by r contains at most m elements. Thus, if d has a shaded, untouched region, r, then d expresses that r represents the empty set. For example, in diagram d_1 in Fig. 3, the shaded region $\{(\{A\}, \{B, C\}), (\{A, C\}, \{B\})\}$ is untouched by any spider and therefore represents the empty set. In diagram d_2 in Fig. 3, the same region is shaded and touched by two spiders and so the set it represents contains at most two elements.

Each constant spider asserts that the individual it represents is in the set represented by its habitat. Moreover, the individuals represented by constant spiders are distinct from those represented by existential spiders. Therefore, if a region contains an existential spider and a constant spider, s, we can deduce that there are at least two elements in that region, including that represented by s. Within a unitary diagram, no two constant spiders represent the same individual unless they are joined by a tie. Constant spiders joined by ties represent the same individual if and only if there exists a zone, z, in their web and they both represent individuals in the set represented by z. So, the presence of a tie between two constant spiders has the effect of potentially reducing the upper and lower cardinality constraints placed on the set represented by the union of their habitats. In diagram d_3 in Fig. 3, the constant spiders s and t represent different individuals unless both the individuals they represent are in the set represented by the zone $(\{A\}, \{B, C\})$ or both are in the set represented by $(\{C\}, \{A, B\})$, in which case they must represent the same individual.

To formalize the semantics of spider diagrams with constants we shall map constant spider labels, contour labels, zones and regions to subsets of some universal set. We wish constant spider labels to act like constants in first-order predicate logic, so they will map to single element subsets of the universal set, unless the universal set is the empty set. We could, equivalently, choose to map constant spiders to elements of the universal set. However, the *semantics predicate* (defined below) is more elegant when we map constant spiders to sets, as are the details of some of the proofs below. Our formalization of the semantics extends that given for spider diagrams without constants in [12].

Definition 3.1 An **interpretation of constant spider labels, contour labels, zones and regions**, or simply an **interpretation**, is a pair (U, Ψ) where U is a set and $\Psi : \mathcal{CL} \cup \mathcal{Z} \cup \mathcal{R} \cup \mathcal{CS} \to \mathbb{P}U$ is a function mapping constant spider labels, contour labels, zones and regions to subsets of U such that the images of the zones and regions are completely determined by the images of the contour labels as follows:

1. for each zone (a, b), $\Psi(a, b) = \bigcap_{l \in a} \Psi(l) \cap \bigcap_{l \in b} \overline{\Psi(l)}$ where $\overline{\Psi(l)} = U - \Psi(l)$ and we define $\bigcap_{l \in \emptyset} \Psi(l) = U = \bigcap_{l \in \emptyset} \overline{\Psi(l)}$ and
2. for each region r, $\Psi(r) = \bigcup_{z \in r} \Psi(z)$ and we define $\Psi(\emptyset) = \bigcup_{z \in \emptyset} \Psi(z) = \emptyset$

and either the universal set is the empty set or the constant spiders map to singleton subsets of U. More formally

$$U = \emptyset \ \vee \ \forall s_i \in \mathcal{CS} \ \left|\Psi(s_i)\right| = 1.$$

We will write $\Psi : \mathcal{R} \cup \mathcal{CS} \to \mathbb{P}U$ when strictly speaking we mean $\Psi : \mathcal{CL} \cup \mathcal{Z} \cup \mathcal{R} \cup \mathcal{CS} \to \mathbb{P}U$.

We introduce a *semantics predicate* which identifies whether a diagram expresses a true statement, with respect to an interpretation.

Definition 3.2 Let D be a spider diagram with constants and let $m = (U, \Psi)$ be an interpretation. We define the **semantics predicate** of D, denoted $P_D(m)$. If $D = \perp$ then $P_D(m)$ is $\perp$. If D $(\neq \perp)$ is a unitary diagram then $P_D(m)$ is the conjunction of the following conditions.

1. **Plane Tiling Condition.** The union of the sets represented by the zones in D is the universal set: $\bigcup_{z \in Z(D)} \Psi(z) = U$.
2. There exists an extension of $\Psi : \mathcal{R} \cup \mathcal{CS} \to \mathbb{P}U$ to $\Psi : \mathcal{R} \cup \mathcal{CS} \cup ES(D) \to \mathbb{P}U$ such that the following conditions are satisfied.
   (a) **Spiders Condition.** Each spider represents the existence of an element (strictly, a single element set) in the set represented by its habitat and existential spiders do not represent the same elements as any constant spiders:

$$\forall s \in ES(D) \ \left(\left|\Psi(s)\right| = 1 \wedge \Psi(s) \subseteq \Psi\big(\eta(s)\big)\right)$$

and

$$\forall s \in CS(D)\ \left(\left|\Psi(s)\right| = 1 \wedge \Psi(s) \subseteq \Psi\left(\theta_D(s)\right)\right)$$

and

$$\forall e \in ES(D)\ \forall s_i \in CS(D)\ \Psi(e) \neq \Psi(s_i).$$

(b) **Existential Spiders Condition.** No two existential spiders represent the existence of the same element:

$$\forall e_1, e_2 \in ES(D)\ \left(\Psi(e_1) = \Psi(e_2) \Rightarrow e_1 = e_2\right).$$

That is, the function Ψ is injective when the domain is restricted to $ES(d)$.

(c) **Constant Spiders Condition.** Two constant spiders represent the same individual if and only if they both represent an individual in the set denoted by some zone in their web:

$$\begin{aligned}&\forall s_i, s_j \in CS(D)\ \left(\Psi(s_i) = \Psi(s_j)\right.\\ &\quad \Leftrightarrow\ \left.\exists z \in \omega_D(s_i, s_j)\ \Psi(s_i) \cup \Psi(s_j) \subseteq \Psi(z)\right).\end{aligned}$$

(d) **Shading Condition.** Each shaded zone, z, represents a subset of the set of elements represented by the spiders touching z:

$$\forall z \in Z^*(D)\ \Psi(z) \subseteq \bigcup_{s \in T(\{z\}, D)} \Psi(s).$$

If $\Psi : \mathcal{R} \cup ES(D) \rightarrow \mathbb{P}U$ ensures $P_D(m)$ is true then Ψ is a **valid extension to existential spiders** for D. If $D = D_1 \vee D_2$ then $P_D(m) = P_{D_1}(m) \vee P_{D_2}(m)$. If $D = D_1 \wedge D_2$ then $P_D(m) = P_{D_1}(m) \wedge P_{D_2}(m)$. We say m **satisfies** D, or m is a **model** for D, denoted $m \models D$, if and only if $P_D(m)$ is true. If all the models for D_1 are models for D_2, then D_1 **semantically entails** D_2, denoted $D_1 \vDash D_2$. If $D_1 \vDash D_2$ and $D_2 \vDash D_1$, then D_1 and D_2 are **semantically equivalent**, denoted $D_1 \equiv_{\vDash} D_2$.

As an example, the interpretation $m = (\{1, 2, 3, 4\}, \Psi)$ partially defined by $\Psi(s_1) = \{1\}$, $\Psi(s_2) = \{2\}$, $\Psi(L_1) = \{1, 2\}$ and $\Psi(L_2) = \{2, 3, 4\}$ is a model for d_1 in Fig. 4 but not for d_2.

Theorem 3.3 *Let d $(\neq \perp)$ be a unitary spider diagram with constants. Then d is satisfiable.*

The proof strategy is to construct an interpretation that we call a *standard model* for d, following a similar approach to that for spider diagrams without constants in [12]. Essentially, this contains only the elements that are forced to exist by the presence of spiders in the diagram: for each spider in the diagram we choose one the zones in its habitat and place an element there; in extending this construction to constants we just have to make sure that these elements are identified when ties require that to be so. It is straightforward to show that any standard model for d satisfies d. This standard model is also used in the proof of completeness. More formally, a standard model is defined as follows:

Definition 3.4 Let d be a unitary spider diagram with constants. Let $f : S(d) \rightarrow Z(d)$ be a function such that for each spider s, $f(s)$ is in the habitat of s. For each constant spider, s_i, we define

$$[s_i] = \{s_j \in CS(d) : f(s_j) = f(s_i) \wedge f(s_i) \subseteq \omega_d(s_i, s_j)\}$$

(these sets $[s_i]$ give rise to an equivalence relation and, hence, form a partition of $CS(d)$). Define

$$U = ES(d) \cup \{[s_i] : s_i \in CS(d)\}.$$

For each contour label, L, in d define

$$\Psi(L) = \{e \in ES(d) : f(e) = (in, out) \wedge L \in in\} \cup \{[s_i] : s_i \in CS(d) \wedge f(s_i) = (in, out) \wedge L \in in\}$$

and each constant spider, s_k, in d, maps to the set

$$\Psi(s_k) = \{[s_k]\}.$$

Then (U, Ψ) is a **standard model** for d.

4 Reasoning Rules

We will now develop a set of sound and complete reasoning rules for spider diagrams with constants. All of the reasoning rules given for spider diagrams without constants in [12] can be extended—sometimes in a non-trivial way—to spider diagrams with constants; we omit most of the formal definitions of the extended rules.

4.1 Unitary to Unitary Reasoning Rules

In this section we introduce a collection of reasoning rules that apply to, and result in, a unitary diagram.

Rule 1 (*Introduction of a shaded zone*) Let d_1 be a unitary diagram that has a missing zone. If d_2 is the same as d_1 except that d_2 contains a new, shaded and 'untouched' zone then d_1 is logically equivalent to d_2.

In Fig. 5, Rule 1 (introduction of a shaded zone) is applied to d_1 to give d_2. Applying the introduction of a shaded zone rule results in a semantically equivalent diagram. The next two rules are not information preserving.

Rule 2 (*Erasure of shading*) Let d_1 be a unitary diagram with a shaded region r. Let d_2 be identical to d_1 except that r is completely non-shaded in d_2. Then d_1 logically entails d_2.

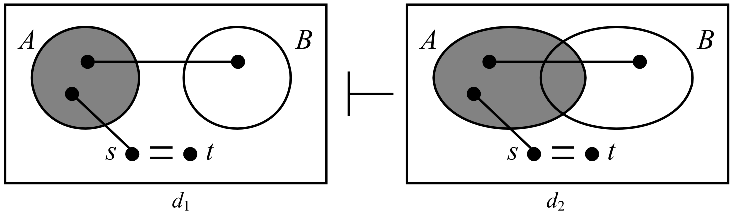


Fig. 5 An application of Rule 1 (introduction of a shaded zone)

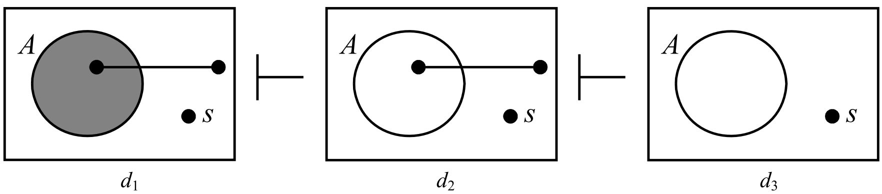


Fig. 6 Applications of Rule 2 (erasure of shading) and Rule 3 (erasure of a spider)

In Fig. 6, Rule 2 (erasure of shading) is applied to d_1 to give d_2.

Rule 3 (*Erasure of a spider*) Let d_1 be a unitary diagram containing a spider s with a completely non-shaded habitat. Let d_2 the same as d_1 except that d_2 does not contain s or any ties that were connected to s. Then d_1 logically entails d_2.

In Fig. 6, Rule 3 (erasure of a spider) is applied to d_2 to give d_3.

4.2 *Unitary to Compound Reasoning Rules*

We now specify five further rules, each of which is reversible, that allow a unitary diagram to be replaced by a compound diagram. The first of these rules allows us to introduce a contour. In the logic for spider diagrams without constants, the introduction of a contour rule applies to, and results in, a unitary diagram [12].

Before we formulate the introduction of a contour rule, we look at an example. In Fig. 7, we examine how to introduce the contour with label C to d_1, which contains constant spiders. When we do so, each zone must split into two new zones, thus ensuring that information is preserved. The habitats of the existential spiders are similarly altered. More care must be taken with the constant spiders, however, due to the presence of ties. Consider, for example, the constant spiders s and t. The individual represented by both s and t must be either in $C - (A \cup B)$ or in $U - (A \cup B \cup C)$. The constant spider u represents an individual that is either in $A - (B \cup C)$ or $(A \cap C) - B$. This gives rise to four possibilities, shown in d_2, d_3, d_4 and d_5. We call these four diagrams the C-*extensions* of d_1. The diagram d_1 is semantically equivalent to $d_2 \vee d_3 \vee d_4 \vee d_5$. We could replace d_1 with the disjunction of just two unitary diagrams, each with u having a two zone habitat: $(\{A\}, \{B, C\})$ and $(\{A, C\}, \{B\})$. However, it is not the case that the single unitary diagram d_6 in Fig. 8 is semantically equivalent to d_1. The constant spiders s and t must represent

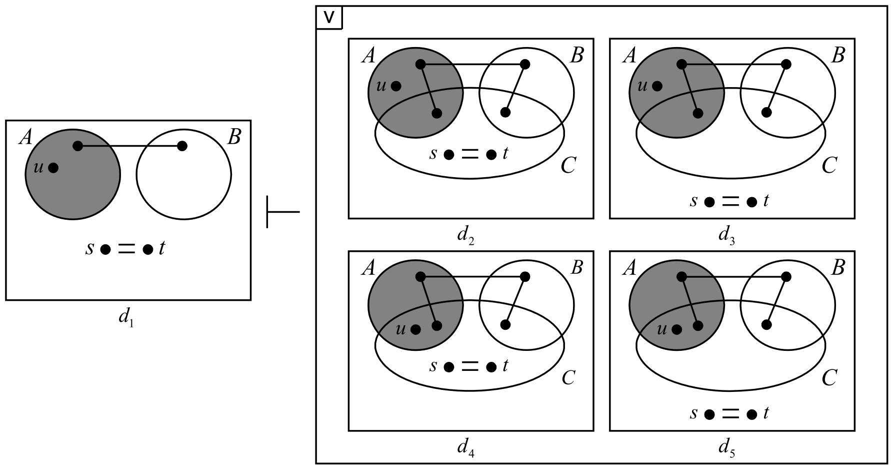
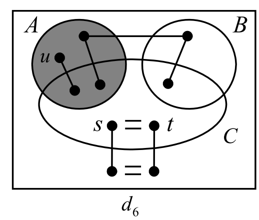


Fig. 7 A diagram with its C-extensions

Fig. 8 Introducing a contour: an incorrect application



the same individual in d_1 but this is not the case in d_6, since the semantics of ties are zone based.

To define this rule formally, we first define the component parts of the resulting disjunction. We call these component parts L_i-*extensions*, where L_i is the contour label introduced.

Definition 4.1 Let d_1 be a unitary diagram such that each constant spider in d_1 has a single zone habitat. Let L_i be a contour label that is not in d_1, that is $L_i \in \mathcal{CL} - L(d_1)$. Let d_2 be a unitary diagram such that each constant spider in d_2 has a single zone habitat. If the following conditions hold then d_2 is an $\boldsymbol{L_i}$**-extension** of d_1.

1. The contour labels of d_2 are those of d_1, together with L_i: $L(d_2) = L(d_1) \cup \{L_i\}$.
2. The constant spider labels match: $CS(d_1) = CS(d_2)$.
3. There exists a surjection, $h : Z(d_2) \to Z(d_1)$ defined by $h(a, b) = (a - \{L_i\}, b - \{L_i\})$ such that
   (a) each zone in d_1 is mapped to by two distinct zones in d_2,
   (b) each zone is shaded in d_2 if and only if it maps to a shaded zone,
   (c) the existential spiders match and their habitats are preserved under h: there exists a bijection, $\sigma : ES(d_1) \to ES(d_2)$ that satisfies

$$\forall e \in ES(d_1)\ \eta\big(\sigma(e)\big) = \big\{z \in Z(d_2) : h(z) \in \eta(e)\big\},$$

   and

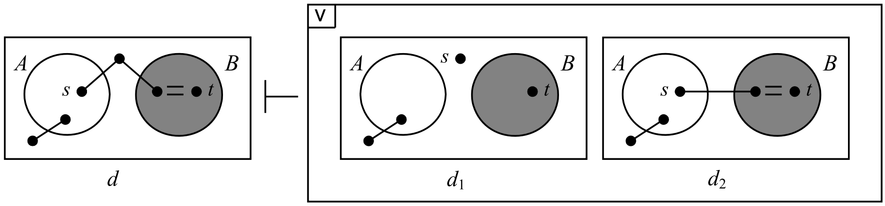


Fig. 9 An application of Rule 5, splitting spiders

(d) the habitat of each constant spider, c, in d_2 satisfies $h(\theta_{d_2}(c)) = \theta_{d_1}(c)$.

4. Spider webs are preserved. Since the constant spiders have a single zone habitat we may formalize this as follows:

$$\forall c_1, c_2 \in CS(d_2)\ \big(\omega_{d_1}(c_1, c_2) \neq \emptyset \Leftrightarrow \omega_{d_2}(c_1, c_2) \neq \emptyset\big).$$

We define $\mathcal{EXT}(L_i, d_1)$ to be the set of all L_i-extensions of d_1.

Rule 4 (*Introduction of a contour label*) Let d_1 $(\neq \bot)$ be a unitary diagram such that each constant spider has a single zone habitat. Let $L_i \in \mathcal{CL} - L(d_1)$. Then d_1 is logically equivalent to the diagram

$$\bigvee_{d_2 \in \mathcal{EXT}(L_i, d_1)} d_2.$$

Rule 5 (*Splitting spiders*) Let d be a unitary diagram with a spider s touching every zone of two disjoint regions r_1 and r_2. Let d_1 and d_2 be unitary diagrams that are identical to d except that neither contains s, but instead each contains an extra spider, s_1 and s_2 respectively, whose habitats are regions r_1 in d_1 and r_2 in d_2. If s is a constant spider, then

1. s_1 and s_2 have the same label as s and
2. any ties joined to s in d are joined to the appropriate instance of s in d_1 and d_2.

Then d is logically equivalent to the diagram $d_1 \vee d_2$.

Figure 9 illustrates an application of the splitting spiders rule. The spider s in d splits into two spiders, one in d_1, the other in d_2. Intuitively, the individual represented by s is either in the set $U - (A \cup B)$ or the set $A \cup B$.

Rule 6 (*Excluded middle*) Let d be a unitary diagram with a completely non-shaded region r. Let d_1 and d_2 be unitary diagrams that are the same as d except that d_1 contains an extra existential spider whose habitat is r and in d_2 the region r is shaded. Then d is logically equivalent to the diagram $d_1 \vee d_2$.

For example, the diagram d in Fig. 10 can be replaced by $d_1 \vee d_2$ by applying the excluded middle rule.

Before we introduce the next rule, we look at an example, and then make a definition that is key to formulating the rule itself. Given a unitary diagram, d, that has only non-empty models (in which case d contains at least one spider), we can deduce that the

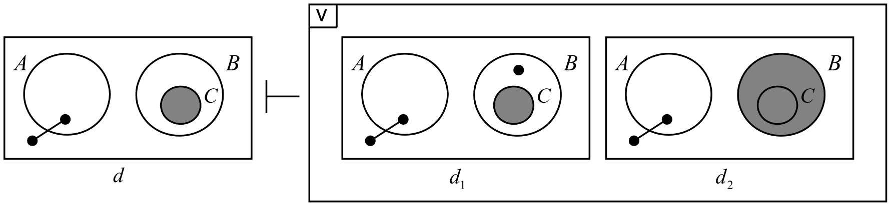


Fig. 10 An application of Rule 6, excluded middle

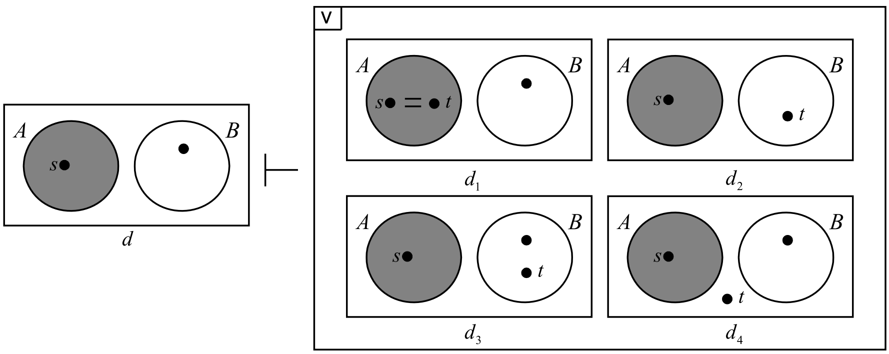


Fig. 11 A unitary diagram with its t-extensions

individual represented by a constant spider label, t, belongs to one of the sets denoted by the zones in d. Moreover, this individual must either be the same as, or different from, the elements already represented in d.

As an example, consider d in Fig. 11 which has only non-empty models. Thus, in any model for d the constant spider (label) t maps to some individual (technically, single element set). Then t is in $A - B$, $B - A$ or $U - (A \cup B)$. If t is in $A - B$ then it must equal s, since the region inside A is entirely shaded, shown in d_1. If t is in the set $B - A$ then it may be either equal to or different from the element represented by the existential spider in B in the diagram d; these cases are represented by d_2 and d_3 respectively. Finally, if t is not in $A - B$ or $B - A$ then, since $A \cap B = \emptyset$, t must be in $U - (A \cup B)$, represented by d_4. The diagrams d_1, d_2, d_3 and d_4 are called *t-extensions* of d. A diagram in which all spiders have a single zone habitat is called an **α-diagram**.

Definition 4.2 Let d_1 be a unitary α-diagram such that $S(d_1) \neq \emptyset$ and there exists $s_i \in \mathcal{CS} - CS(d_1)$. Let d_2 be a unitary α-diagram. If the following conditions are satisfied then d_2 is an **s_i-extension** of d_1.

1. The zones match: $Z(d_1) = Z(d_2)$.
2. The shaded zones match: $Z^*(d_1) = Z^*(d_2)$.
3. The constant spiders match except that s_i is in d_2: $CS(d_1) \cup \{s_i\} = CS(d_2)$.
4. The habitats of the existing constant spiders are preserved: $\theta_{d_1} = \theta_{d_2}|_{CS(d_1)}$.
5. The existing webs are preserved: $\omega_{d_1} = \omega_{d_2}|_{CS(d_1) \times CS(d_1)}$.

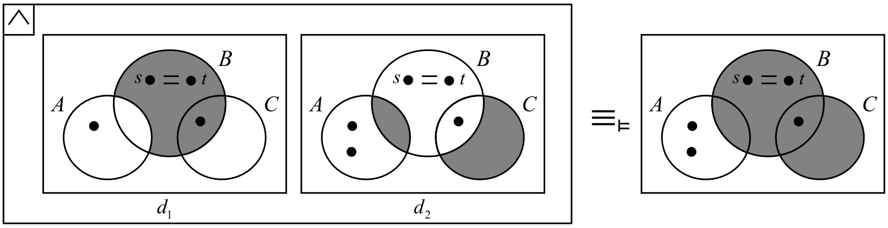


Fig. 12 Combining diagrams

6. If s_i has a shaded habitat, z, in d_2 then either the number of existential spiders inhabiting z is one less than the number in d_1 or s_i is joined to another (constant) spider by a tie: if $\theta_{d_2}(s_i) \subseteq Z^*(d_2)$ then
   (a) $\forall s_j \in CS(d_1)\ \omega_{d_2}(s_i, s_j) = \emptyset \wedge \exists e \in ES(\theta_{d_2}(s_i), d_1)\ ES(d_2) = ES(d_1) - \{e\}$ or
   (b) $\exists s_j \in CS(d_1)\ \omega_{d_2}(s_i, s_j) \neq \emptyset \wedge ES(d_1) = ES(d_2)$.
7. If s_i has a non-shaded habitat in d_2 then either the number of existential spiders inhabiting z is the same as, or one less than the number in d_1 or s_i is joined to another (constant) spider by a tie and the number of existential spiders is the same: if $\theta_{d_2}(s_i) \cap Z^*(d_2) = \emptyset$ then
   (a) $\forall s_j \in CS(d_1)\ \omega_{d_2}(s_i, s_j) = \emptyset \wedge (ES(d_1) = ES(d_2) \vee \exists e \in ES(\theta_{d_2}(s_i), d_1)$ $ES(d_2) = ES(d_1) - \{e\})$ or
   (b) $\exists s_j \in CS(d_1)\ \omega_{d_2}(s_i, s_j) \neq \emptyset \wedge ES(d_1) = ES(d_3)$.

We define $\mathcal{EXT}(s_i, d_1)$ to be the set of all s_i-extensions of d_1.

Rule 7 (*Introduction of a constant spider*) Let d_1 be a unitary α-diagram such that $S(d_1) \neq \emptyset$ and there exists $s_i \in \mathcal{CS} - CS(d_1)$. Then d_1 is logically equivalent to the diagram

$$\bigvee_{d_2 \in \mathcal{EXT}(s_i, d_1)} d_2.$$

Introducing the constant spider t to d in Fig. 11, results in $d_1 \vee d_2 \vee d_3 \vee d_4$.

The final rule in this section, called *combining*, replaces two unitary α-diagrams, with the same zone sets and constant spider label sets, taken in conjunction by a single unitary diagram, illustrated in Fig. 12. We combine $d_1 \wedge d_2$ to give d^*. Any shading in either d_1 or d_2 occurs in d^*. Moreover, the number of spiders in any zone in d^* is the same as the maximum number that occur in that zone in d_1 or d_2. The diagram $d_1 \wedge d_2$ is semantically equivalent to d^*.

We now give a further example in a build-up to the definition of the combining rule. In Fig. 13, d_1 and d_2 contain contradictory information. We observe the following.

1. The zone $z_1 = (\{A\}, \{B, C\})$ is shaded in d_1 and contains more spiders in d_2. Moreover, z_1 represents the empty set in any model for d_1. In any model for d_2, z_1 does not represent the empty set.
2. The constant spider u has different habitats in the two diagrams. In any model for d_1, u represents an individual that is not in the set $A \cup C$. In any model for d_2, u represents an individual in the set C.

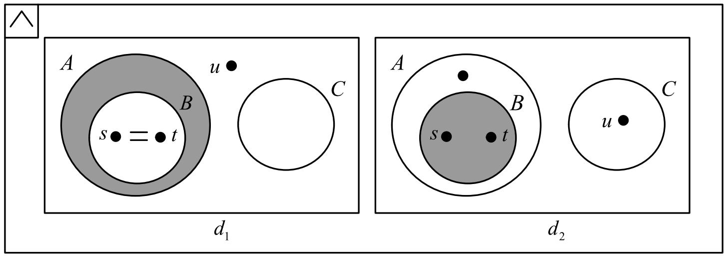


Fig. 13 An unsatisfiable diagram

3. The constant spiders s and t are joined by a tie in d_1 but not in d_2. In any model for d_1, s and t represent the same individual, but in any model for d_2 they represent distinct individuals.

From any one of these three observations we can deduce that $d_1 \wedge d_2$ is unsatisfiable.

Definition 4.3 Let d_0 and d_1 be unitary α-diagrams. Then d_0 and d_1 are **comparable** if one of the following three conditions holds.

1. $Z(d_0) = Z(d_1)$ and $CS(d_0) = CS(d_1)$.
2. $Z(d_0) = Z(d_1)$.
3. for one of the d_is where $i \in \{0, 1\}$, $Z^*(d_i) = Z(d_i)$ and $S(d_i) = \emptyset$.
4. $d_0 = \bot$ or $d_1 = \bot$.

Recall that $S(\{z\}, d) = \{s \in S(d) : \eta(s) = \{z\}\}$.

Definition 4.4 Let d_0 and d_1 be comparable unitary α-diagrams. Then d_0 and d_1 are in **contradiction** if one of the following four conditions holds.

(i) Either $d_0 = \bot$ or $d_1 = \bot$.
(ii) There is a zone that is shaded in one diagram and contains more spiders in the other. More formally, there exists $z \in Z(d_i)$ for some $i = 0, 1$ such that $z \in Z^*(d_j)$ and $|S(\{z\}, d_i)| > |S(\{z\}, d_j)|$ where $j = 1 - i$.
(iii) There is a constant spider with different habitats in d_0 and d_1. More formally, $\theta_{d_0} \neq \theta_{d_1}$.
(iv) There are two constant spiders that are joined by a tie in one diagram but not the other. More formally, $\omega_{d_0} \neq \omega_{d_1}$.

It may be helpful to note that if d_0 and d_1 are comparable and not in contradiction then $\omega(d_0) = \omega(d_1)$.

Lemma 4.5 *Let d_0 and d_1 be comparable unitary α-diagrams. Then d_0 and d_1 are in contradiction if and only if $d_0 \wedge d_1$ is unsatisfiable.*

Definition 4.6 Let d_0 and d_1 be comparable unitary α-diagrams. Then their **combination**, denoted $d^* = d_0 * d_1$, is a unitary α-diagram defined as follows.

1. If d_0 and d_1 are in contradiction then $d_0 * d_1 = \bot$.

2. Otherwise $d^* = d_0 * d_1$ is a unitary α-diagram such that the following hold.
   (a) The set of zones in the combined diagram is the same as the set of zones in the original diagrams: $Z(d^*) = Z(d_0)$.
   (b) The shaded zones in $d^* = d_0 * d_1$ are those that are shaded in at least one of the original diagrams: $Z^*(d^*) = Z^*(d_0) \cup Z^*(d_1)$.
   (c) The number of existential spiders in any zone in the combined diagram is the maximum number of existential spiders inhabiting that zone in the original diagrams:

$$\forall z \in Z\big(d^*\big)\ ES\big(\{z\}, d^*\big) = ES\big(\{z\}, d_0\big) \cup ES\big(\{z\}, d_1\big).$$

   Equivalently, $ES(d^*) = ES(d_0) \cup ES(d_1)$.
   (d) The constant spiders in the combined diagram are the same as those in the original diagrams: $CS(d^*) = CS(d_0)$.
   (e) The habitats of the constant spiders in the combined diagram are the same as those in the original diagrams: $\theta_{d^*} = \theta_{d_0}$.
   (f) The webs of the constant spiders in the combined diagram are the same as those in the original diagrams: $\omega(d^*) = \omega(d_0)$.

Rule 8 (*Combining*) Let d_0 and d_1 be comparable unitary α-diagrams. Then $d_0 \wedge d_1$ is logically equivalent to $d_0 * d_1$.

4.3 Logic Reasoning Rules

We now introduce a collection of rules, all of which have (obvious) analogies in symbolic logic. The next rule is analogous to $P \vdash P \vee Q$, for any propositions P, Q.

Rule 9 (*Connecting a diagram*) Let D_1 and D_2 be spider diagrams. Then D_1 logically entails $D_1 \vee D_2$.

Rule 10 (*Inconsistency*) The diagram $\perp$ logically entails any diagram.

Rule 11 ($\vee$-*Idempotency*) Any spider diagram D is logically equivalent to $D \vee D$.

Rule 12 ($\wedge$-*Idempotency*) Any spider diagram D is logically equivalent to $D \wedge D$.

Rule 13 ($\vee$-*Commutativity*) Let D_1 and D_2 be spider diagrams. Then $D_1 \vee D_2$ is logically equivalent to $D_2 \vee D_1$.

Rule 14 ($\wedge$-*Commutativity*) Let D_1 and D_2 be spider diagrams. Then $D_1 \wedge D_2$ is logically equivalent to $D_2 \wedge D_1$.

Rule 15 ($\vee$-*Associativity*) Let D_1, D_2 and D_3 be spider diagrams. Then $D_1 \vee (D_2 \vee D_3)$ is logically equivalent to $(D_1 \vee D_2) \vee D_3$.

Rule 16 ($\wedge$-*Associativity*) Let D_1, D_2 and D_3 be spider diagrams. Then $D_1 \wedge (D_2 \wedge D_3)$ is logically equivalent to $(D_1 \wedge D_2) \wedge D_3$.

Rule 17 (∨-*Distributivity*) Let D_1, D_2 and D_3 be spider diagrams. Then $D_1 \vee (D_2 \wedge D_3)$ is logically equivalent to $(D_1 \vee D_2) \wedge (D_1 \vee D_3)$.

Rule 18 (∧-*Distributivity*) Let D_1, D_2 and D_3 be spider diagrams. Then $D_1 \wedge (D_2 \vee D_3)$ is logically equivalent to $(D_1 \wedge D_2) \vee (D_1 \wedge D_3)$.

Rule 19 (∨-*Simplification*) Let D_1, D_2 and D_3 be spider diagrams. If diagram D_2 can be transformed into diagram D_3 by one of reasoning rules then $D_1 \vee D_2$ logically entails $D_1 \vee D_3$.

Rule 20 (∧-*Simplification*) Let D_1, D_2 and D_3 be spider diagrams. If diagram D_2 can be transformed into diagram D_3 by one of the reasoning rules then $D_1 \wedge D_2$ logically entails $D_1 \wedge D_3$.

4.4 Obtainability

To conclude this section on reasoning rules we define obtainability.

Definition 4.7 Let D_1 and D_2 be two spider diagrams with constants. Diagram D_2 is **obtainable** from D_1, denoted $D_1 \vdash D_2$, if and only if there is a sequence of diagrams $\langle D^1, D^2, \ldots, D^m \rangle$ such that $D^1 = D_1$, $D^m = D_2$ and D^{k+1} can be obtained from D^k (where $1 \leq k < m$) by applying a reasoning rule. If $D_1 \vdash D_2$ and $D_2 \vdash D_2$, we write $D_1 \equiv_{\vdash} D_2$.

5 Soundness

In this section we show the soundness of the logic of spider diagrams with constants introduced in Sect. 4.

To prove that the system is sound, the strategy is to start by showing that each of the reasoning rules is sound. We show that the introduction of a constant spider rule is sound as an illustration but omit the remaining proofs. The soundness theorem then follows by a simple induction argument.

Lemma 5.1 *Rule 7 (introduction of a constant spider) is sound. Let d_1 be unitary α-diagram such that $S(d_1) \neq \emptyset$ and there exists $s_i \in \mathcal{CS} - CS(d_1)$. Then*

$$d_1 \equiv_{\models} \bigvee_{d_2 \in \mathcal{EXT}(s_i, d_1)} d_2.$$

Proof Let $m = (U, \Psi)$ be an interpretation. Assume that $m \models d_1$. We will show that $m \models d_2$, for some $d_2 \in \mathcal{EXT}(s_i, d_1)$. Let $\Psi_1 : \mathcal{R} \cup \mathcal{CS} \cup ES(d_1) \to \mathbb{P}U$ be a valid extension to existential spiders for d_1. Using d_1 and Ψ_1, we define a diagram, d_2, as follows.

1. The zones match: $Z(d_1) = Z(d_2)$.

2. The shaded zones match: $Z^*(d_1) = Z^*(d_2)$.
3. The constant spiders in d_1 are in d_2 and, additionally, d_2 contains s_i: $CS(d_1) \cup \{s_i\} = CS(d_2)$.
4. The habitats of the constant spiders match and the habitat of s_i in d_2 is determined by Ψ_1:

$$\theta_{d_1} = \theta_{d_2}|_{CS(d_1)}$$

   and

$$\theta_{d_2}(s_i) = \{z\}$$

   where z is the unique zone in $Z(d_1)$ such that $\Psi(s_i) \subseteq \Psi(z)$. Such a zone exists because the plane tiling condition holds for d_1.
5. The existing webs in d_1 are preserved in d_2: $\omega_{d_1} = \omega_{d_2}|_{CS(d_1)\times CS(d_1)}$.
6. We now consider three cases in order to define the existential spiders (and their habitats) and the remaining webs of d_2.
   (a) *There is an existential spider, s, in d_1 such that $\Psi_1(s) = \Psi(s_i)$.* In this case, we choose $e_n(\{\eta(s)\})$, where $(n, \eta(s)) \in ESD(d_1)$, and we define $ES(d_2) = ES(d_1) - \{e_n(\eta(s))\}$. For the remaining webs, we define, for all $s_j \in CS(d_1)$, $\omega_{d_2}(s_i, s_j) = \emptyset$. We note, by the spiders condition for d_1, $\theta_{d_2}(s_i) = \eta(s)$.
   (b) *There is a constant spider, c, in d_1 such that $\Psi(c) = \Psi(s_i)$.* In this case, $ES(d_1) = ES(d_2)$, and, for the remaining webs, we start by defining $\omega_{d_2}(s_i, c) = \theta_{d_1}(c)$; since d_1 is an α-diagram, $\theta_{d_1}(c)$ is a single zone. It follows that s_i is also joined by a tie to all the constant spiders that are joined to c in d_1 and, by (5) above and the transitivity of ties, not joined by a tie to any other constant spiders. We note, by the spiders condition for d_1, $\theta_{d_2}(s_i) = \theta_{d_1}(c)$.
   (c) *No spider, s, in $S(d_1)$ satisfies $\Psi_1(s) = \Psi(s_i)$.* In this case, we have $ES(d_1) = ES(d_2)$ and for all $c \in CS(d_1)$, $\omega_{d_2}(s_i, c) = \emptyset$.

It is straightforward to verify that d_2 is an s_i-extension of d_1.

We now show that $m \models d_2$. Clearly, the plane tiling condition holds for d_2, since $Z(d_1) = Z(d_2)$. If case 6(a) holds then we suppose, without loss of generality, that $s = e_n(\eta(s))$. If either case 6(b) or 6(c) holds then no supposition is necessary. We define an extension of Ψ to the existential spiders in d_2 by $\Psi_2 = \Psi_1|_{\mathcal{R}\cup\mathcal{CS}\cup ES(d_2)}$. The function Ψ_2 is a valid extension of Ψ to existential spiders for d_2. Hence $m \models d_2$, and it follows that

$$d_1 \vDash \bigvee_{d_2 \in \mathcal{EXT}(s_i, d_1)} d_2.$$

For the converse, it can be shown that each $d_2 \in \mathcal{EXT}(s_i, d_1)$ satisfies $d_2 \vDash d_1$. Assuming that $m \models d_2$, the proof strategy is to take a valid extension of Ψ to existential spiders for d_2 and use this to construct a valid extension of Ψ to existential spiders for d_1. Thus,

$$\bigvee_{d_2 \in \mathcal{EXT}(s_i, d_1)} d_2 \vDash d_1.$$

Hence

$$d_1 \equiv_{\models} \bigvee_{d_2 \in \mathcal{EXT}(s_i, d_1)} d_2,$$

that is, Rule 7 (introduction of a constant spider) is sound. □

Theorem 5.2 (*Soundness*) *Let D_1 and D_2 be spider diagrams. If $D_1 \vdash D_2$ then $D_1 \models D_2$.*

Proof The proof is by induction on the length, n, of a sequence establishing $D_1 \vdash D_2$, since each individual step can be shown to be sound along the lines of the proof of Lemma 5.1 above. □

6 Completeness and Decidability

In this section we show the completeness and decidability of the logic of spider diagrams with constants introduced in Sect. 4. We begin with an informal overview, before giving details of the various stages of the proof.

6.1 Overview

The completeness proof strategy for spider diagrams without constants given in [12] extends to the more general case here. The extended strategy, outlined in Fig. 14, is as follows. Suppose that $D_1 \models D_2$. The aim is to transform D_1 and D_2 into disjunctions of unitary α-diagrams using reversible rules (i.e. those which are logical equivalences) where, roughly speaking, each unitary part has some specified contour label set and constant spider label set.

Firstly, we split the constant spiders in D_1 and D_2 until, in each unitary part, all the constant spiders have a single zone habitat, giving D_1^S and D_2^S respectively. This allows us to add contours to the unitary parts in both D_1^S and D_2^S using the reversible Rule 4 (introduction of a contour label), until each (non-false) unitary part has the same contour label set, L. This gives D_1^L and D_2^L respectively. For the next step, zones are introduced to each unitary part until all (non-false) unitary parts have the same zone set, Z. This is done using the reversible Rule 1 (introduction of a shaded zone) and yields D_1^Z and D_2^Z respectively. Now we obtain α-diagrams using the reversible Rule 5 (splitting spiders), yielding D_1^α and D_2^α respectively. The formalization of the diagrams D_i^L, D_i^Z and D_i^α readily generalize those given in [12] for spider diagrams without constants.

We wish to introduce constant spiders to each side until each unitary part has the same constant spider label set. However, we can only introduce constant spiders when our diagrams contain at least one spider (ensuring non-empty models). Thus the next step we take is to apply the excluded middle rule to both sides until all the (non-false) unitary parts are either entirely shaded or contain at least one spider. The reversible Rule 7 (introduction of a constant spider) is then applied, introducing constant spiders to all unitary parts that contain a spider, until all such unitary parts have some specified constant spider label set, C. This gives D_1^C and D_2^C respectively.

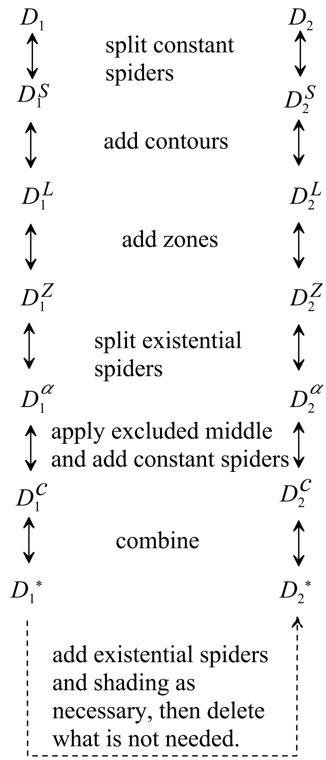


Fig. 14 The completeness proof strategy

We now apply Rule 8 (combining) to remove all the conjuncts, giving two disjunctions of unitary α-diagrams, D_1^* and D_2^*. We call D_1^* (D_2^*) the **disjunctified diagram associated with D_1 (D_2) given D_2 (D_1)**. All of the unitary parts of D_1^* and D_2^* are either

1. $\perp$,
2. have zone set Z and are entirely shaded and contain no spiders, or
3. have zone set Z and constant spider label set C.

Note that $D_1 \equiv_\vdash D_1^*$ and $D_2 \equiv_\vdash D_2^*$, since all the rules applied so far are reversible. The diagram D_i^* is a normal form that reflects the semantics of D_i clearly. We now apply the excluded middle rule to D_1^* until there are sufficiently many existential spiders and there is enough shading to ensure that each unitary part on the left hand side syntactically entails a unitary part of D_2^*.

The details of the proof are given in the following sections. The major differences between the completeness proof strategy here and that for spider diagrams without constants are the addition of the first step (splitting the constant spiders), with knock on changes to details of the other steps, and the insertion of an extra stage between splitting existential spiders and combining diagrams. In addition, we note that the details of the proofs are more complex.

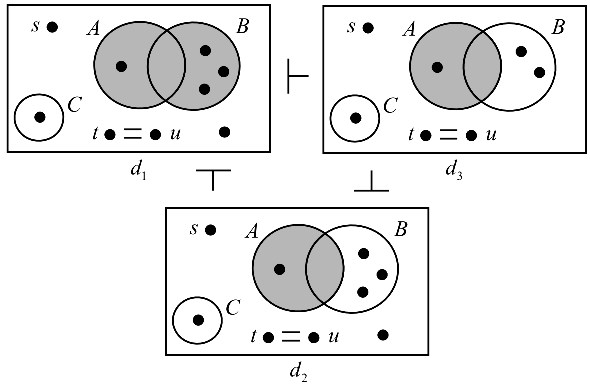


Fig. 15 Completeness for unitary α-diagrams

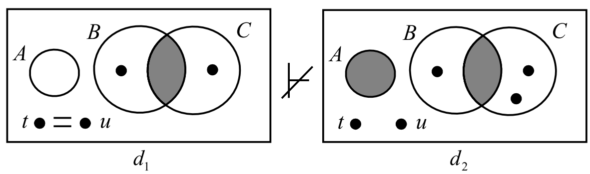


Fig. 16 Completeness for unitary α-diagrams

6.2 Completeness for Unitary α-Diagrams

We show that if $d_1 \vDash d_2$, where d_1 and d_2 are unitary α-diagrams with some fixed zone set and constant spider label set, then we can erase existential spiders and shading from d_1 to give d_2.

Example The diagrams d_1 and d_2 in Fig. 15 satisfy the following.

(a) Every shaded zone in d_2 is shaded in d_1 and contains the same number of existential spiders in both diagrams.
(b) Every zone in d_2 contains the same number or fewer existential spiders than in d_1.
(c) The constant spiders habitats match, as do their webs.

Under these conditions, the diagram d_2 can be obtained from d_1 by applying Rule 2 (erasure of shading), and Rule 3 (erasure of an existential spider) can then be used to give d_3. The properties (a), (b) and (c) above relate to properties 3(a), 3(b) and 3(c) in Theorem 6.1.

Example The diagram d_2 in Fig. 16 cannot be obtained from d_1 for three reasons.

(a) The zone $(\{A\}, \{B, C\})$ is shaded in d_2 but not shaded in d_1. There is a model for d_1 that will cause the shading condition for d_2 to fail whenever the spiders condition for d_2 holds.
(b) The zone $(\{C\}, \{A, B\})$ contains a two existential spiders in d_2 but only a single existential spider in d_1. Again we can deduce that there is a model for d_1 that does not satisfy d_2. For example, at least one model, $m = (U, \Psi)$ for d_1 ensures that $|\Psi(\{C\}, \{A, B\})| = 1$. In the interpretation m, it cannot be that case that both the spiders condition and the existential spiders condition hold for d_2.

(c) The constant spiders t and u have the same habitat in both diagrams, but different webs. In any model for d_1, t and u represent the same individual, but in any model for d_2 they represent distinct individuals.

From any one of the above observations we can deduce that $d_1 \not\vDash d_2$.

The following theorem gives syntactic conditions on unitary α-diagrams equivalent to semantic and syntactic entailment. The theorem forms the heart of the proof of completeness and is modified from the corresponding result in [12] to take account of the fact the our spider diagrams now include constant spiders.

Theorem 6.1 *Let d_1 ($\neq \bot$) and d_2 ($\neq \bot$) be two unitary α-diagrams. If $Z(d_1) = Z(d_2)$ and $CS(d_1) = CS(d_2)$ then the following three statements are equivalent*:

1. $d_1 \vdash d_2$.
2. $d_1 \vDash d_2$.
3. (a) *every zone that is shaded in d_2 is shaded in d_1 and contains the same number of existential spiders in both diagrams*:

$$Z^*(d_2) \subseteq Z^*(d_1) \ \wedge \ \forall z \in Z^*(d_2) \ ES\big(\{z\}, d_2\big) = ES\big(\{z\}, d_1\big),$$

(b) *every zone in d_2 contains at most the same number of existential spiders as in d_1*:

$$\forall z \in Z(d_2) \ ES\big(\{z\}, d_2\big) \subseteq ES\big(\{z\}, d_1\big),$$

and

(c) *the constant spiders have the same habitats and the same webs in both diagrams*: $\theta_{d_1} = \theta_{d_2}$ *and* $\omega_{d_1} = \omega_{d_2}$.

Proof By soundness, $d_1 \vdash d_2 \Rightarrow d_1 \vDash d_2$.

We now show that 2 (i.e., $d_1 \vDash d_2$) implies 3. Suppose that $d_1 \vDash d_2$ and let $m = (U, \Psi)$ be a standard model for d_1. We define, for each existential spider, e_1, in d_1, $\Psi_1(e) = \{e\}$ and the mapping Ψ_1 yields a valid extension to existential spiders for d_1. Since $d_1 \vDash d_2$, m is a model for d_2. Let $\Psi_2 : \mathcal{R} \cup \mathcal{CS} \cup ES(d_2) \to \mathbb{P}U$ be a valid extension to existential spiders for d_2. We will show that Ψ_2 induces an injective, habitat preserving map $\sigma : ES(d_2) \to ES(d_1)$. Now, Ψ_2 ensures that the spiders condition holds for d_2. Therefore, for each existential spider, e_2, in d_2, there exists an existential spider, e_1, in d_1 such that $\Psi_2(e_2) = \{e_1\}$ (each constant spider, s_i, in d_2 maps to $[s_i]$). Define σ by

$$\sigma(e_2) \in \Psi_2(e_2).$$

By the spiders condition for d_1,

$$\big\{\sigma(e_2)\big\} = \Psi_1\big(\sigma(e_2)\big) \subseteq \Psi\Big(\eta\big(\sigma(e_2)\big)\Big)$$

and, by the spiders condition for d_2,

$$\big\{\sigma(e_2)\big\} = \Psi_2(e_2) \subseteq \Psi\big(\eta(e_2)\big).$$

We deduce that, since distinct zones in d_1 represent disjoint sets,

$$\eta\big(\sigma(e_2)\big) = \eta(e_2).$$

Therefore σ is habitat preserving. We now show that σ is injective. Suppose that $\sigma(e_2) = \sigma(e_3)$ for some $e_3 \in ES(d_2)$. Then $\Psi_2(e_2) = \Psi_2(e_3)$, which implies, by the existential spiders condition for d_2, $e_2 = e_3$. Hence σ is injective. We deduce that 3(b) holds. It can also be shown that, for all $z \in Z^*(d_2)$,

$$ES\big(\{z\}, d_2\big) = ES\big(\{z\}, d_1\big).$$

Moreover, it is obvious that $d_1 \vDash d_2$ implies $Z^*(d_2) \subseteq Z^*(d_1)$. Thus 3(a) holds.

We now consider 3(c). The spiders condition for d_1 states, in part,

$$\forall s_i \in CS(d_1)\ \Psi(s_i) \subseteq \Psi\big(\theta_{d_1}(s_i)\big).$$

Since $CS(d_1) = CS(d_2)$, we deduce that

$$\forall s_i \in CS(d_2)\ \Psi(s_i) \subseteq \Psi\big(\theta_{d_1}(s_i)\big). \tag{1}$$

The spiders condition for d_2 states, in part,

$$\forall s_i \in CS(d_2)\ \Psi(s_i) \subseteq \Psi\big(\theta_{d_2}(s_i)\big). \tag{2}$$

Since distinct zones in d_1 represent disjoint sets, it follows from (1) and (2) that

$$\forall s_i \in CS(d_2)\ \theta_{d_1}(s_i) = \theta_{d_2}(s_i).$$

Hence $\theta_{d_1} = \theta_{d_2}$. Suppose that constant spiders s_i and s_j are joined by a tie in d_1. That is,

$$\omega_{d_1}(s_i, s_j) = \theta_{d_1}(s_i).$$

Then $\Psi(s_i) = \Psi(s_j)$, by the constant spiders condition for d_1. By the constant spiders condition for d_2,

$$\exists z \in \omega_{d_2}(s_i, s_j)\ \Psi(s_i) = \Psi(s_j).$$

Therefore, s_i and s_j are joined by a tie in d_2. That is,

$$\omega_{d_2}(s_i, s_j) = \theta_{d_2}(s_i) = \theta_{d_1}(s_i).$$

Alternatively, suppose that spiders s_i and s_j are not joined by a tie in d_1. That is,

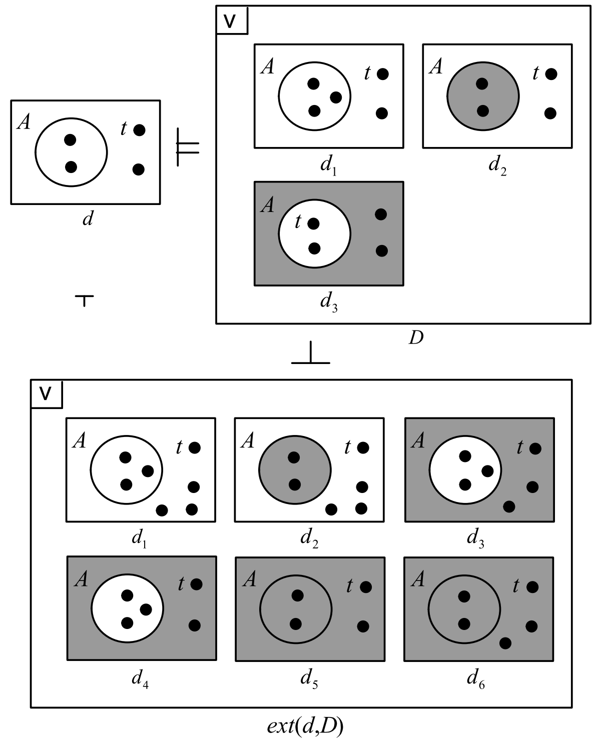
$$\omega_{d_1}(s_i, s_j) = \emptyset.$$

Then $\Psi(s_i) \neq \Psi(s_j)$ so it cannot be that s_i and s_j are joined by a tie in d_2. That is,

$$\omega_{d_2}(s_i, s_j) = \emptyset.$$

Hence $\omega_{d_1} = \omega_{d_2}$. Thus 3(c) holds.

Fig. 17 An α-diagram and an extended diagram



Finally to show that 3 implies 1, it can be shown that shading and existential spiders can be deleted from d_1, using Rules 2 and 3 respectively, to give d_2. Hence all three statements are equivalent. □

6.3 Extended Diagrams

Example In Fig. 17, the diagram D is a semantic consequence of d but no unitary component of D is semantically entailed by d; that is $d \nvDash d_1$, $d \nvDash d_2$ and $d \nvDash d_3$. The diagram $ext(d, D)$ can be obtained from d (and vice versa) by applying Rules 6 (excluded middle) and 19 ($\vee$-simplification). The spiders and shading introduced to d to obtain $ext(d, D)$ are determined by D. For example, consider the outside zone $(\emptyset, \{A\})$. In d_3, this zone is shaded and contains two existential spiders and no other unitary component of D contains more than two existential spiders in this zone. In $ext(d, D)$, this zone contains either one, two or three existential spiders in any unitary component. The process of constructing $ext(d, D)$ will be described in Definitions 6.2 and 6.3 below.

Note that we have

$$d_1' \vDash d_1, \quad d_2' \vDash d_2, \quad d_3' \vDash d_1, \quad d_4' \vDash d_1, \quad d_5' \vDash d_2, \quad \text{and} \quad d_6' \vDash d_2$$

so, for each unitary component d'_i of $ext(d, D)$, there exists a unitary component d_j of D such that $d'_i \vDash d_j$. In fact,

$$d'_1 \vee d'_3 \vee d'_4 \vdash d_1 \quad \text{and} \quad d'_2 \vee d'_5 \vee d'_6 \vdash d_2.$$

Therefore

$$ext(d, D) = d'_1 \vee d'_3 \vee d'_4 \vee d'_2 \vee d'_5 \vee d'_6 \vdash d_1 \vee d_2.$$

By Rule 9 (connecting a diagram) $d_1 \vee d_2 \vdash D$ and by transitivity $ext(d, D) \vdash D$. Therefore $d \vdash D$, since $d \equiv_{\vdash} ext(d, D)$.

In general, the diagram $ext(d, D)$ will be constructed by taking copies of d and adding shading and existential spiders, as specified below. The unitary components of $ext(d, D)$ are called *extended unitary components associated with* d, which we now define. Firstly, we define $comp(D)$ to be the set of all the unitary parts of D.

Definition 6.2 Let d ($\neq \bot$) be a unitary α-diagram and D be an α-diagram. Then, given D, a unitary α-diagram ${}^e d$ is an **extended unitary component associated with** d, denoted $d \sqsubseteq_e^D {}^e d$, if and only if the following seven conditions are satisfied.

1. The diagrams d and ${}^e d$ have the same zones: $Z(d) = Z({}^e d)$.
2. All shading in d occurs in ${}^e d$: $Z^*(d) \subseteq Z^*({}^e d)$.
3. All existential spiders in d occur in ${}^e d$: $ES(d) \subseteq ES({}^e d)$.
4. If zone z is shaded in d then the existential spiders match in d and ${}^e d$: $\forall z \in Z^*(d)$ $ES(\{z\}, d) = ES(\{z\}, {}^e d)$.
5. If zone z is not shaded in d but is shaded in some unitary component of D and the number, m say, of existential spiders that z contains in d is at most the number that z contains in any unitary component of D in which z is shaded then
   (a) if z is shaded in ${}^e d$ then z contains at most m spiders in ${}^e d$; and
   (b) if z is not shaded in ${}^e d$ then z contains $m + 1$ spiders in ${}^e d$.
   More formally:

$$\begin{aligned}
&\forall z \in Z(d) - Z^*(d) \\
&\Bigg(\bigg(z \in \bigcup_{d_i \in comp(D)} Z^*(d_i) \wedge ES\big(\{z\}, d\big) \subseteq \bigcup_{\substack{d_i \in comp(D) \\ z \in Z^*(d_i)}} ES\big(\{z\}, d_i\big)\bigg) \\
&\Rightarrow \Bigg(\bigg(z \in Z^*\big({}^e d\big) \wedge ES\big(\{z\}, {}^e d\big) \subseteq \bigcup_{\substack{d_i \in comp(D) \\ z \in Z^*(d_i)}} ES\big(\{z\}, d_i\big)\bigg) \\
&\quad \vee \bigg(z \in Z\big({}^e d\big) - Z^*\big({}^e d\big) \wedge \big|ES\big(\{z\}, {}^e d\big)\big| = \bigg|\bigcup_{\substack{d_i \in comp(D) \\ z \in Z^*(d_i)}} ES\big(\{z\}, d_i\big)\bigg| + 1\bigg)\Bigg)\Bigg).
\end{aligned}$$

6. If a non-shaded zone z in d is not shaded in any unitary component of D or z contains more spiders in d than any shaded occurrence of z in D then z is not shaded in ${}^e d$ and

z contains the same number of spiders in ${}^e d$ as in d. More formally:

$$\forall z \in Z(d) - Z^*(d)$$
$$\Big(z \notin \bigcup_{d_i \in comp(D)} Z^*(d_i) \vee ES\big(\{z\}, d\big) \supset \bigcup_{\substack{d_i \in comp(D)\\ z \in Z^*(d_i)}} ES\big(\{z\}, d_i\big)\Big)$$
$$\Rightarrow \big(z \in Z\big({}^e d\big) - Z^*\big({}^e d\big) \wedge ES\big(\{z\}, {}^e d\big) = S\big(\{z\}, d\big)\big).$$

7. The constant spiders and their webs match: $CS(d_1) = CS(d_2)$, $\theta_{d_1} = \theta_{d_2}$ and $\omega_{d_1} = \omega_{d_2}$.

If $d = \perp$ then the **extended unitary component associated with** d is $\perp$.

Definition 6.3 Let d be a unitary α-diagram and let D be a disjunction of unitary α-diagrams such that d is comparable to each $d_i \in comp(D)$. Given D, let $\mathcal{D}_e^d$ be the set of all extended unitary components associated with d

$$\mathcal{D}_e^d = \big\{d' \in \mathcal{D}_0 : d \sqsubseteq_e^D d'\big\}.$$

Then the diagram

$$ext(d, D) = \bigvee_{d' \in \mathcal{D}_e^d} d'$$

is the **extended diagram associated with** d **in the context of** D.

Example In Fig. 17, each d_i' $(i = 1, \ldots, 6)$ is an extended unitary component associated with d, given D. Indeed, all such extended components ${}^e d$ are present, so $ext(d, D)$ is the extended diagram associated with d in the context of D.

Theorem 6.4 *Let d be a unitary α-diagram and let D be a disjunction of unitary α-diagrams such that d is comparable to each $d_i \in comp(D)$. Then d is syntactically equivalent to $ext(d, D)$, the extended diagram associated with d in the context of D*:

$$d \equiv_\vdash ext(d, D).$$

Sketch of proof Follows by repeated application of Rules 6 (excluded middle) and 19 ($\vee$-simplification) to d in the case where $d \neq \perp$. When $d = \perp$ the result follows immediately. □

6.4 The Completeness Theorem

The next result is the final prerequisite to our proof of completeness.

Theorem 6.5 *Let d $(\neq \perp)$ be a unitary α-diagram such that $S(d) \neq \emptyset$. Let D be a disjunction of unitary α-diagrams such that d is comparable to each $d_i \in comp(D)$. Given D,* *let* ${}^{e}d \in \mathcal{D}_{e}^{d}$. *If* ${}^{e}d \vDash D$ *then there exists a unitary component of* D, *say* d_i, *such that* ${}^{e}d \vDash d_i$:

$$ {}^{e}d \vDash D \quad \Rightarrow \quad \exists d_i \in comp(D)\ {}^{e}d \vDash d_i. $$

Proof The proof is by contradiction. Assume ${}^{e}d \vDash D$ but there is no $d_i \in comp(D)$ for which ${}^{e}d \vDash d_i$. We will show that a standard model, $m = (U, \Psi)$, for ${}^{e}d$ does not satisfy D, giving the contradiction we seek. The interpretation m does not satisfy D if and only if m does not satisfy any unitary part, d_i, of D. There are three types of d_i to consider.

1. $d_i = \bot$. Clearly m does not satisfy $\bot$.
2. $Z(d) = Z(d_i)$ and $Z^*(d_i) = Z(d_i)$ and $S(d_i) = \emptyset$. Since d contains at least one spider, so too does ${}^{e}d$. Therefore $U \neq \emptyset$. But d_i has only one model: the empty model (that is, $U = \emptyset$). Therefore m does not satisfy d_i.
3. $Z(d) = Z(d_i)$ and $CS(d_i) = CS(d)$ and $S(d_i) \neq \emptyset$. Firstly, suppose that m satisfies d_i and we will reach a contradiction, thus completing the proof that m does not satisfy any unitary part of D. Since m satisfies d_i, it must be that $m \models {}^{e}d \wedge d_i$, so ${}^{e}d$ and d_i are not in contradiction. We immediately deduce, by Lemma 4.5, that the following conditions do not hold.

   (a_1) There is a zone that is shaded in one diagram and contains more spiders in the other diagram. More formally, either

   $$ \exists z \in Z^*({}^{e}d)\ \big|S(\{z\}, d_i)\big| > \big|S(\{z\}, {}^{e}d)\big| $$

   or

   $$ \exists z \in Z^*(d_i)\ \big|S(\{z\}, {}^{e}d)\big| > \big|S(\{z\}, d_i)\big|. $$

   (b_1) There are two constant spiders that are joined by a tie in one diagram but not the other. More formally, $\omega_{d_i} \neq \omega_{{}^{e}d}$.

   Since (b_1) does not hold, we deduce that

   $$ \omega_{d_i} = \omega_{{}^{e}d}. \tag{3} $$

   Since (a_1) does not hold, we deduce that

   $$ \forall z \in Z^*(d_i)\ \big|S(\{z\}, {}^{e}d)\big| \leq \big|S(\{z\}, d_i)\big|. $$

   Since $m \models d_i$, and the fact that d_i is an α-diagram, for all $z \in Z^*(d_i)$,

   $$ \big|\Psi(z)\big| = \big|ES(\{z\}, d_i)\big| + \big|ConS(z, d_i)\big|. \tag{4} $$

   Moreover, if z is not shaded in ${}^{e}d$, then, by the construction of $ext(d, D)$, z contains more existential spiders in ${}^{e}d$ than in d_i:

   $$ \big|ES(\{z\}, {}^{e}d)\big| > \big|ES(\{z\}, d_i)\big|. $$

   So,

   $$ \begin{aligned} \big|\Psi(z)\big| &\geq \big|ES(\{z\}, {}^{e}d)\big| + \big|ConS(z, {}^{e}d)\big| \\ &= \big|ES(\{z\}, {}^{e}d)\big| + \big|ConS(z, d_i)\big| \quad \text{since } \omega_{d_1} = \omega_{{}^{e}d} \end{aligned} $$

$$> \left|ES\big(\{z\}, d_i\big)\right| + \left|ConS(z, d_i)\right|.$$

This contradicts (4). Therefore, it must be that z is shaded in ${}^e d$. Furthermore, it can be shown that $|ES(\{z\}, {}^e d)| = |ES(\{z\}, d_i)|$. Hence

$$\forall z \in Z^*(d_i)\ z \in Z^*\big({}^e d\big) \wedge ES\big(\{z\}, d_i\big) = ES\big(\{z\}, {}^e d\big). \tag{5}$$

Since ${}^e d \nvDash d_i$, by Theorem 6.1 one of the following three conditions holds.

(a$_2$) $\exists z \in Z^*(d_i)\ z \notin Z^*({}^e d) \vee ES(\{z\}, d_i) \neq ES(\{z\}, {}^e d)$.
(b$_2$) $\exists z \in Z(d_i)\ ES(\{z\}, {}^e d) \subset ES(\{z\}, d_i)$.
(c$_2$) $\exists s_i, s_j \in CS(d_1)\ \omega_{d_1}(s_i, s_j) \neq \omega_{d_2}(s_i, s_j)$.

We now consider each of these three possibilities (a$_2$), (b$_2$) and (c$_2$) in turn. Firstly, (a$_2$) contradicts (5) above, so does not hold. Secondly, (c$_2$) contradicts (3) above, so does not hold. Finally we consider (b$_2$). In the model m for ${}^e d$ we have,

$$\left|\Psi(z)\right| = \left|ES\big(\{z\}, {}^e d\big)\right| + \left|ConS\big(z, {}^e d\big)\right|.$$

Now, because m is a model for d_i we have

$$\left|\Psi(z)\right| \geq \left|ES\big(\{z\}, d_i\big)\right| + \left|ConS(z, d_i)\right|$$

from which we deduce that

$$\left|ES\big(\{z\}, {}^e d\big)\right| + \left|ConS\big(z, {}^e d\big)\right| \geq \left|ES\big(\{z\}, d_i\big)\right| + \left|ConS(z, d_i)\right|.$$

Therefore, since $\omega_{d_i} = \omega_{{}^e d}$,

$$\left|ES\big(\{z\}, {}^e d\big)\right| \geq \left|ES\big(\{z\}, d_i\big)\right|.$$

Thus

$$\forall z \in Z(d_i)\ ES\big(\{z\}, d_i\big) \subseteq ES\big(\{z\}, {}^e d\big),$$

which contradicts (b$_2$). Thus in any of the three cases, m does not satisfy d_i.

It follows that the interpretation, m, does not satisfy any unitary part of D. Therefore m does not satisfy D giving a contradiction. Hence if ${}^e d \vDash D$ then there exists a unitary component of D, say d_i, such that ${}^e d \vDash d_i$:

$${}^e d \vDash D \quad \Rightarrow \quad \exists d_i \in comp(D)\ {}^e d \vDash d_i. \qquad \square$$

Theorem 6.6 (*Completeness*) *Let D_1 and D_2 be spider diagrams with constants. Then $D_1 \vDash D_2$ implies $D_1 \vdash D_2$.*

Proof Suppose that $D_1 \vDash D_2$. Let D_1^* be the disjunctified diagram associated with D_1 given D_2. Let D_2^* be the disjunctified diagram associated with D_2 given D_1. To recap, the diagrams D_1^* and D_2^* both have the following properties:

1. they are disjunctions of unitary α-diagrams, and
2. there exists a set of zones Z and a set of constant spider labels C such that each unitary part, d_i satisfies

(a) $d_i = \bot$,
(b) $Z(d_i) = Z$ and $Z^*(d_i) = Z(d_i)$ and $S(d_i) = \emptyset$, or
(c) $Z(d_i) = Z$ and $C(d_i) = C$ and $S(d_i) \neq \emptyset$.

For each unitary part, d_1 of D_1^* obtain the diagram $ext(d_1, D_2^*)$. Since $D_1 \equiv_\vDash D_1^*$, $D_2 \equiv_\vDash D_2^*$ and $D_1 \vDash D_2$ it follows that $d_1 \vDash D_2$. Therefore, $ext(d_1, D_2^*) \vDash D_2^*$. Thus, each unitary part, ${}^e d_1$ of $ext(d_1, D_2^*)$ satisfies ${}^e d_1 \vDash D_2^*$. By Theorem 6.5, ${}^e d_1 \vDash d_2$, for some $d_2 \in comp(D_2^*)$. We now consider three possibilities for d_1.

1. $d_1 = \bot$. In this case, $d_1 = {}^e d$ and it is trivial that $d_1 \vdash d_2$.
2. $Z(d_1) = Z$ and $Z^*(d_1) = Z(d_1)$ and $S(d_1) = \emptyset$. In this case, $d_1 = {}^e d$. Since ${}^e d \vDash D_2^*$, it must be the case that some unitary part, d_2 say, of D_2^* has an empty model. In which case, d_2 does not contain any spiders and so, by the construction of D_2^*, is entirely shaded. Thus $d_2 = {}^e d$ and it is trivial that ${}^e d \vdash d_2$.
3. $Z(d_1) = Z$ and $C(d_1) = C$ and $S(d_1) \neq \emptyset$. In this case, ${}^e d \vdash d_2$ by Theorem 6.1.

In each case, we have shown that ${}^e d \vdash d_2$ and we deduce that ${}^e d \vdash D_2^*$, by Rule 9 (connecting a diagram). It follows that $ext(d_1, D_2^*) \vdash D_2^*$. By transitivity, $d_1 \vdash D_2$. Using Rule 19 ($\vee$-simplification), $D_1^* \vdash D_2^*$. Thus $D_1^* \vdash D_2$. By transitivity, $D_1 \vdash D_2$. Hence the system is complete. □

6.5 Decidability

The proof of completeness provides an algorithmic method for constructing a proof that $D_1 \vdash D_2$ whenever $D_1 \vDash D_2$. It is simple to adapt this algorithm to determine, for any D_1 and D_2, whether $D_1 \vdash D_2$.

Theorem 6.7 (*Decidability*) *There exists an algorithm that determines whether, for any spider diagrams D_1 and D_2, $D_1 \vdash D_2$.*

7 Implementation

We have seen that equality between spider diagrams including constants is decidable, and so it is possible to build computer-based tools that will be able to check decidability, but also which can construct equality proofs when they exist, whether automatically or with user guidance. In this short section we discuss the state of the art in implementing tools for this and other purposes.

The development of tools to support diagrammatic reasoning is well underway, and recent advances provide a basis for automated support for spider diagrams with constants. Such tools require varied functionality and the research challenges can be viewed as more broad than for symbolic logics. There are at least two major differences: first, it is more difficult to parse a 2D diagram than a 1D symbolic sentence; more significantly, when automatically generating proofs, the diagrams must be laid out in order for the user to read the proof. In respect of the second difference, possibly the hardest aspect of spider diagram layout is in the initial generation of the underlying Euler diagram. There have been many recent efforts in this regard, including [1, 5, 15, 19, 26]. Spiders can be automatically added later, as demonstrated in [17].

In terms of automated reasoning, this has been investigated for unitary Euler diagrams [24] and, to some extent, for spider diagrams, for example [7]. The approaches used rely on a heuristic search, guided by a function that provides a lower bound on proof length. Roughly speaking, the better this lower bound, the more efficiently the theorem prover finds proofs. It has been possible to produce better proof search techniques for reasoning with unitary spider diagrams [7] than for compound diagrams [6]. As was demonstrated in [25], the translation of a unitary spider diagram with constants results in (except in trivial cases), a compound diagram. So, it is highly likely to be beneficial, from an automated reasoning perspective, to develop theorem provers for spider diagrams with constants using the rules presented in this paper rather than use translations and subsequently employ theorem provers for spider diagrams. An Euler diagram theorem prover, called EDITH, is freely available for download from http://www.cmis.brighton.ac.uk/research/vmg/autoreas.htm. We note that the main goals of automated reasoning in diagrammatic systems need not include outperforming symbolic theorem provers in terms of speed; of paramount importance is the production of proofs that are accessible to the reader and it may be that this readability constraint has a big impact on the time taken to find a proof.

8 Conclusion

We have provided formal syntax and semantics for the language of spider diagrams with constants and presented a set of reasoning rules for this language. We have shown that the resulting system is sound, complete and decidable. Although the inclusion of constant spiders does not increase expressive power, we believe that if one wishes to make statements about specific individuals then it is natural to do so using constants explicitly. Thus augmenting with constants, although it brings no expressiveness benefits, is likely to increase the usability of the notation. With the reasoning rules developed in this paper, users can reason with the language when constants are included. Such reasoning systems provide an essential basis for permitting diagrams to be used for mathematical formalization and reasoning.

In the future, we plan to investigate the use of constants in notations that extend spider diagrams. These include constraint diagrams [14] and their generalizations [22]. Recent research has begun to develop a variation of constraint diagrams that is suitable for specifying and reasoning about ontologies [13, 18].

Acknowledgement This work is supported by the UK EPSRC grant "Defining Regular Languages with Diagrams" [EP/H012311/1].

References

1. Chow, S., Ruskey, F.: Drawing area-proportional Venn and Euler diagrams. In: Proceedings of Graph Drawing 2003, Perugia, Italy. LNCS, vol. 2912, pp. 466–477. Springer, Berlin (2003)

2. Clark, R.: Failure mode modular de-composition using spider diagrams. In: Proceedings of Euler Diagrams 2004. Electronic Notes in Theor. Comput. Sci., vol. 134, pp. 19–31 (2005)
3. De Chiara, R., Erra, U., Scarano, V.: VennFS: a Venn diagram file manager. In: Proceedings of Information Visualisation, pp. 120–126. IEEE Comput. Soc., Los Alamitos (2003)
4. Euler, L.: Lettres a une princesse d'Allemagne sur divers sujets de physique et de philosophie. Opera Omnia **2**, 102–108 (1775)
5. Flower, J., Howse, J.: Generating Euler diagrams. In: Proceedings of 2nd International Conference on the Theory and Application of Diagrams, Georgia, USA, pp. 61–75. Springer, Callaway Gardens (2002)
6. Flower, J., Masthoff, J., Stapleton, G.: Generating proofs with spider diagrams using heuristics. In: Proceedings of Distributed Multimedia Systems, International Workshop on Visual Languages and Computing, pp. 279–285. Knowledge Systems Institute, San Francisco (2004)
7. Flower, J., Masthoff, J., Stapleton, G.: Generating readable proofs: a heuristic approach to theorem proving with spider diagrams. In: Proceedings of 3rd International Conference on the Theory and Application of Diagrams, Cambridge, UK. LNAI, vol. 2980, pp. 166–181. Springer, Berlin (2004)
8. Gurr, C.: Aligning syntax and semantics in formalisations of visual languages. In: Proceedings of IEEE Symposia on Human-Centric Computing Languages and Environments, pp. 60–61. IEEE Comput. Soc., Los Alamitos (2001)
9. Hayes, P., Eskridge, T., Saavedra, R., Reichherzer, T., Mehrotra, M., Bobrovnikoff, D.: Collaborative knowledge capture in ontologies. In: Proceedings of the 3rd International Conference on Knowledge Capture, pp. 99–106 (2005)
10. Howse, J., Molina, F., Shin, S.-J., Taylor, J.: Type-syntax and token-syntax in diagrammatic systems. In: Proceedings FOIS-2001: 2nd International Conference on Formal Ontology in Information Systems, Maine, USA, pp. 174–185. ACM, New York (2001)
11. Howse, J., Molina, F., Taylor, J., Kent, S., Gil, J.: Spider diagrams: a diagrammatic reasoning system. J. Vis. Lang. Comput. **12**(3), 299–324 (2001)
12. Howse, J., Stapleton, G., Taylor, J.: Spider diagrams. LMS J. Comput. Math. **8**, 145–194 (2005)
13. Howse, J., Stapleton, G., Taylor, K., Chapman, P.: Visualizing ontologies: a case study. In: International Semantic Web Conference 2011. Springer, Bonn (2011)
14. Kent, S.: Constraint diagrams: visualizing invariants in object oriented modelling. In: Proceedings of OOPSLA97, pp. 327–341. ACM, New York (1997)
15. Kestler, H., Muller, A., Kraus, J., Buchholz, M., Gress, T., Kane, D., Zeeberg, B., Weinstein, J.: VennMaster: area-proportional Euler diagrams for functional go analysis of microarrays. BMC Bioinformatics **9**(67) (2008)
16. Lovdahl, J.: Towards a visual editing environment for the languages of the semantic web. PhD thesis, Linkoping University (2002)
17. Mutton, P., Rodgers, P., Flower, J.: Drawing graphs in Euler diagrams. In: Proceedings of 3rd International Conference on the Theory and Application of Diagrams, Cambridge, UK. LNAI, vol. 2980, pp. 66–81. Springer, Berlin (2004)
18. Oliver, I., Howse, J., Stapleton, G., Nuutila, E., Törmä, S.: Visualising and specifying ontologies using diagrammatic logics. In: 5th Australasian Ontologies Workshop. Conf. Res. Pract. Inf. Technol., vol. 112. CRPIT, Melbourne (2009)
19. Rodgers, P., Zhang, L., Fish, A.: General Euler diagram generation. In: International Conference on Theory and Applications of Diagrams, pp. 13–27. Springer, Herrsching (2008)
20. Shimojima, A.: Inferential and expressive capacities of graphical representations: survey and some generalizations. In: Proceedings of 3rd International Conference on the Theory and Application of Diagrams, Cambridge, UK. LNAI, vol. 2980, pp. 18–21. Springer, Berlin (2004)
21. Stapleton, G.: Spider diagrams augmented with constants: a complete system. In: Visual Languages and Computing, pp. 292–299 (2008)
22. Stapleton, G., Delaney, A.: Evaluating and generalizing constraint diagrams. J. Vis. Lang. Comput. **19**(4), 499–521 (2008)
23. Stapleton, G., Thompson, S., Howse, J., Taylor, J.: The expressiveness of spider diagrams. J. Log. Comput. **14**(6), 857–880 (2004)
24. Stapleton, G., Masthoff, J., Flower, J., Fish, A., Southern, J.: Automated theorem proving in Euler diagrams systems. J. Autom. Reason. **39**, 431–470 (2007)

25. Stapleton, G., Taylor, J., Howse, J., Thompson, S.: The expressiveness of spider diagrams augmented with constants. J. Vis. Lang. Comput. **20**(1), 30–49 (2009)
26. Stapleton, G., Rodgers, P., Howse, J., Zhang, L.: Inductively generating Euler diagrams. IEEE Trans. Vis. Comput. Graph. **17**(1), 88–100 (2011)
27. Swoboda, N., Allwein, G.: Using DAG transformations to verify Euler/Venn homogeneous and Euler/Venn FOL heterogeneous rules of inference. J. Softw. Syst. Model. **3**(2), 136–149 (2004)
28. Zhao, Y., Lövdahl, J.: A reuse based method of developing the ontology for e-procurement. In: Proceedings of the Nordic Conference on Web Services, pp. 101–112 (2003)

G. Stapleton (✉) · J. Howse · J. Taylor · P. Chapman
Visual Modelling Group, University of Brighton, Brighton, UK
e-mail: g.e.stapleton@brighton.ac.uk

J. Howse
e-mail: john.howse@brighton.ac.uk

J. Taylor
e-mail: john.taylor@brighton.ac.uk

P. Chapman
e-mail: p.b.chapman@brighton.ac.uk

S. Thompson
School of Computing, University of Kent, Canterbury, UK
e-mail: s.j.thompson@kent.ac.uk