Spider Diagrams: a diagrammatic reasoning system

John Howse, Fernando Molina, John Taylor

School of Computing and Mathematical Sciences

University of Brighton, UK

{John.Howse, F.Molina, John.Taylor}@brighton.ac.uk

Stuart Kent

Computing Laboratory

University of Kent, Canterbury, UK

S.J.H.Kent@ukc.ac.uk

Joseph (Yossi) Gil

Department of Computing Science

Technion–IIT, Haifa 32000, Israel

yogi@cs.technion.ac.il

Abstract

Spider diagrams combine and extend Venn diagrams and Euler circles to express constraints on sets and their relationships with other sets. These diagrams can be used in conjunction with object-oriented modelling notations such as the Unified Modeling Language. This paper summarises the main syntax and semantics of spider diagrams. It also introduces inference rules for reasoning with spider diagrams and a rule for combining spider diagrams. This system is shown to be sound but not complete. Disjunctive diagrams are considered as one way of enriching the system to allow combination of diagrams so that no semantic information is lost. The relationship of this system of spider diagrams to other similar systems, which are known to be sound and complete, is explored briefly.

Keywords Diagrammatic reasoning, visual formalisms.

1. Introduction

Diagrammatic notations involving circles or closed curves, which we will call contours, have been in use since at least the Middle Ages [11]. In the middle of the 18th century, the Swiss mathematician Leonhard Euler introduced the notation we now call Euler circles (or Euler diagrams) [2] for the representation of classical syllogisms. This notation uses the topological properties of enclosure, exclusion and intersection to represent the set-theoretic notions of subset, disjoint sets, and intersection, respectively. The 19th century logician John Venn [16] modified this notation to represent logical propositions. In Venn diagrams all contours must intersect. Moreover, for each non-empty subset of the contours, there must be a connected region of the diagram, such that the contours in this subset intersect at exactly that region. Shading is then used to show that a particular region represents the empty set.

Venn diagrams are expressive as a visual notation for writing constraints on sets and their relationships with other sets, but complicated to draw because all possible intersections have to be drawn and then some regions shaded. Drawing the Venn diagram of four or more sets is quite challenging. More [12], in the late 1950s, developed an algorithm for adding a new contour to a Venn diagram. It is possible to add contours indefinitely, but the contours quickly assume weird and wonderful shapes, and the resulting diagram is very complicated and difficult to follow. Indeed, it is rare to see Venn diagrams of four or more contours. On the other hand, Euler circles are intuitive and easier to draw, but are not as expressive as Venn diagrams because they lack provisions for shading.

An indication of the popularity and intuitiveness of Venn and Euler diagrams is the fact that they are used in elementary schools for teaching set theory as an introduction to mathematics. In fact, it is usually a hybrid of the two notations that is used for teaching purposes; in view of their relative merits, it does seems natural to combine the two notations, by relaxing the demand that all curves in Venn diagrams must intersect or by introducing shading into Euler diagrams. This combined notation forms the basis of *spider diagrams*.

In the 1890s, Peirce modified Venn diagrams by including *X-sequences* to introduce elements and disjunctive information into the system [13]. Recently, full formal semantics and inference rules have been developed for Venn-Peirce diagrams [15] and Euler diagrams [6]; see also [1, 5] for related work. Shin [15] proves soundness and completeness results for two systems of Venn-Peirce diagrams.

In object-oriented software development, diagrammatic modelling notations are used to specify systems. Recently, the Unified Modelling Language (UML) [14] has become the Object Management Group's (OMG) standard for such notations. In UML, constraints, such as invariants, preconditions and postconditions, are expressed using the Object Constraint Language (OCL) [17] which is essentially a stylised, textual form of first-order predicate logic and is part of the UML standard. *Constraint diagrams* [10, 4] provide a diagrammatic notation for expressing constraints and can be used in conjunction with UML and OCL.

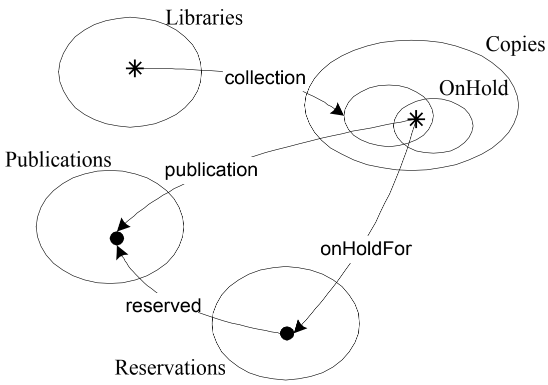


Figure 1.1

The constraint diagram in figure 1.1 expresses (amongst other constraints) an invariant on a model of a library system: for any library object, and any of that library's copies which is on hold, that copy's publication must be the same as that associated with the reservation for which it is on hold. The notation is based on a mixture of Venn and Euler diagrams.

Spider diagrams [3] emerged from work on constraint diagrams. They combine and extend Venn diagrams and Euler circles to express constraints on sets and their relationships with other sets. This paper summarises the syntax and semantics of spider diagrams and extends the diagrammatic inference rules for Venn-Peirce diagrams to spider diagrams.

A more detailed discussion of spider diagrams is conducted in section 2, where the main syntax and semantics of the notation is introduced. Section 3 introduces inference rules for reasoning with spider diagrams and a rule which governs the equivalence of Venn and Euler forms of spider diagrams, and discusses the rule for combining two spider diagrams. Section 4 discusses soundness and completeness of the system and indicates one possible way of enriching the system in order to combine spider diagrams so that no semantic information is lost. Subsystems of spider diagrams which include disjunctive diagrams are shown to be sound and complete.

2. Spider diagrams

This section introduces the main syntax and semantics of spider diagrams; see [3] for more details and examples. Spider diagrams are Euler circles augmented with *shaded regions* and *spiders*. Spider diagrams also include the concepts of *Schrödinger spiders* and *projections*; these are not necessary for this paper and are omitted from this discussion. In [3], the distinction is made between 'given' and 'existential' spiders; a given spider denotes a given element of the corresponding set (in the same way that contours represent given sets) whereas an existential spider denotes existential quantification over the corresponding set. In this paper, all spiders are given (except for the system introduced in section 4.3).

2.1. Syntactic elements of spider diagrams

A *contour* is a simple closed plane curve. A *boundary rectangle* properly contains all other contours. A *district* (or *basic region*) is the bounded area of the plane enclosed by a contour. A *region* is defined as follows: any district is a region; if r_1 and r_2 are regions, then the union, intersection, or difference, of r_1 and r_2 are regions provided these are non-empty. A *zone* (or *minimal region*) is a region having no other region contained within it. Contours and regions denote sets.

A *spider* is a tree with nodes (called *feet*) placed in different zones; nodes are represented by small squares and the connecting edges (called *legs*) are straight lines. A spider *touches* a zone if one of its feet appears in that region. A spider may only touch a zone once. A spider is said to *inhabit* the region which is the union of the zones it touches. For any spider s, the *habitat* of s, denoted $\eta(s)$, is the region inhabited by s. The set of spiders touching region r is denoted by $T(r)$. Spiders are used to denote elements; in this paper, all spiders represent given elements of the corresponding sets. Two distinct spiders denote distinct elements, unless they are joined by a *tie* or by a *strand*.

A *tie* is a double, straight line (an equals sign) connecting two feet, from different spiders, placed in the same zone. Ties indicate equality of the corresponding elements. The *nest* of spiders s and t, written $\tau(s, t)$, is the union of those zones z having the property that the feet of s and t are connected by a tie in z. Two spiders which have a non-empty nest are referred to as *mates*. If both the elements denoted by spiders s and t belong to the set denoted by the same zone in the nest of s and t, then s and t denote the same element.

A *strand* is a wavy line connecting two feet, from different spiders, placed in the same zone. Strands indicate that the corresponding elements *may* (but not necessarily must) be equal. The *web* of spiders s and t, written $\zeta(s, t)$, is the union of zones z having the property that there is a sequence of spiders

$$s = s_0, s_1, s_2, \dots, s_n = t$$

such that, for $i = 0, \dots, n-1$, s_i and s_{i+1} are connected by a tie or by a strand in z. So $\tau(s, t)$ is a subregion of $\zeta(s, t)$. Two spiders with a non-empty web are referred to as *friends*. Two spiders s and t may (but not necessarily must) denote the same element if that element is in the set denoted by the web of s and t. Clearly, if there is a tie between feet, then a strand between those feet is redundant. Similarly, multiple strands or ties between the same pairs of feet are redundant. Thus, on the syntactic level, we allow at most one tie or strand between any pair of feet.

In later sections, we will need to compare regions across diagrams. To facilitate this, we extend the notation and use, for example, $\zeta(s, t, D)$ and $\tau(s, t, D)$ to denote the web and nest respectively of spiders s and t in the diagram D.

Every region is a union of zones. A region is *shaded* if each of its component zones is shaded. A shaded region containing no spiders denotes the empty set. Shading a region r which includes spiders has the effect of placing an upper limit on the number of elements in the set denoted by the region. An upper bound is $|T(r)|$, but this might not be a least upper bound.

A *spider diagram* is a finite collection of contours (exactly one of which must be a boundary contour U), spiders, strands, ties and shaded regions. For any spider diagram D, we use $C = C(D)$, $R = R(D)$, $Z = Z(D)$, $Z^* = Z^*(D)$ and $S = S(D)$ to denote the sets of contours, regions, zones, shaded zones and spiders of D, respectively.

The *Venn form* of a spider diagram contains every possible intersection of contours; otherwise, the diagram is in *Euler form*. A spider diagram with n (non-boundary) contours has 2^n zones if and only if it is in Venn form.

The spider diagram D in figure 2.1 is in Venn form. It has three non-boundary contours A, B, C and two spiders s and t. The label s refers to the whole spider and not just to any particular node. There is a tie between s and t in $(A \cap C) - B$, a strand between s and t in $B - (A \cup C)$ and no syntactic connection between s and t in $(A \cap B) - C$. Hence, if the elements denoted by s and t both belong to $(A \cap C) - B$ then they are equal; if these elements both belong to $B - (A \cup C)$ they may be equal or distinct; and if these elements both belong to $(A \cap B) - C$ then they are distinct. Below are some properties of (the denotation of) D where, for simplicity, we use the same label for a contour and the set it denotes and we use the same label for a spider and the element it denotes.

$$A - (B \cup C) = \varnothing,$$

$$|(B \cap C) - A| \leq 1$$

$$s \in (B - C) \cup (A \cap C - B),$$

$$t \in (B - A \cap B \cap C) \cup (A \cap C - B),$$

$$s, t \in A \cap C - B \Rightarrow s = t,$$

$$s, t \in A \cap B - C \Rightarrow s \neq t.$$

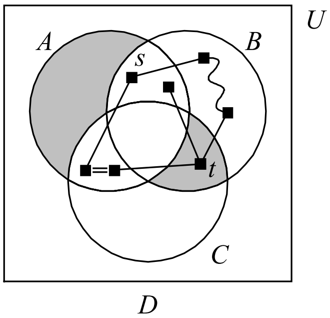


Figure 2.1

2.2. Semantics of spider diagrams

The semantics of a spider diagram D is given in terms of the semantic functions

$$\Psi : C \to Set\,\mathbf{U}, \;\; \psi : S \to \mathbf{U}$$

where $\mathbf{U}$ is a given universal set of D and $Set\,\mathbf{U}$ denotes the power set of $\mathbf{U}$. Contours and regions are interpreted as subsets of $\mathbf{U}$, and spiders as elements of $\mathbf{U}$. The boundary contour is interpreted as $\mathbf{U}$.

A zone is uniquely defined by the contours containing it and the contours not containing it; its interpretation is the intersection of the sets denoted by the contours containing it and the complements of the sets denoted by those contours not containing it. We extend the domain of Ψ to interpret regions as subsets of $\mathbf{U}$. First define $\Psi : Z \to Set\,\mathbf{U}$ by

$$\Psi(z) = \bigcap_{c \in C^+(z)} \Psi(c) \;\cap\; \bigcap_{c \in C^-(z)} \overline{\Psi(c)}$$

where $C^{+}(z)$ is the set of contours containing the zone z, $C^{-}(z)$ is the set of contours not containing z and $\overline{\Psi(c)} = \mathbf{U} - \Psi(c)$, the *complement* of $\Psi(c)$. Since any region is a union of zones, we may define $\Psi: R \to Set\ \mathbf{U}$ by

$$\Psi(r) = \bigcup_{z \in Z(r)} \Psi(z)$$

where, for any region r, $Z(r)$ is the set of zones contained in r.

The semantics of a diagram D is the conjunction of the following conditions.

Plane Tiling Condition: All elements fall within sets denoted by zones:

$$\bigcup_{z \in Z} \Psi(z) = \mathbf{U}$$

Spider Condition: The element denoted by a spider is in the set denoted by the habitat of the spider:

$$\bigwedge_{s \in S} \psi(s) \in \Psi(\eta(s))$$

Strangers Condition: The elements denoted by two distinct spiders are distinct unless they fall within the set denoted by the spiders' web:

$$\bigwedge_{\substack{s,\, t \in S \\ s \neq t}} \left(\psi(s) = \psi(t) \;\Rightarrow\; \psi(s), \psi(t) \in \Psi(\zeta(s, t))\right)$$

Mating Condition: If the elements denoted by two distinct spiders fall within the set denoted by the same zone in the spiders' nest, then the elements are equal:

$$\bigwedge_{s,\, t \in S} \; \bigwedge_{z \in Z(\tau(s, t))} \left(\psi(s), \psi(t) \in \Psi(z) \;\Rightarrow\; \psi(s) = \psi(t)\right)$$

Shading Condition: The set denoted by a shaded zone contains no elements other than those denoted by the spiders:

$$\bigwedge_{z \in Z^*} \left(\Psi(z) \subseteq \bigcup_{s \in S} \{\psi(s)\} \right.$$

We will require the following lemma which follows from the spider and shading conditions.

Lemma 2.1. The set denoted by a shaded zone not containing the feet of any spiders is empty.

3. Reasoning with spider diagrams

In this section we introduce rules for manipulating single diagrams. Except the last rule, each is an inference rule that allows us to obtain one diagram from a given diagram by adding or removing diagrammatic elements. The last rule governs the equivalence of the Euler and Venn forms of spiders diagrams. Throughout this section we use D and D' respectively to denote the diagrams before and after a single application of one of the rules. To link the semantics of the 'before' and 'after' diagrams, we assume that U_D and $U_{D'}$ denote the same (universal) set, which we denote U, and that any two contours or spiders tagged with the same label in D and D' denote the same set or element.

3.1. Rules of transformation

We introduce seven rules for manipulating single diagrams. The first six are inference rules that allow us to obtain one diagram from a given diagram by removing, adding or modifying diagrammatic elements. The last rule governs the equivalence of the Euler and Venn forms of spiders diagrams.

Rule 1: Introduction of a strand. A strand may be drawn between the feet of any two spiders in the same zone. Similarly, any tie may be replaced with a strand.

Example 3.1 Introducing a strand between two non-connected feet in a zone weakens the information contained in the diagram. In figure 3.1, the spiders *s* and *u* in diagram *D* represent distinct elements but in *D′* they may represent the same element of $B - A$.

Similarly, replacing a tie between the feet of two spiders with a strand also weakens the semantic information given by the diagram. If the element denoted by *s* lies in $A - B$, then, in *D*, *s* and *t* are necessarily equal whereas in *D′* they need not be.

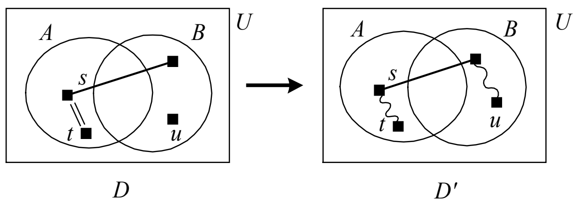


Figure 3.1

Rule 2: Spreading the feet of a spider. If a diagram has a spider *s*, then we may draw a node in any zone *z* which does not contain a foot of *s* and connect it to *s*. If *z* contains the foot of another spider *t*, then we may join the feet of *s* and *t* with a strand or a tie or leave the feet separated in *z*.

Example 3.2 Rule 2 is illustrated by the diagrams in figure 3.2. The inference from *D* to *D′* requires two applications of rule 2, but is clearly valid since it just represents a weakening of information. From *D* we know that the element corresponding to *s* belongs to $A - B$. Having spread its feet in *D′*, we may only infer that this element belongs to $A \cup B$.

In the zone corresponding to $A \cap B$, we have chosen to keep the feet of *s* and *t* separated; in the zone corresponding to $B - A$, we have joined the feet of *s* and *t* with a strand.

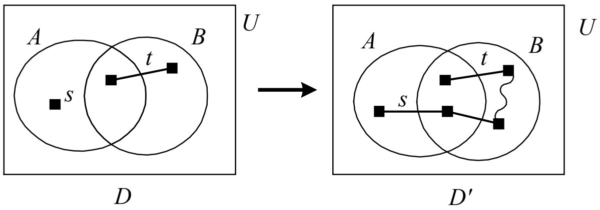


Figure 3.2

Rule 3: Erasure of a spider. We may erase a complete spider on any non-shaded region and any strand or tie connected to it. If removing a spider disconnects any component of the 'strand-tie graph' in a zone, then the components so formed should be reconnected using one or more strands to restore the original component.

Example 3.3 In figure 3.3a, erasing the spider u and its two connecting strands disconnects spiders s and t in the zone $A - B$. However, the web of s and t is the region $A - B$, and this should not change with the deletion of u. Hence in D' the spiders are explicitly 'reconnected' by joining them with a strand. In the diagram D given in figure 3.3b, the elements denoted by spiders s and t need not be equal (unless the element denoted by u belongs to $A \cap \overline{B}$) which is why they are again reconnected by a strand and not a tie in the diagram D'.

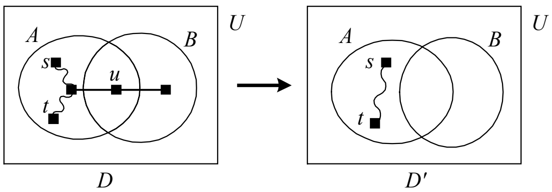


Figure 3.3a

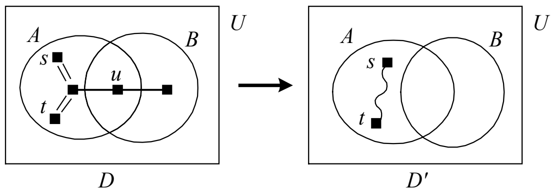


Figure 3.3b

Example 3.4 The requirement that the region from which a spider is removed should be non-shaded is a necessary one. Figure 3.4 illustrates that the removal of a spider from a shaded zone may result in an invalid inference (see section 3.2). In diagram D, the set corresponding to region $A - B$ contains a single element, whereas in D', the corresponding set is empty.

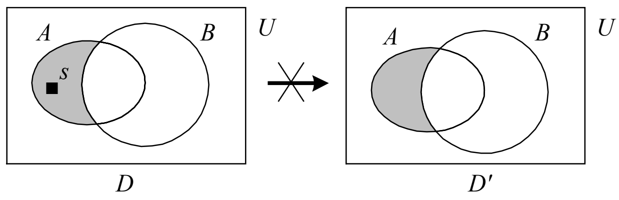


Figure 3.4

Rule 4: Erasure of shading. We may erase the shading in an entire zone.

Example 3.5 In the diagram D given in figure 3.5, the set corresponding to region $A - B$ contains at most a single element, whereas in D', the corresponding set is not constrained.

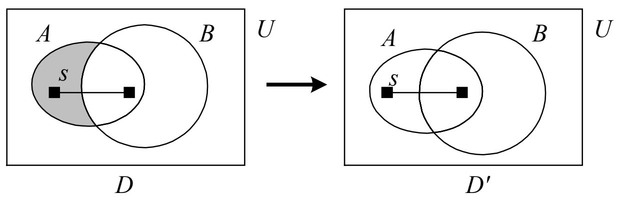


Figure 3.5

Rule 5: Erasure of a contour. We may erase a contour.

When a contour is erased:

- any shading remaining in only a part of a zone should also be erased.
- if a spider has feet in two regions which combine to form a single zone with the erasure of the contour, then these feet are replaced with a single foot connected to the rest of the spider and any ties connecting it in the new zone should be replaced by strands.

Example 3.6 Erasing a contour can cause both syntactic and semantic difficulties.

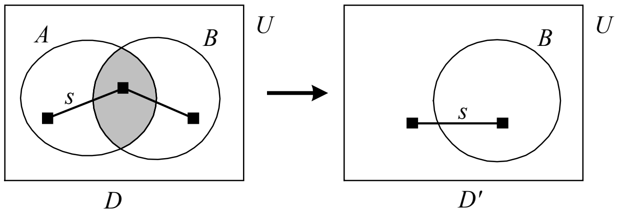


Figure 3.6

Figure 3.6 illustrates the syntactic difficulties. Simply erasing the contour *A* in the diagram *D*, the (new) zone *B* becomes partially shaded and the spider *s* has two feet in the new zone *B*. To ensure that the resulting diagram *D′* is well-formed, the partial shading must be erased and the feet of *s* in *B* should be replaced with a single foot.

The last part of rule 5 concerns semantic difficulties connected with erasing a contour and is a little more subtle.

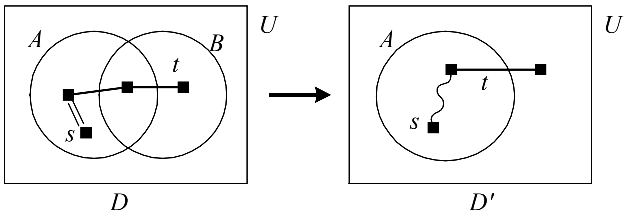


Figure 3.7

Consider the diagram *D* shown in figure 3.7. The diagram has a model in which the elements corresponding to spiders *s* and *t* both belong to the set *A* but are distinct; namely, the model where $s \in A - B$ and $t \in A \cap B$. When the contour *B* is removed, these two zones $A \cap B$ and $A - B$ 'combine' to form the single zone *A* in *D′*. Since it is possible for *s* and *t* to represent distinct elements of *A*, the tie connecting them must be replaced with a strand.

Rule 6: Introduction of a contour. A new contour may be drawn interior to the bounding rectangle observing the partial-overlapping rule: each zone splits into two zones with the introduction of the new contour. If the zone is shaded, then both corresponding new zones are shaded. Each foot of a spider is replaced with a connected pair of feet, one in each new zone. Likewise, each strand or tie bifurcates and becomes a pair of strands or ties, one in each new zone.

Example 3.7 In figure 3.8, a new contour *B* is introduced satisfying the partial overlapping rule. Each zone in *D* becomes a pair of zones in *D′* and each foot of spiders *s*, *t* and *u* bifurcates to become two feet, one in each new zone. The strand and tie also bifurcate. The zones in *D′* corresponding to the shaded zone in *D* also become shaded.

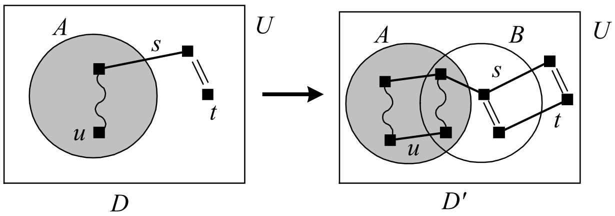


Figure 3.8

Rule 7: Equivalence of Venn and Euler forms. We may replace a diagram *D* in which some regions do not exist by a diagram *V*(*D*) in Venn form where those regions are shaded. All other diagrammatic elements—other shaded regions, spiders, strands and ties—remain unchanged.

Conversely, we may replace a diagram *D* in Venn form which has a set of shaded zones containing no spider by a diagram *E* where (some of) those regions do not exist. Again, all other diagrammatic elements—other shaded regions, spiders, strands and ties—remain unchanged.

The transition from the Euler to the Venn form of a spider diagram is algorithmic. There are various known algorithms for constructing a Venn diagram with *n* contours—for example, see [6]. Given a spider diagram *D* in Euler form, first construct the underlying Venn diagram whose set of contours is *C*(*D*). Shade any zones which were not present in the original Euler form *D*. Finally add spiders, strands and ties in order to replicate the strand-tie graph in each zone of *D*. The resulting spider diagram is *V*(*D*), the Venn form of *D*.

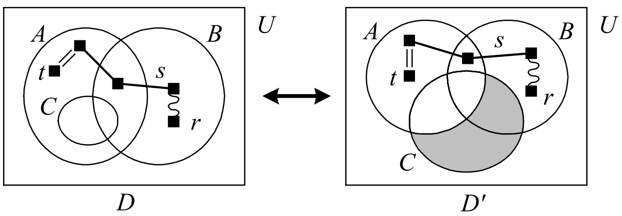


Figure 3.9

Example 3.8 Figure 3.9 illustrates the equivalence between the Euler and Venn forms of a spider diagram. The Euler form D does not contain zones corresponding to $\overline{A} \cap \overline{B} \cap C$ or $\overline{A} \cap B \cap C$. In the Venn form D', the corresponding regions are shaded, but the strand-tie graph in every other zone is the same as the corresponding graph in D.

3.2. Comparing regions

Later we will need to be able to identify 'corresponding' regions in different diagrams. For simplicity, we consider the case where a diagram D' is obtained from a diagram D by adding contours, so that

$$C(D) \subseteq C(D').$$

There is a natural mapping

$$\alpha : Z(D) \to R(D')$$

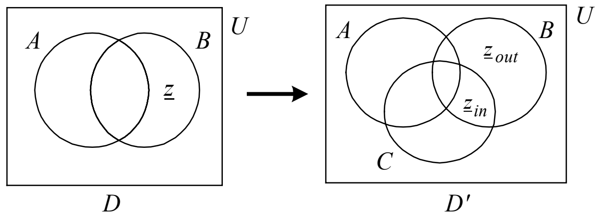
which identifies zones in D with their corresponding regions in D'. The mapping may be defined inductively, with the inductive step as follows. Suppose that D' is obtained from D by adding a single contour. According to Rule 6, each zone z in D bifurcates into two zones z_{in} and z_{out} in D'; z_{in} is that part of z enclosed within the new contour and z_{out} is that part of z lying outside the new contour (see figure 3.10). In this case, we define

$$\alpha(z) = z_{in} \cup z_{out}.$$

Given any zone z' in D', there is a unique zone z in D such that $z' \subseteq \alpha(z)$. The association $z' \mapsto z$ defines a mapping

$$\beta : Z(D') \to Z(D)$$

so that $\beta(z')$ is the unique zone in D that denotes a superset of the set represented by z'. The mappings α and β are illustrated in figure 3.10.



$\alpha(z) = z_{in} \cup z_{out},\ \beta(z_{in}) = z = \beta(z_{out})$

Figure 3.10

By taking unions of zones, these mappings extend to mappings

$$\alpha : R(D) \rightarrow R(D'), \quad \beta : R(D') \rightarrow R(D).$$

These mappings are related as follows. For all regions $r \in R(D)$, $\beta\alpha(r) = r$ and for all regions $r' \in R(D')$, $r' \subseteq \alpha\beta(r')$. The first of these statements says that β is a left inverse for α and α is a right inverse for β. It follows that α is injective and β is surjective.

We say that a region $r \in R(D)$ *corresponds* to a region $r' \in R(D')$ if $\alpha(r) = r'$. We will need the following lemma.

Lemma 3.1

(i) Let D' be the diagram formed from D by adding a contour C' satisfying the partial overlapping rule. If the zones z_{in} and z_{out} in D' are formed from the zone z in D as described above, then $\Psi(z, D) = \Psi(z_{in}, D') \cup \Psi(z_{out}, D')$.

(ii) Let D' be the diagram formed from D by adding a contours satisfying the partial overlapping rule. If a region $r \in R(D)$ *corresponds* to a region $r' \in R(D')$ then they denote the same set: $\Psi(r) = \Psi(r')$.

3.3. Combining diagrams

Given two diagrams, D_1 and D_2, we wish to combine them to produce a single diagram D which retains as much of their combined semantic information as possible. Of course, this is only meaningful if the pair D_1, D_2 is consistent. In this section we describe the construction of such a combined diagram D. Even in simple cases, some information contained in the pair D_1, D_2 will be lost in the combination. In the next section, we will indicate one possible way of enriching the system of spider diagrams to overcome this problem.

Suppose two diagrams D_1 and D_2 are given which do not contain conflicting information. To simplify the process of combination, we first construct the equivalent Venn form of each diagram, $V(D_1)$ and $V(D_2)$ respectively. The combined diagram clearly must contain any contour which appears in either D_1 or D_2, so the first step in combining the diagrams is to construct a Venn diagram whose set of contours is

$$C(D_1) \cup C(D_2).$$

From this underlying Venn diagram, we add diagrammatic elements—shading, spiders, strands and ties—to produce the final combined diagram D. Since D is obtained from each of the diagrams $V(D_1)$ and $V(D_2)$ by adding contours, the 'corresponding region' mappings introduced in the previous section are defined between $V(D_1)$ and D and between $V(D_2)$ and D. These are denoted, respectively, α_1, β_1 and α_2, β_2.

Any shaded zone in the Venn forms $V(D_1)$ or $V(D_2)$ must correspond to a shaded region in D. Hence a zone z of D is shaded if and only if $\beta_1(z) \in Z^*(V(D_1))$ or $\beta_2(z) \in Z^*(V(D_2))$. As a consequence, we have:

$$\bigcup_{z \in Z^*(D)} z = \bigcup_{z \in Z^*(V(D_1))} \alpha_1(z) \cup \bigcup_{z \in Z^*(V(D_2))} \alpha_2(z) .$$

This step is illustrated in figure 3.11 (where $D_1 = V(D_1)$ and $D_2 = V(D_2)$).

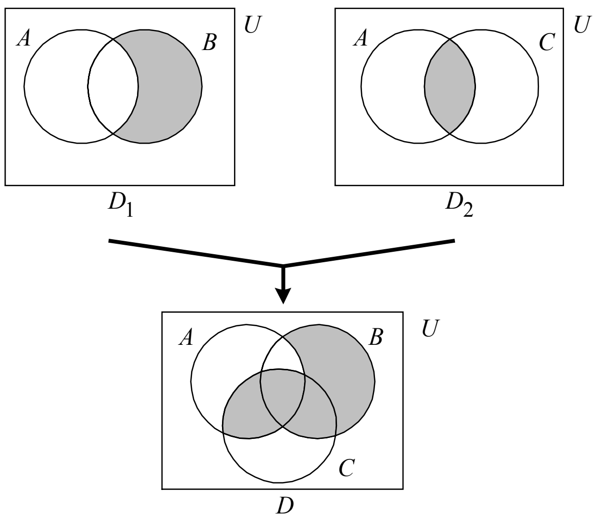


Figure 3.11

Next, we add spiders to D. Since $\eta(s)$ defines the region to which s belongs, intuition suggests that, for each spider, its habitat in D should be the intersection of the corresponding habitats in $V(D_1)$ and $V(D_2)$. This is not quite correct, however, since it does not take account of regions which are known to be empty.

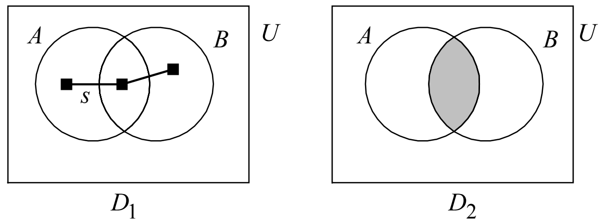


Figure 3.12

This is illustrated in figure 3.12. The habitat of the spider s in the combined diagram must exclude the region $A \cap B$ since, from D_2, this corresponds to an empty set. We define a region of a spider diagram D to be *empty* if it is shaded and is not touched by any spider's foot. We denote by $E(V(D))$ the set of the empty zones of $V(D)$:

$$E(V(D)) = Z^*(V(D)) \cap \{z \in Z(V(D)) \mid T(z) = \varnothing\}.$$

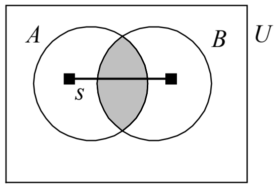
For each spider $s \in S(V(D_1)) \cup S(V(D_2))$, we need to define its habitat in D. There are essentially two cases. If s belongs to both diagrams D_1 and D_2 then its habitat in D is the intersection of its habitats in each diagram:

$$s \in S(V(D_1)) \cap S(V(D_2)) \;\Rightarrow\; \eta(s, D) = \alpha_1(\eta(s, (V(D_1))) \cap \alpha_2(\eta(s, V(D_2)))$$

If s belongs to exactly one of the diagrams D_1 and D_2 then its habitat in D is reduced by removing from it the empty zones in the other diagram:

$$s \in S(V(D_1)) - S(V(D_2)) \;\Rightarrow\; \eta(s, D) = \alpha_1(\eta(s, (V(D_1))) - \bigcup_{z \in E(V(D_2))} \alpha_2(z)\,.$$

With these definitions, the composition of the two diagrams in figure 3.12 is given in figure 3.13.



D

Figure 3.13

Finally, we consider strands and ties. Suppose two spiders are such that each has a foot in a zone z of the combined diagram D. Then z corresponds to zones $z_1 = \beta_1(z)$ and $z_2 = \beta_2(z)$ in $V(D_1)$ and $V(D_2)$, respectively. Again there are several cases to consider.

- If neither diagram $V(D_1)$ nor $V(D_2)$ contains both spiders, then they should be joined by a strand in z. In this case, one spider belongs to $V(D_1)$ and the other belongs to $V(D_2)$, so we have no information concerning their equality or otherwise if they belong to z; hence the spiders should be connected in the most general way.
- If exactly one of the diagrams, $V(D_i)$ say, contains both spiders, then they should be connected in z in the same manner as in z_i.
- If both diagrams contain both spiders then:
  - they are connected by a tie in z if they are joined by a tie in one of the regions z_1, z_2 and a tie or strand in the other region;
  - they are not connected in z if they are not connected in one of the regions z_1, z_2 and are either not connected or connected by a strand in the other region;
  - otherwise they are connected by a strand in z.

Example 3.9. Consider the diagrams given in figure 3.14. Since $C(D_1) = C(D_2)$, it follows that each of the correspondence mappings α_1, β_1, α_2, β_2 defined above is the identity mapping. Since there are also no shaded regions, it follows that

$$\eta(t, D) = \eta(t, D_1) \cap \eta(t, D_2).$$

The habitat of the spider s is equal in all three diagrams.

We need to consider separately each zone in D which contains feet of both spiders. For $B - A$, the spiders are connected by a tie in one diagram (D_2) and a strand in the other; hence in the combined diagram, they are connected by a tie. For $A \cap B$, the spiders are separated in one diagram (D_1) and joined by a strand in the other; hence the spiders should be separated in D.

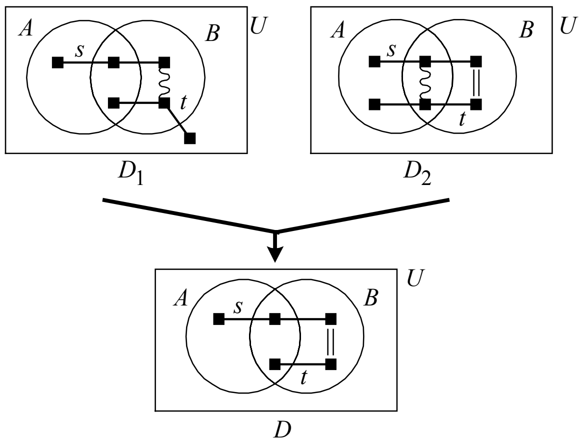


Figure 3.14

Example 3.10. This example illustrates that it is possible for two spiders, s and t, to be separated in z_1 and be joined by a tie in z_2. As before, z_1 and z_2 denote zones in D_1 and D_2, respectively, which correspond to the zone z in the combined diagram D containing feet of both s and t.

Consider the zone $z = A \cap B \cap C$ in the composite diagram D shown in figure 3.15 below. This zone contains feet both of s and of t. The zone z together with the corresponding zones $z_1 = \beta_1(z)$ and $z_2 = \beta_2(z)$ are illustrated with thickened borders. Note that s and t are separated in z_1 but are tied in z_2.

Although it is not possible for the elements corresponding to s and t *both* to belong to $A \cap B \cap C$, this information is not captured in D. Thus it could be argued that it is immaterial how s and t are connected in z. We have chosen to connect their feet with a strand so that each pair of diagrams, D_1, D and D_2, D, is consistent.

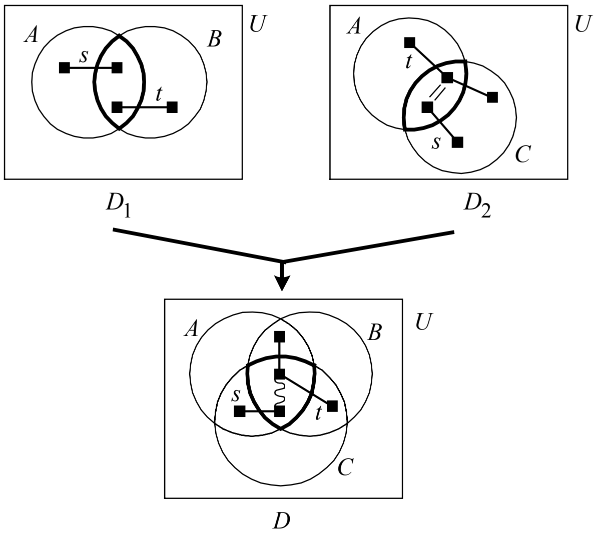


Figure 3.15

4. Soundness and completeness

In this section, we outline the proofs of validity of the inference rules introduced in the previous section and discuss soundness and completeness of this system and related systems. A *model* for a diagram D is a triple $m = (\mathbf{U}, \Psi, \psi)$ where $\Psi: C \to Set\ \mathbf{U}$ and $\psi: S \to \mathbf{U}$ are the semantic functions defined in section 2.2. We say a model m *complies* with D, denoted $m \vDash D$, if it satisfies the conjunction of the semantic predicates introduced in section 2.2. A diagram D' is a *consequence* of a diagram D, denoted $D \vDash D'$, if every compliant model for D is also a compliant model for D'. We assume that contours with the same labels in D and D' represent the same set. To say that a rule is *valid*, we mean that whenever a diagram D' is obtained from another diagram D by a single application of the rule, then $D \vDash D'$.

4.1. Validity of the inference rules.

Several of the rules amount to 'throwing away' some of the semantic information contained in a diagram, in the sense described in the following lemma. Note that we adopt the convention that the conjunction of an empty set of propositions equates to true.

Lemma 4.1 If diagrams D and D' have semantics of the form $\bigwedge_{i\in I} P_i$ and $\bigwedge_{i\in J} P_i$ respectively, where $J \subseteq I$, then D' is a valid inference from D.

Rule 1: Introducing a strand.

Suppose two spiders s and t have feet which are separated (that is, not joined by a strand or a tie) in a zone z belonging to diagram D. Let D' be the diagram obtained from D by adding a strand between the feet of s and t in z. Then

$$\zeta(s, t, D') = \zeta(s, t, D) \cup z.$$

The Strangers Condition is the only semantic condition which involves the web of s and t; for these spiders the condition is

$$\psi(s) = \psi(t) \;\Rightarrow\; \psi(s), \psi(t) \in \Psi(\zeta(s, t, D)).$$

Since $\zeta(s, t, D) \subseteq \zeta(s, t, D')$, we can infer the corresponding condition for D'. All the other semantic conditions are identical for D and D', so the first part of rule 1 is valid.

To justify the validity of the second part of the rule, suppose D and D' are as described above except that, in D, the spiders s and t are joined by a tie in z. In this case, the web of s and t is unchanged, but their nest changes between the diagrams:

$$\tau(s, t, D') = \tau(s, t, D) - z.$$

Thus it is only the Mating Condition which changes in D'. For s and t, the Mating Condition is a conjunction of terms of the form

$$\big(\psi(s) \in \Psi(z) \wedge \psi(t) \in \Psi(z)\big) \;\Rightarrow\; \psi(s) = \psi(t),$$

one term for each zone z in the nest of s and t. By lemma 3.1, we may infer the Mating Condition of D' from that of D.

Rule 2: Spreading the feet of a spider.

Suppose D' is obtained from D by spreading the feet of spider s into the zone z. We consider the semantic conditions that are changed in passing from D to D'.

Spider Condition. Spreading the feet of s extends its habitat so that

$$\eta(s, D') = \eta(s, D) \cup z.$$

Since $\eta(s, D) \subseteq \eta(s, D')$, the spider condition for D' follows from that for D.

To complete the proof, we suppose z contains a spider t and consider the three cases given in rule 2.

(a) If s and t are joined by a strand in z then $\zeta(s, t, D') = \zeta(s, t, D) \cup z$ so $\zeta(s, t, D) \subseteq \zeta(s, t, D')$. Hence the Strangers Condition for D' follows from that for D. In this case the Mating Condition is unchanged.

(b) If s and t remain separated in m then their web and nest are unchanged and hence so are the Strangers and Mating Conditions.

(c) If s and t are joined by a tie in z then $\zeta(s, t, D') = \zeta(s, t, D) \cup z$ and $\tau(s, t, D') = \tau(s, t, D) \cup z$. The Strangers Condition for D' follows as in case (a). To obtain the Mating Condition for D', we add the conjunct

$$\psi(s),\ \psi(t) \in \Psi(z) \ \Rightarrow\ \psi(s) = \psi(t)$$

to the Mating Condition for D. However, from the Spider Condition for D, we know $\psi(s) \notin \Psi(z)$ since z does not form part of the habitat of s in D. Therefore the additional conjunct is true and the Mating Condition for D' follows.

Rule 3: Erasure of shading.

Erasing the shading in a zone only changes the Shading Condition by removing conjuncts, so the validity of the first part of rule 2 follows by lemma 4.1.

Rule 4: Erasure of a spider.

The validity of the rule for erasing a spider follows similarly. However, in passing from the semantics of D to that of D', one or more conjuncts may be lost from the Spider, Strangers and Mating conditions.

Rule 5: Erasure of a contour.

When we erase a contour C, there exist two possibilities:

- zone z' in D' is corresponds to a zone z in D (e.g., zones z_3 and z_3' in figure 4.1).
- there exist zones z_{in} and z_{out} in D that become a single zone z' in D' (e.g., zones z_1, z_2 and z' in figure 4.1).

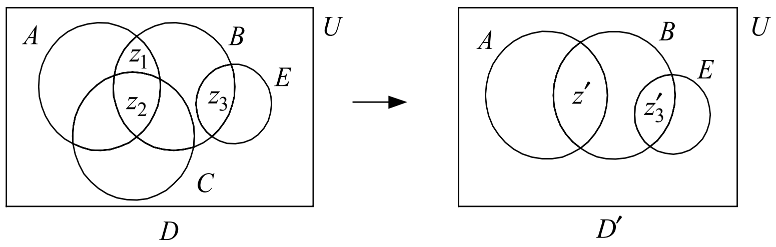


Figure 4.1

When a contour is erased all the semantic conditions are changed. We will consider each condition separately.

The Plane Tiling Condition. Since $C(D') \cup \{C\} = C(D)$, every region in D' has a corresponding region in D and therefore the Plane Tiling Condition for D' follows from the Plane Tiling Condition for D.

The Shading Condition. Erasing any shading remaining in only a part of a zone (z_4 for example in figure 4.2) only changes the Shading Condition by removing conjuncts. Therefore the validity of the first part of the rule follows by lemma 4.1.

For any pair of shaded zones z_{in}, z_{out} in D that combine to form a single shaded zone z' in D' after the erasure of contour C, a pair of conjuncts in the Shading Condition for D becomes a conjunct in the Shading Condition for D' (for example, in figure 4.2, z_1 and z_2 in D combine to form z' in D').

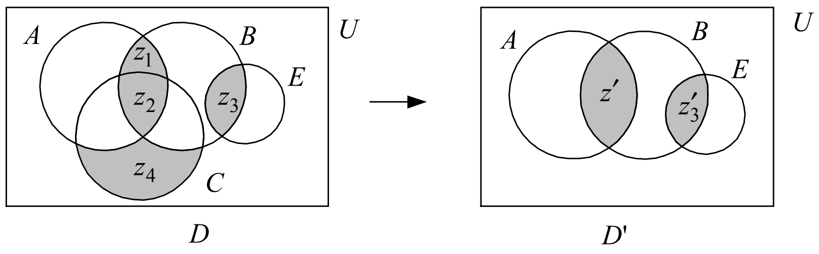


Figure 4.2

By the Shading Condition for *D* we have

$$\Psi(z_{in}) \subseteq \bigcup_{s \in S} \{\psi(s)\} \quad \text{and} \quad \Psi(z_{out}) \subseteq \bigcup_{s \in S} \{\psi(s)\}$$

whence

$$(\Psi(z_{in}) \cup \Psi(z_{out})) \subseteq \bigcup_{s \in S} \{\psi(s)\}$$

which is equivalent, by lemma 3.1, to

$$\Psi(z') \subseteq \bigcup_{s \in S} \{\psi(s)\}$$

the Shading Condition for z' in D'.

The third and final possibility is that a shaded zone z in D (e.g., z_3 in figure 4.2) corresponds to a unique zone z' in D'. Its conjunct in the Shading Condition for D has an equivalent for the shaded zone z' in the Shading Condition for D'.

Spider Condition. For any spider s, the set denoted by its habitat in D is a subset of the set denoted by its habitat in D':

$$\eta(\text{s}, D) \subseteq \eta(s, D')$$

Therefore, we may infer the Spider Condition for D' from that of D.

Mating Condition. For any tie replaced by a strand connecting spiders r and s in a zone z the Mating Condition changes by removing the conjunct

$$\psi(r), \psi(s) \in \Psi(z) \;\Rightarrow\; \psi(r) = \psi(s) \,.$$

For any other tie connecting spiders r and s in zone z and not having to be replaced by a strand, there exists no region k containing feet of r or s such that $\Psi(z) \cup \Psi(k) = \Psi'(z')$ for some z' in D' (otherwise, the tie should be replaced by a strand) or there exists a corresponding region n' in D'. Therefore, their nest changes between the diagrams:

$$\tau(s, t, D) \subseteq \tau(s, t, D')$$

and for any conjunct including spiders r and s in the Mating Condition for D, there is a corresponding conjunct in the Mating Condition for D'.

Strangers Condition. By lemma 3.1, it follows that $\zeta(r, s, D) \subseteq \zeta(r, s, D')$

Therefore for any two spiders r and s in a zone z in D, $\psi(r) = \psi(s) \Rightarrow \psi(r), \psi(s) \in \Psi(\zeta(s,t))$

The Strangers Condition in D' follows: $\psi(r) = \psi(s) \Rightarrow \psi(r), \psi(s) \in \Psi'(\zeta(s,t))$.

Rule 6: Introduction of a contour.

To identify and keep track of the zones in D' that arise from 'splitting' zones in D with the introduction of a contour C'. We denote by z_{in} and z_{out} the two zones in D' which are formed by splitting a zone z in D; z_{in} is that part of z enclosed within the new contour C' and z_{out} is that part of z lying outside C' (see figure 4.3).

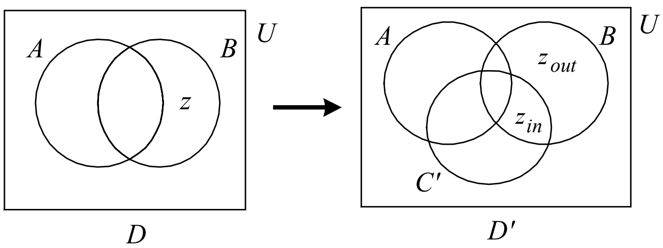


Figure 4.3

In passing from D to D' by adding contour C' as described in the rule*,* a number of semantic conditions change. We consider each condition in turn.

Plane Tiling Condition. The zones in D' may be grouped in pairs of the form z_{in}, z_{out} for some zone z in D. Hence the Plane Tiling Condition for D' follows from the corresponding condition for D by lemma 3.1.

Spider Condition. Suppose a spider s has a foot in the zone z of D. In D' the foot bifurcates giving a foot in each of the zones z_{in} and z_{out} in D'. Hence, by lemma 3.1,

$$\Psi(\eta(s, D)) = \Psi(\eta(s, D')),$$

so the Spider condition for D' follows from that of D.

Mating Condition. Each tie connecting spiders s and t bifurcates in D', so lemma 3.1 ensures that the nest of s and t is unchanged:

$$\tau(s, t, D) = \tau(s, t, D').$$

Suppose that spiders s and t are joined by a tie in a zone z in D. The Mating Condition for z in D gives

$$\psi(s), \psi(t) \in \Psi(z) \Rightarrow \psi(s) = \psi(t).$$

If $\psi(s)$, $\psi(t) \in \Psi'(z_{in}) \subseteq \Psi(z)$ then $\psi(s) = \psi(t)$; similarly, if $\psi(s)$, $\psi(t) \in \Psi'(z_{out}) \subseteq \Psi(z)$ then $\psi(s) = \psi(t)$. The Mating Condition for D' follows.

Strangers Condition. This is similar to the spiders condition. Bifurcating each tie, strand and node of spiders s and t we ensure that sets denoted by their webs in D and D' are identical

$$\zeta(s, t, D) = \zeta(s, t, D').$$

The Strangers Conditions for s and t in D' then follows from the corresponding condition for D.

Shading Condition. Suppose that z is a zone in D. Then both the zones z_{in} and z_{out} in D' are also shaded. The Shading Condition for z, is the following.

$$\Psi(z) \subseteq \bigcup_{s \in S} \{\psi(s)\}$$

Since $\Psi(z) = \Psi'(z_{in}) \cup \Psi'(z_{out})$, this gives $\Psi(z_{in}) \subseteq \bigcup_{s\in S}\{\psi(s)\}$ and $\Psi(z_{out}) \subseteq \bigcup_{s\in S}\{\psi(s)\}$

These are precisely the Shading Conditions for z_{in} and z_{out} in D'.

Rule 7: Equivalence of Venn and Euler forms.

Given that the transition between the Venn and the Euler form of a spider diagram only affects the representation of empty zones it is only the Plane Tiling Condition and the Shading Condition which change. Suppose a diagram D in Venn form has a set $ZE^*(D)$ of shaded zones that are not contained in the Euler form D'. The Shading Condition for D may be separated into terms whose zones belong to $Z^*(D) - ZE^*(D)$ and terms whose zones belong to $ZE^*(D)$:

$$\bigwedge_{z\in Z^*(D)-ZE^*(D)}\left(\Psi(z) \subseteq \bigcup_{s\in S}\{\psi(s)\}\right) \quad \text{and} \quad \bigwedge_{z\in ZE^*(D)}\left(\Psi(z) \subseteq \bigcup_{s\in S}\{\psi(s)\}\right)$$

Note that, since $Z^*(D) - ZE^*(D) = Z^*(D')$, the first collection of conjuncts,

$$\bigwedge_{z\in Z^*(D)-ZE^*(D)}\left(\Psi(z) \subseteq \bigcup_{s\in S}\{\psi(s)\}\right)$$

is equivalent to the Shading Condition for D'.

It can also be shown that the second collection of conjuncts,

$$\bigwedge_{z\in ZE^*(D)}\left(\Psi(z) \subseteq \bigcup_{s\in S}\{\psi(s)\}\right)$$

is equivalent to the Plane Tiling Condition for D'

$$\bigcup_{z\in Z(D')}\Psi(z) = \mathbf{U}$$

since any empty zone in $ZE^*(D)$ does not exist in D' and therefore it is not included in the Plane Tiling Condition for D. Note that the Plane Tiling Condition for the diagram D in Venn form is true since D contains all possible regions.

Combining diagrams

Suppose two diagrams D_1 and D_2 are given which do not contain conflicting information. Their combined diagram is formed by adding syntactic elements into the Venn diagram whose set of contours is $C(D_1) \cup C(D_2)$. By the rule of equivalence of Venn and Euler forms and the rule of introducing contours, we may assume, without loss of generality, that D_1 and D_2 are in Venn form and have the same sets of contours. Thus $C(D) = C(D_1) = C(D_2)$. In this case, $D_1 = V(D_1)$, $D_2 = V(D_2)$ and the 'corresponding region' mappings, α_1, β_1 and α_2, β_2, are identity mappings. We consider each of the semantic conditions for D in turn.

Plane Tiling Condition. Since D is in Venn form, all possible zones appear in the diagram and the Plane Tiling Condition for D follows.

Spider Condition. Let s be a spider in D. There are two cases to consider.

Suppose that s belongs to both D_1 and D_2. Then its habitat in D is the intersection of its habitats in D_1 and D_2: $\eta(s, D) = \eta(s, D_1) \cap \eta(s, D_2)$. In this case we have:

$$\psi(s) \in \Psi(\eta(s, D_1)) \wedge \psi(s) \in \Psi(\eta(s, D_2)) \qquad \text{(from the Spider Conditions in } D_1 \text{ and } D_2\text{)}$$

$$\Rightarrow \quad \psi(s) \in \Psi(\eta(s, D_1)) \cap \Psi(\eta(s, D_2))$$

$$\Rightarrow \quad \psi(s) \in \Psi(\eta(s, D_1) \cap \eta(s, D_2)) \qquad \text{(since corresponding regions denote the same set)}$$

$$\Rightarrow \quad \psi(s) \in \Psi(\eta(s, D))$$

Suppose that s belongs exactly one of the diagrams; say, s belongs to both D_1 but not D_2. Then its habitat in D is

$$\eta(s, D) = \eta(s, D_1) - \bigcup_{z \in E(D_2)} z \,.$$

Now $\psi(s) \in \Psi(\eta(s, D_1))$ from the Spider Condition in D_1 and $\psi(s) \notin \bigcup_{z \in E(D_2)} \Psi(z)$ since $\Psi(z) = \varnothing$ for any zone $z \in E(D_2)$, by lemma 2.1. Hence $\psi(s) \in \Psi(\eta(s, D))$. Therefore the Spider Condition is satisfied in D.

Strangers Condition. The rule for combining diagrams implies that, for all spiders s, t in D, $\zeta(s, t, D_1) \cap \zeta(s, t, D_2) \subseteq \zeta(s, t, D)$. Hence the Strangers Condition for D follows from the Strangers Conditions for D_1 and D_2.

Mating Condition. Let z be a zone in D which forms part of the nest of spiders s and t; that is, $z \subseteq \tau(s, t, D)$. Then z forms part of the nest of s and t in at least one of the diagrams D_1 and D_2. Therefore the Mating Condition for D follows from the Mating Conditions for D_1 and D_2.

Shading Condition. Let z be a shaded zone of D. Then z is shaded in at least one of the diagrams D_1 and D_2. Suppose the spider s has a foot in z in the combined diagram D; that is, $z \subseteq \eta(s, D)$. Then in at least one of the diagrams D_1 and D_2, s has a foot in z and z is shaded; that is, $z \subseteq \eta(s, D_i) \cap \{z^* : z^* \in Z^*(D_i)\}$ for $i = 1$ or 2. Hence if $x \in \Psi(z)$ then $x = \psi(s)$ for some $s \in S(D_i)$ for $i = 1$ or 2, by the Shading Condition for D_i. Since $S(D) = S(D_1) \cup S(D_2)$, the Shading Condition for D follows from the Shading Conditions for D_1 and D_2.

Hence the rule of combining diagrams is valid.

4.2. Soundness

We write $D \vdash D'$ to denote that the diagram D' can be obtained from the diagram D by applying a finite sequence of transformations. Similarly, we write $\{D_1, D_2, \ldots, D_n\} \vdash D'$ if D' can be obtained from the set of diagrams $\{D_1, D_2, \ldots, D_n\}$ by applying a finite sequence of transformations, including the rule for (pairwise) combination of diagrams.

The semantics of a set of diagrams is the conjunction of the semantics of the individual diagrams; the boundary rectangles of all diagrams are interpreted as the same set **U** and contours with the same labels in different individual diagrams are interpreted as the same set.

The following soundness theorem for the spider diagram system follows by induction from the validity of each of the transformation rules and the rule for combining diagrams, established in the previous section.

Theorem 4.1 (Soundness Theorem) If $\{D_1, D_2, \ldots, D_n\} \vdash D'$ then $\{D_1, D_2, \ldots, D_n\} \vDash D'$.

The system of spider diagrams introduced here is not complete, as the following example shows.

Example 4.1. In figure 4.4, diagram D' can be inferred from diagram D, $D \vDash D'$. However, when removing spider s from D, rule 2 would require a strand between spiders r and t in the resulting diagram, a weaker result. The rules of inference do not allow D' to be obtained syntactically from D.

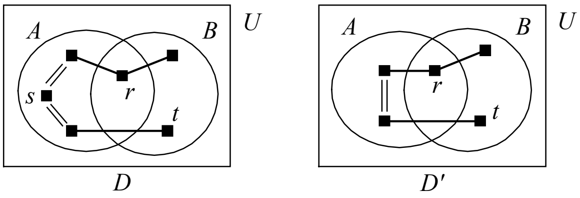


Figure 4.4

4.3. Relationship to other systems.

In order to obtain a system that is complete, we would require a rule for combining diagrams in which no semantic information is lost. In this section we describe one possible solution. We could give figure 4.5 as the combined diagram for the diagrams D_1 and D_2 in example 3.10 (see figure 3.15).

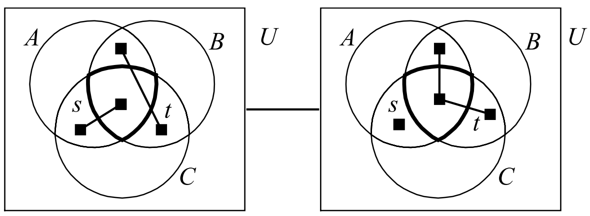


Figure 4.5

Figure 4.5 is a *compound* (or disjunctive) spider diagram. Either the left-hand component holds or the right-hand component holds (see [15] where compound Venn-Peirce diagrams are used). It is not possible for the elements corresponding to s and t *both* to belong to $A \cap B \cap C$. In D, either s can be in $A \cap B \cap C$ and t not, as in the left-hand component, or vice versa, as in the right-hand component. All the semantic information of D_1 and D_2 is captured in disjunctive diagram *D*.

The semantics of a compound diagram is the disjunction of the semantics of its component unitary diagrams; the boundary rectangles of the component unitary diagrams are interpreted as the same set **U**. Contours with the same labels in different component unitary diagrams of a compound diagram *D* are interpreted as the same set. Thus the compound diagram in figure 4.5 asserts that:

$$\left(s \in A \cap C \ \wedge \ t \in (A \cap B \cap \overline{C}) \cup (\overline{A} \cap B \cap C)\right) \ \vee \ \left(s \in A \cap \overline{B} \cap C \ \wedge \ t \in B \cap (A \cup C)\right).$$

We have not considered in detail this system of 'spider diagrams with disjunction'. Instead, we have explored a variety of related systems with slightly different (syntax and) semantics. In [7, 8, 9] we consider diagrams with 'existential spiders' which have feet denoted by small discs (rather than squares) and which represent the existence of an element in the corresponding set. Ties play no part in these systems since $\exists\, x, y \bullet x \in X \wedge y \in X \wedge x = y$ is logically equivalent to $\exists\, x \bullet x \in X$. Figure 4.6 is an example of such a diagram; it asserts that:

$$\left((\exists x, y \bullet x \in A \cap C \wedge y \in B - C) \ \wedge \ (C - (A \cup B) = \emptyset)\right) \ \vee \ \left(\exists x, y \bullet x \in B \wedge y \in A - (B \cup C)\right).$$

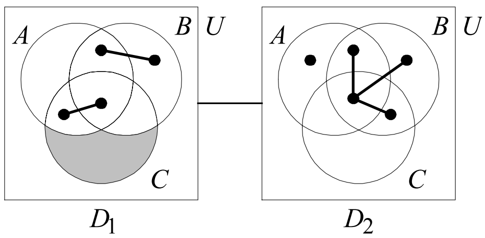


Figure 4.6

The diagram in figure 4.6 is an example of a system of diagrams, called simple spider diagrams, considered in [7]. The system is based on Venn (rather than Euler) diagrams, includes compound (disjunctive) diagrams, does not allow spiders' feet to touch shaded regions and does not contain strands. The transformation rules given in this paper are adapted and extended in [7] and we include a definition of combining diagrams that does not lose semantic information. This system is both sound and complete. The basic strategy to prove completeness (i.e., if $\{D_1, D_2, \ldots, D_n\} \vDash D'$ then $\{D_1, D_2, \ldots, D_n\} \vdash$

D'), is firstly combine the individual diagrams in $\{D_1, D_2, \ldots, D_n\}$ into a single diagram D^*; then expand both D^* and D' into compound diagrams in a way similar to disjunctive normal form in symbolic logic; it then follows that for each unitary component in the expanded form of D^* there exists a component in the expanded form of D' that follows logically from it; finally from the diagrammatic conditions that must hold between these two components, it follows that one can be transformed into the other syntactically. This proof strategy extends, with some modification, to other spider and constraint diagram systems.

In [8] we increase the expressiveness of the system by allowing spiders' feet to touch shaded regions (as we do for the system considered in this paper). The syntactic elements of the system are further extended in [9] where we allow Euler diagrams and we include strands and 'Schrödinger spiders'. The extended system in [9] also allows the negation of a diagram to be represented in a straightforward way. Both these extended systems are shown to be sound and complete [8, 9].

5. Conclusions

In this paper, we have given the main syntax and semantics of spider diagrams. We have given inference rules, a rule governing the equivalence of the Venn and Euler forms of spider diagrams and a rule for combining spider diagrams. These rules have been shown to be sound; but, in some cases, the rules do not give as strong an inference as possible and so the system is not complete.

There are related systems of spider diagrams that have existential spiders and include compound diagrams. These systems are known to be both sound and complete. The general aim of this work is to provide the necessary mathematical underpinning for the development of software tools to aid reasoning with diagrams. In particular, we aim to prove similar results for constraint diagrams and to develop the tools that will enable them to become part of the software development standard.

Acknowledgements

We gratefully acknowledge Richard Bosworth, Paul Courtney, Richard Mitchell, Dan Simpson and Kees van Deempter for comments on earlier versions of this paper. Authors Howse and Kent acknowledge support of the UK EPSRC grant number GR/M02606.

References

1. G. Allwein & J. Barwise (editors) (1996) Logical Reasoning with Diagrams. Oxford University Press, New York.

2. L. Euler (1772) Lettres a Une Princesse d'Allemagne. Vol. 2, Sur divers subject de physique et de philosophie, Letters No. 102-108. (Reprint of 1795 edition: (1997) Thoemmes Press, Bristol)

3. Y Gil, J Howse & S Kent (1999) Formalizing Spider Diagrams. In: Proceedings of 1999 IEEE Symposium on Visual Languages (VL99), IEEE Computer Society Press, Los Alamitos, pp 130-137.

4. Gil, Y., Howse, J., Kent, S. (1999) Constraint Diagrams: a step beyond UML. In: Proceedings of Technology of Object-Oriented Languages and Systems (TOOLS 30), IEEE Computer Society Press, Los Alamitos, pp 454-463.

5. Glasgow, J, Narayanan, N, Chandrasekaran, B (editors) (1995) Diagrammatic Reasoning, AAAI Press, Menlo Park Ca.

6. Hammer, E.M. (1995) Logic and Visual Information. CSLI Publications, Stanford.

7. J. Howse, F. Molina, & J. Taylor (2000) A Sound and Complete Diagrammatic Reasoning System. In: Proceedings of 3rd IASTED International Conference on Artificial Intelligence and Soft Computing (ASC 2000), (to appear).

8. J. Howse, F. Molina, & J. Taylor (2000) SD2: A Sound and Complete Diagrammatic Reasoning System. In: Proceedings of 2000 IEEE Symposium on Visual Languages (VL2000), IEEE Computer Society Press, Los Alamitos, (to appear).

9. J. Howse, F. Molina, & J. Taylor (2000) On the Completeness and Expressiveness of Spider Diagram Systems. In: Proceedings of Diagrams 2000, lecture Notes in Artificial Intelligence, Springer, (to appear).

10. S. Kent (1997) Constraint Diagrams: Visualising Invariants in Object Oriented Models. In: Proceedings of 1997 ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages and Applications (OOPSLA 97), ACM Press, pp 347-341.

11. R. Lull (1517) Ars Magma. Lyons.

12. T. More (1959) On the construction of Venn diagrams. Journal of Symbolic Logic, 24, 303-304.

13. C. Peirce (1933) Collected Papers. Vol. 4. Ed. C Hartshorne & P Weiss, Harvard University Press, Cambridge Ma.

14. J. Rumbaugh, I. Jacobson & G. Booch (1999) Unified Modeling Language Reference Manual. Addison-Wesley, Reading Ma.

15. S-J. Shin (1994) The Logical Status of Diagrams. Cambridge University Press, Cambridge.

16. J. Venn (1880) On the Diagrammatic and Mechanical Representation of Propositions and Reasonings. Phil. Mag.(5) 9 , 1-18.

17. Warmer, J. and Kleppe, A. (1998) The Object Constraint Language: Precise Modeling with UML. Addison-Wesley, Reading Ma.