Projections in Venn-Euler Diagrams

Joseph (Yossi) Gil
Department of Computing Science
Technion–IIT, Haifa 32000, Israel
yogi@cs.technion.ac.il

John Howse
School of Computing and Mathematical Sciences
University of Brighton, UK
John.Howse@brighton.ac.uk

Stuart Kent
Computing Laboratory
University of Kent, Canterbury, UK
S.J.H.Kent@ukc.ac.uk

John Taylor
School of Computing and Mathematical Sciences
University of Brighton, UK
John.Taylor@brighton.ac.uk

Abstract

Venn diagrams and Euler circles have long been used to express constraints on sets and their relationships with other sets. However, these notations can get very cluttered when we consider many closed curves or contours. In order to reduce this clutter, and to focus attention within the diagram appropriately, the notion of a *projected* contour, or *projection*, is introduced. Informally, a projected contour is a contour describes a set of elements limited to a certain *context*. Through a series of examples, we develop a formal semantics of projections and discuss the visual language design issues involved in introducing these.

Keywords Visual formalisms, diagrammatic notations

1. Introduction

Diagrammatic notations involving circles and other closed curves, which we will call contours, have been in use for the representation of classical syllogisms since at least the Middle Ages [10]. In the middle of the 18th century, the Swiss mathematician Leonhard Euler introduced the notation we now call Euler circles (or Euler diagrams) [1] to illustrate relations between sets. This notation uses the topological properties of *enclosure*, *exclusion* and *intersection* to represent the set-theoretic notions of *containment*, *disjointness*, and *intersection*, respectively. The 19th century logician John Venn [15] 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 single 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.

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. However, as we will see next, both notations have their limitations.

Venn diagrams are expressive as a visual notation for writing constraints on sets and their relationships with other sets, but difficult to draw because all possible intersections have to be drawn and then some regions shaded. Although it may first seem impossible to draw a Venn diagram of more than three contours, there are in fact many ways of doing so. Venn himself developed a scheme for drawing such a diagram for any number of diagrams. Yet another such scheme is due to More [11]. Since then, there was a large body of research on the drawing of Venn diagrams, their topological properties, etc. The interested reader is referred to e.g., [4,5] for more information on the topic, which involves some beautiful mathematics, which results in some very aesthetically pleasing drawing. For example, Figure 1 shows a symmetrical Venn diagram of four contours, while Figure 2 is the only simple symmetric Venn diagram of five contours.

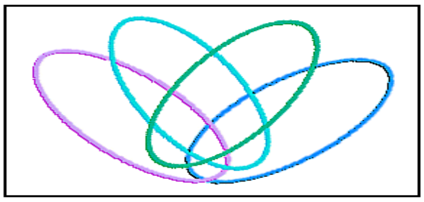

Figure 1 - A simple and symmetrical Venn diagram of four contours

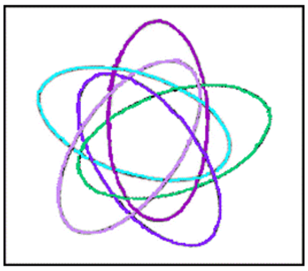

Figure 2 – The simple symmetrical Venn diagram of five contours

Examining these two figures, it is clear why it is so rare to see Venn diagrams of four or more contours used in visual formalism. Most regions require a bit of pondering before it is clear which are the contours that contain it. As shown in Figure 3, the situation worsens with the increase in the number of curves.

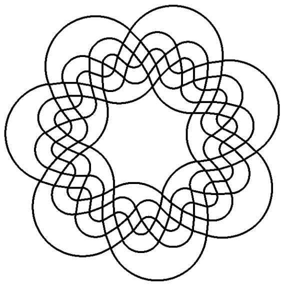

Figure 3 - Adelaide, a symmetrical Venn diagram of seven contours

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. It is therefore the case that an informal hybrid of the two notations that is used for teaching purposes. We use the term *Venn-Euler diagrams* for the notation obtained by a relaxation of the demand that all curves in Venn-diagrams must intersect or conversely, by introducing shading into Euler diagrams. Gil, Howse and Kent [2] provided formalism for Venn-Euler diagrams as part of the more general *spider diagrams* notation.

Rather informally, we use the following terminology: A *contour* is a simple closed plane curve. A *boundary contour* is not contained in and does not intersect with any other contour. A *district* (or *basic region*) is the set of points in 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 (defined as sets of points of the plane) 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.

Every region is a union of zones. A region is *shaded* if each of its component zones is shaded. A shaded region denotes the empty set.

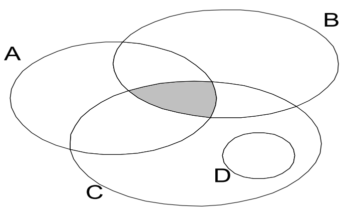

Figure 4 - A Venn-Euler diagram

The Venn-Euler diagram D in Figure 4 has four non-boundary contours A, B, C, D and the boundary is omitted. Its interpretation includes $D \subseteq (C - B) - A$ and $A \cap B \cap C = \varnothing$.

However, even Venn-Euler diagrams can get very cluttered when many contours are involved. The issue of clutter becomes even more crucial when such diagrams are used as foundation for other, more advanced visual formalism. A case in point is the *constraint diagrams* notation [3] which uses arrows and other diagrammatic elements to model constraints not only on simple sets, but also on mathematical *relations*. Constraint diagrams can be used in conjunction with the Unified Modeling Language (UML) [13], which has become the Object Management Group's (OMG) standard for object-oriented modelling notations, and the Object Constraint Language (OCL) [16], a textual notation for expressing constraints that is part of UML.

In order to reduce this clutter, and to focus attention within the diagram appropriately, the notion of a *projected* contour, or *projection*, can be used as an addition to the Venn-Euler based notation. In Figure 5 for example, the set Women is projected into the set of employees. The projected contour represents the set of women employees; it doesn't say that all women are employees.

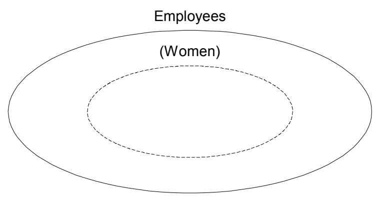

Figure 5 - Example projection

As a slightly more interesting example, consider the constraint diagram in Figure 6. This diagram states (among other things) that the sets Kings and Queens are disjoint, that the set Kings has an element named Henry VIII, that all women that Henry VIII married were queens and that there was at least one woman he married who was executed. The dotted contour is a projection of the set Executed; it is the set of all executed people projected into the set of people married to Henry VIII, that is, it gives all the queens who were married to Henry VIII and executed.

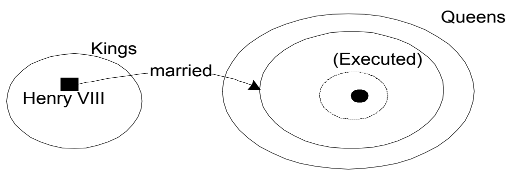

Figure 6 - A constraint diagram with projection

Thus, projection, denoted by a dotted contour can be thought of as a notation for intersection. In the example, the inner most circle labelled "(Executed)" denotes the intersection of the set Executed with the set of women who were married to Henry VIII. The notation is intuitive and more concise than the alternative, which would have been drawing a large ellipse that intersects the Queens contour. As shown in Figure 7, this ellipse must also intersect with the Kings contour, or otherwise the diagram would imply that no kings were ever executed.

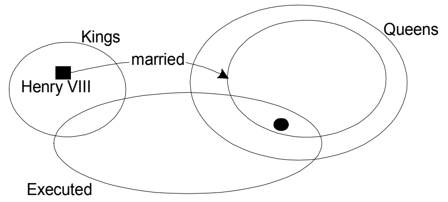

Figure 7 – The constraint diagram of Figure 6 redrawn without projections.

Moreover, Figure 6 does not specify whether Henry VIII was executed or not. The elimination of projections then requires delving into a history book and explicitly specifying this point as shown in Figure 7. Alternatively, one could use what is known as a spider to refrain from stating whether or not Henry VIII was executed. As shown in Figure 8, this alternative is even more cumbersome.

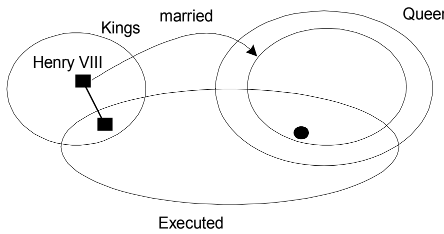

Figure 8 – Using a spider notation to preserve the semantics of Figure 6 while eliminating projections from it.

There are non-trivial issues in dealing with this seemingly neat idea. For example, the notation must have a well-defined semantics when a projection intersects with a contour, and not only when it is disjoint to it, or contained in it. Moreover, a diagram may contain more than one projection that may interact in subtle ways.

The projection concept was first suggested as part of the constraint diagram language. However, these complicating matters were not dealt with. Instead, there was a tacit understanding that only “simple” use of projections, which avoided these issues was allowed. This work represents the first attempt to systematically deal with the semantics of projections.

The rest of this paper is structured as follows. Section 2 briefly sketches the formal semantics of Venn-Euler diagrams. Section 3 gives an informal definition of projections. The formal semantics is given in Section 4. In Section 5 we consider interacting projections and give a further syntactic constraint to the notation. Section 6 deals with issues involved in the semantics of projections. Finally, Section 7 gives a conclusion and discussed related work.

2. Semantics of Venn-Euler Diagrams

In this section we sketch the main definitions used in giving semantics to an Venn-Euler diagram. A *Venn-Euler diagram* is a finite collection of contours and a list of shaded zones, where each zone is a non-empty subset of the contours. Exactly one of the contours must be denoted as *boundary* contour. (We frequently omit the boundary contour from drawings.) For any diagram D, we use $C = C(D)$, $R = R(D)$, $Z = Z(D)$, and $Z^* = Z^*(D)$ to denote the sets of contours, regions, zones, and shaded zones of D, respectively.

The semantics of a Venn-Euler diagram D is given in terms of the semantic function

$$\Psi : C \to \wp\mathbf{U},$$

where $\mathbf{U}$ is a given universal set of D and $\wp\mathbf{U}$ denotes the power set of $\mathbf{U}$. Contours are interpreted as subsets of $\mathbf{U}$, and 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 $\Psi : Z \to \wp\mathbf{U}$ 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 $\Psi : R \to \wp\mathbf{U}$ 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:

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

Shading Condition: The set denoted by a shaded zone is empty

$$\bigwedge_{z \in Z^*} \Psi(z) = \emptyset$$

3. Projections

Sometimes it is necessary to show a set in a certain context. Intersection can be used for just this purpose: an intersection of *A* and *B* shows the set *A* in the context of *B* and vice-versa. However, intersections also introduce regions that may not be of interest. Projections are equivalent to taking the intersection of sets, except that they introduce fewer regions, with the effect that regions which are not the focus of attention are not shown, resulting in less cluttered diagrams.

A *projection* is a contour, which is used to denote an intersection of a set with a "context". By convention, we use dashed iconic representation to make the distinction between projections and other contours.

A *determining label*, denoted by $\lambda(p)$, must be associated with any projection *p*. This label is used to denote the set which is being projected. The convention is that determining labels are rendered within parenthesis when drawn in a diagram. A projection can also have a contour label.

Definition 1 The *context* of a projection *p*, denoted $\kappa(p)$, is the smallest region, defined in terms of non-projected contours, that contains the district of *p*.

The set denoted by the context of a projection is calculated from the sets denoted by non-projected contours; it is defined independently of projected contours for reasons discussed in Section 7. A projection *p* denotes the set obtained by intersecting the set denoted by its determining label $\lambda(p)$ with the set denoted by its context $\kappa(p)$.

Figure 9 shows a simple example. The dashed contour labelled *X* denotes the set obtained by "projecting" the set *A* onto the context $D - B$, i.e., $X = A \cap (D - B)$.

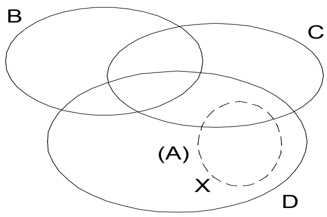

Figure 9 - Simple projection

The same semantics could have been obtained by using More's algorithm [11] to draw the Venn diagram with four contours, as in Figure 10, in which $X = \underline{X1} \cup \underline{X2} = A \cap (D - B)$, where $\underline{X1}$ and $\underline{X2}$ denote the zones in which the labels appear. The simplicity of Figure 3, when compared to that of Figure 4, is self-evident.

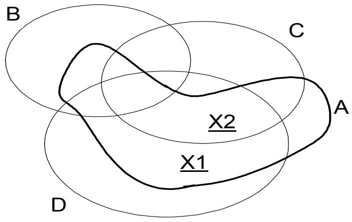

Figure 10 - Semantics of Figure 9

Thus, a projection gives another way of showing the intersection of sets. This gives a clue to its value, given the notorious difficulty of showing the intersection of more than three sets on a Venn diagram: Figure 11 shows how all the regions obtained by intersecting six sets can be obtained using projections. This is an extreme case. More often than not, one is only interested in some of the intersections and not the others: projections provide the freedom to show only those intersections of interest.

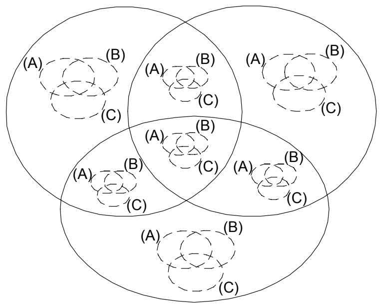

Figure 11 - Six sets

4. Semantics of Simple Projections

Let *P* be the set of all projections and *L* be the set of all determining labels. We extend the domain of the semantic function Ψ to interpret projections and determining labels as subsets of **U**:

$$\Psi : P \to \wp\mathbf{U}, \Psi : L \to \wp\mathbf{U}.$$

Let *p* be a projection with determining label $\lambda(p)$ and context $\kappa(p)$. Then we have:

$$\Psi(p) = \Psi(\lambda(p)) \cap \Psi(\kappa(p)).$$

So, we can add a new semantic condition to the two given earlier. Let *P*(*D*) be the set of all projections in diagram *D*. Then the semantics of a diagram *D* is the conjunction of the Plane Tiling Condition, the Shading Condition and the Projection Condition.

Projection Condition: The set denoted by a projection is the intersection of the set denoted by its determining label and the set denoted by its context:

$$\bigwedge_{p \in P(D)} \Psi(p) = \Psi(\lambda(p)) \cap \Psi(\kappa(p)).$$

5. Interacting Projections

In this section we consider examples of interacting projections and highlight some problems. The solution of these problems requires a syntactic constraint on projections. There are two main cases to consider: disjoint projections and intersecting projections. The case of projections contained in each other is similar to that of intersection projections.

5.1 Disjoint Projections

The intuitive interpretation of the diagram in Figure 12 is that $X = A \cap B$ and $Y = A \cap C$ and that $A \cap B$ and $A \cap C$ are disjoint.

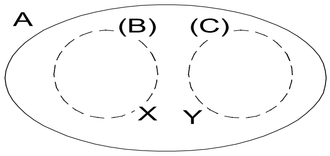

Figure 12 - Disjoint projections

We will interpret $\Psi(A)$ as *A,* etc., for simplicity (and, of course, this will almost always be the intention of the producer of the diagram).

Now, $\kappa(X) = A$ and $\kappa(Y) = A$. The Projection Condition gives $X = A \cap B$ and $Y = A \cap C$ and the Plane Tiling Condition says that *X* and *Y* are disjoint, which is the intuitive interpretation. Note that this specifies that $A \cap B$ and $A \cap C$ are disjoint, so even though we are not explicitly showing the contours *B* and *C*, we can still constrain the sets that they represent.

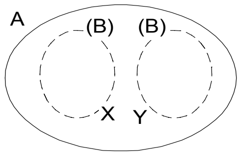

Figure 13- An illegal diagram

Now, consider the diagram in Figure 13. We have $\kappa(X) = A$ and $\kappa(Y) = A$. The Projection Condition gives $X = A \cap B$ and $Y = A \cap B$ and the Plane Tiling Condition says that *X* and *Y* are disjoint. So, we have $A \cap B = \varnothing$. We could have said the same thing by shading a single projection of *B* in *A* as in Figure 14.

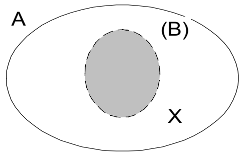

Figure 14 – Empty projection

There are various extensions of Venn-Euler diagrams in which elements of sets can be shown diagrammatically; these include Peirce diagrams [4, 12, 14] and spider diagrams [2]. If a diagram such as that in Figure 8 occurred in such a system, then the diagram could be made inconsistent by placing an element icon in one of the projections. Because of this and other similar problems, this situation is not allowed. We have the following syntactic constraint on projections:

if two projections have the same context, then they must have different determining labels.

Formally: let p_1 and p_2 be projections, then

$$\kappa(p_1) = \kappa(p_2) \Rightarrow \lambda(p_1) \neq \lambda(p_2).$$

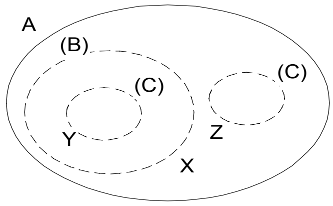

Figure 15 - Another illegal diagram

The diagram in Figure 15 is illegal because *Y* and *Z* have the same context and the same determining label. The diagram in Figure 15 is probably all that is intended.

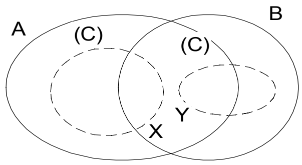

Figure 16 - Yet another illegal diagram

In fact, this syntactic constraint is not strong enough. Consider the diagram in Figure 16. The context of *X* is *A*, the context of *Y* is *B*. The Projection Condition gives $X = A \cap C$ and $Y = B \cap C$ and the Plane Tiling Condition says that *X* and *Y* are disjoint. Hence, we have $A \cap B \cap C = \varnothing$, which, again, is problematic and could lead to inconsistent diagrams in some systems. The complete (syntactic) constraint that prevents these situations is the following:

Projection Label Constraint: *if two projections have the same determining label, then they must have disjoint contexts.*

Formally: let p_1 and p_2 be projections, then

$$\lambda(p_1) = \lambda(p_2) \Rightarrow \kappa(p_1) \cap \kappa(p_2) = \varnothing .$$

Theorem 1 Imposing the Projection Label Constraint does not limit expressiveness.

Proof Suppose that projections p_1 and p_2 do not satisfy the constraint:

$$\lambda(p_1) = \lambda(p_2) = \lambda \quad \text{and} \quad \kappa(p_1) \cap \kappa(p_2) = \kappa \neq \varnothing$$

as illustrated in Figure 17.

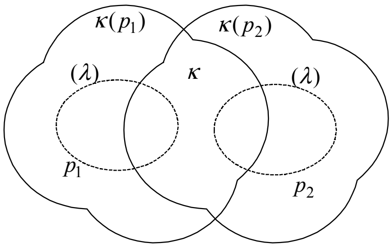

Figure 17 - Intersecting contexts

Suppose that p_1 and p_2 are disjoint (if not, p_1 and p_2 should be replaced by a single projection). Then, the Plane Tiling Condition gives

$$\Psi(p_1) \cap \Psi(p_2) = \varnothing$$

The Projection Condition gives

$$\Psi(p_1) = \Psi(\lambda(p_1)) \cap \Psi(\kappa(p_1))$$
$$\Psi(p_2) = \Psi(\lambda(p_2)) \cap \Psi(\kappa(p_2))$$

Therefore,

$$\Psi(\lambda(p_1)) \cap \Psi(\kappa(p_1)) \cap \Psi(\lambda(p_2)) \cap \Psi(\kappa(p_2)) = \varnothing$$

hence,

$$\Psi(\lambda) \cap \Psi(\kappa(p_1)) \cap \Psi(\kappa(p_2)) = \varnothing$$

i.e.,

$$\Psi(\lambda) \cap \Psi(\kappa) = \varnothing.$$

So, p_1 and p_2 can be replaced by a single projection p whose intersection with κ is shaded. This is expressed by the legal diagram in Figure 13.

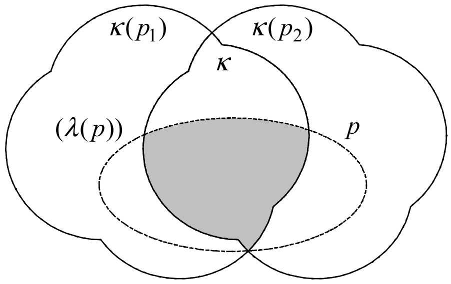

Figure 18 - Legal version of Figure 17

5.2 Containing and Intersecting Projections

Consider the diagram in Figure 14. The intuitive interpretation is that $A \cap C \subseteq A \cap B$, or, in the context of A, C is a subset of B.

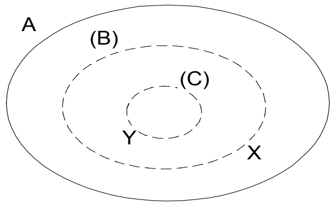

Figure 19 - A containing projection

By the projection condition $X = A \cap B$ and $Y = A \cap C$. By the plane tiling condition $Y \subseteq X$. So, $A \cap C \subseteq A \cap B$, the intuitive interpretation. Note, again, that the sets B and C have been constrained. In fact, we can obtain precise expressions for X and Y: $X = A \cap B$ and $Y = A \cap B \cap C$, because $A \cap \overline{B} \cap C = \varnothing$.

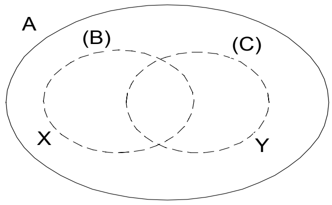

Figure 20 - Intersecting projections

In Figure 20, the two projections intersect in the same context. In this case there are no constraints on the sets B or C (or A).

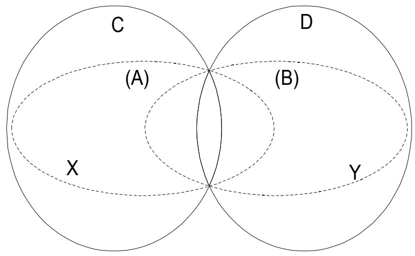

Figure 21 - Interacting projections

In Figure 21 there is a more complicated intersection of projections. The context of both projections *X* and *Y* is $C \cup D$. So $X = (C \cup D) \cap A$ and $Y = (C \cup D) \cap B$. But in this case *X* and *Y* *are* constrained: $X \subseteq C \cup Y$ and $Y \subseteq D \cup X$. Putting all this together, we have

$$X \subseteq C \cup Y = C \cup (C \cup D) \cap B = C \cup D \cap B\,.$$

So,

$$X = (C \cup D) \cap A \cap (C \cup D \cap B) = A \cap (C \cup D \cap B)$$

and further,

$$(C \cup D) \cap A \cap \overline{C \cup D \cap B} = \varnothing$$

hence,

$$A \cap D \cap \overline{C} \cap \overline{B} = \varnothing\,.$$

We can obtain similar expressions for *Y* by symmetry. So we have precise expressions for *X* and *Y*:

$$X = A \cap (C \cup B \cap D) \text{ and } Y = B \cap (D \cup A \cap C)$$

with

$$A \cap D \cap \overline{C} \cap \overline{B} = \varnothing \text{ and } B \cap C \cap \overline{D} \cap \overline{A} = \varnothing\,.$$

A few seconds' thought will confirm that this is an intuitive interpretation.

6. An Alternative Semantics

There is an alternative possibility for the semantics of projections, which is to include projections in the context of a projection. This interpretation of interacting projections involves the solution of simultaneous set equations. In general, these equations have many solutions, but there is usually a "minimal" solution. This minimal solution frequently agrees with the intuitive interpretation. However, there are some cases in which the solution might turn counter-intuitive semantics. We consider a couple of examples of this approach to show why this alternative may lead to undesirable situations and explain why the semantics given earlier was chosen as the appropriate semantics.

Consider again the diagram in Figure 12. Recall that the intuitive interpretation is that $X = A \cap B$ and $Y = A \cap C$ and that $A \cap B$ and $A \cap C$ are disjoint. In the alternative interpretation, $\kappa(X) = A \cap \overline{Y}$ (or $A - Y$) and $\kappa(Y) = A \cap \overline{X}$. The Projection Condition gives:

$$X = B \cap A \cap \overline{Y}$$
$$Y = C \cap A \cap \overline{X}$$

Substituting for $\overline{Y}$ in the first equation gives

$$\begin{aligned} X &= B \cap A \cap \overline{C \cap A \cap \overline{X}} \\ &= B \cap A \cap (\overline{C} \cup \overline{A} \cup X) \\ &= B \cap A \cap (\overline{C} \cup X) \end{aligned}$$

We can use the following lemma from set theory, stated without proof, to solve this set equation.

Lemma 1 For given sets *A*, *B*, the set equation

$$X = A \cap X \cup B$$

has solution

$$X = A \cap S \cup B$$

where *S* is any set. The "minimal" solution is thus $X = B$ and the "maximal" solution is $X = A \cup B$.

By Lemma 1, we have

$$X = B \cap A \cap (\overline{C} \cup S)$$

for any set *S*. Now, substituting for $\overline{X}$ in the second of the original equations gives

$$\begin{aligned} Y &= C \cap A \cap \overline{B \cap A \cap (\overline{C} \cup S)} \\ &= C \cap A \cap (\overline{B} \cup \overline{A} \cup C \cap \overline{S}) \\ &= C \cap A \cap (\overline{B} \cup C \cap \overline{S}) \\ &= C \cap A \cap (\overline{B} \cup \overline{S}) \end{aligned}$$

which is interesting because of the asymmetry of the solutions. If we let $S = \varnothing$, we get $X = A \cap B \cap \overline{C}$, which is probably what our intuition tells us, but $Y = A \cap C$, which is counter-intuitive, because of the asymmetry. What the solution is giving us is the projections of *B* and *C* in *A*, with the added condition that these two projections are disjoint. The semantics does not give us that $A \cap B$ and $A \cap C$ are disjoint (i.e., $A \cap B \cap C = \varnothing$), but that any element in $A \cap B \cap C$ is arbitrarily placed in either the projection *X* or the projection *Y* (but not both). This is a palpably reasonable interpretation, but rather counter-intuitive and non-deterministic.

For a second example, consider Figure 21. *X* and *Y* are projections with determining labels *A* and *B* respectively. How do we now interpret projections *X* and *Y*? With the new interpretation, $\kappa(X) = C \cup Y$ and $\kappa(Y) = D \cup X$. So, by the semantics of projections, we have

$$X = A \cap (C \cup Y)$$
$$Y = B \cap (D \cup X)$$

Substituting for Y in the first equation and using Lemma 1, we obtain

$$X = A \cap (C \cup B \cap D) \cup A \cap B \cap S$$

and similarly, we obtain

$$Y = B \cap (D \cup A \cap C) \cup A \cap B \cap S$$

where S is any set.

Unfortunately, there are many possible solutions. There is, however, a unique "minimal solution" in this case:

$$X = A \cap (C \cup B \cap D)$$
$$Y = B \cap (D \cup A \cap C)$$

This is the intuitive interpretation and agrees with the solution given earlier, except that we do not have

$$A \cap D \cap \overline{C} \cap \overline{B} = \varnothing \text{ and } B \cap C \cap \overline{D} \cap \overline{A} = \varnothing .$$

However, the amount of work it took to get there and the problem of producing the general case, together with the problems involving the disjoint projections, means that this version of the semantics is problematic. Although it does have some things going for it:

- it would allow as legal the diagram in Figure 13, as the context for Y would now be different from the context for Z (but whether you would want this diagram to be legal is another question and if so, does this semantics agree with the intuitive semantics?)
- there are fascinating mathematical intricacies in this approach to the semantics!

The original version of the semantics is simple and always gives an intuitive interpretation.

7. Conclusion and Related Work

We have introduced the concept of projections into Venn-Euler and related diagrammatic systems and have given them simple formal and intuitive semantics. Projections form an integral part of spider diagrams and constraint diagrams. Constraint diagrams have been used, in conjunction with UML, in the modelling of telecommunications systems for industry and projections have proved invaluable in allowing complicated invariants to be expressed with clarity. Formal semantics have been given for spider diagrams [2] and are currently being produced for constraint diagrams. Diagrammatic reasoning rules have been developed for spider diagrams [8] and these have been proved sound and complete for a subset of the notation [7]. Reasoning rules involving projections are currently being developed.

8. Acknowledgements

We gratefully acknowledge Elena Tulchinsky and Virginia Plowman for comments on earlier versions of this paper. Authors Howse and Kent acknowledge some support from the UK EPSRC grant number GR/M02606. The research of the first author was supported by the Vice Provost fund for support of research at the Technion.

9. References

1. Euler, L (1772) *Lettres a Une Princesse d'Allemagne*. Vol. 2, *Sur divers subject de physique et de philosophie*, Letters No. 102-108. Basel, Birkhauser.
2. Gil, Y., Howse, J., Kent, S. (1999) Formalizing Spider Diagrams, *Proceedings of IEEE Symposium on Visual Languages* (VL99), IEEE Press.
3. Gil, Y., Howse, J., Kent, S. (1999) Constraint Diagrams: A step beyond UML, *Proceedings of TOOLS USA 99*, IEEE Press.
4. Grünbaum B. (1992a), *Venn Diagrams I*, Geombinatorics, Volume I, Issue 2, (1992) 5-12.
5. Grünbaum B. (1992b), *Venn Diagrams II*, Geombinatorics, Volume II, Issue 4, (1992) 25-32
6. Hammer, E.M. (1995) *Logic and Visual Information*, CSLI Publications.
7. Howse, J., Molina, F., Taylor, J. (1999) A Sound and Complete Diagrammatic Reasoning System, *Submitted to LICS 2000*.
8. Howse, J., Molina, F., Taylor, J., Kent, S. (1999) Reasoning with Spider Diagrams, *Proceedings of IEEE Symposium on Visual Languages* (VL99), IEEE Press.
9. Kent, S. (1997) Constraint Diagrams: Visualising Invariants in Object Oriented Models. *Proceedings of OOPSLA 97*
10. Lull, R. (1517) *Ars Magma*. Lyons.
11. More, T. (1959) On the construction of Venn diagrams. *Journal of Symbolic Logic*, 24.
12. Peirce, C (1933) *Collected Papers*. Vol. 4. Harvard University Press.
13. Rumbaugh, J., Jacobson, I., Booch, G. (1999) *Unified Modeling Language Reference Manual.* Addison-Wesley
14. Shin, S-J (1994) *The Logical Status of Diagrams*. CUP.
15. Venn, J (1880) On the Diagrammatic and Mechanical Representation of Propositions and Reasonings, *Phil. Mag.* S. 5 Vol. 9 No. 59.
16. Warmer, J. and Kleppe, A. (1998) *The Object Constraint Language: Precise Modeling with UML*, Addison-Wesley.