Expressiveness of Spider Diagrams

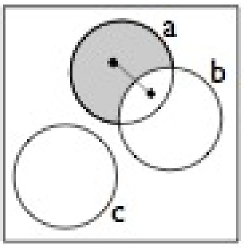
Gem Stapleton & Simon Thompson

12 February 2007

Overview

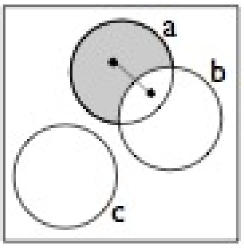
- Spider diagrams: examples, what and why.
- Logical questions: reasoning and expressiveness.
- Equivalences, and the role of negation.
- Monadic first-order logic.
- SD to MLE.
- Examining models of MLE.
- Model classes as spider diagrams.
- MLE to SD
- Summing up.

Spider diagrams



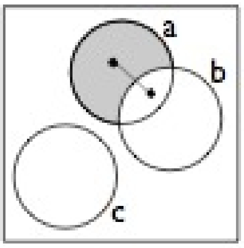
- Describe sets, their membership and interrelations.
- The absence of a zone means that it is empty: e.g. a∩c.
- Shading means that a zone contains no more elements than indicated (upper limit).
- A spider represent an element at one of its feet.

Spider diagrams



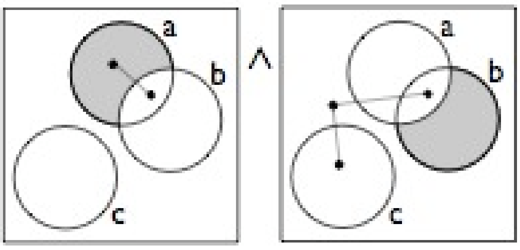
- a contains at least one element.
- a-b contains at most one element.
- a∩c, b∩c are empty.

Unitary spider diagrams



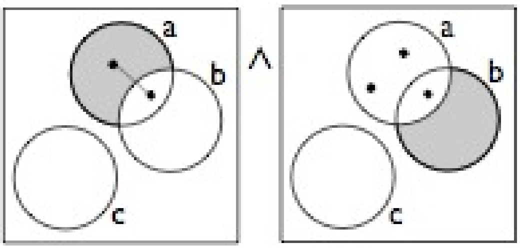
- Every unitary spider diagram is satisfiable. For example:
- a = {1,2,3}, b = {2,3}, c = {4,5}.
- Minimal models can be read off from the diagram.
- a = b = {1}, c = {}.
- a = {1}, b = c = {}.

General spider diagrams



- Combine unitary diagrams with ∧, ∨ (and ¬).

General spider diagrams



- Not all spider diagrams are satisfiable.

Why spider diagrams?

- Improvement on Venn diagrams, which quickly become unreadable because all 2^n zones need to be shown.
- Useful in visual description of set-theoretic constraints, such as those that apply in OO modelling.
- Very pleasant logical properties.
- Intuitive.
- A case study for more complicated visual representations, of constructs like ∃ and ∀.

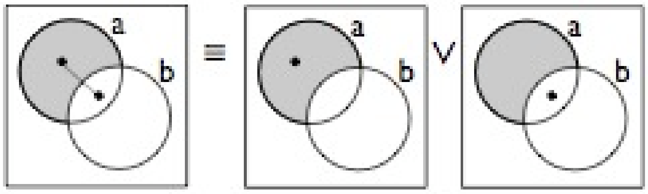
Logical questions

- Which diagrams are satisfiable (i.e. true in some model)?
- Which diagrams are valid (i.e. true in all models)?
- Is there a notion of logical equivalence between pairs of diagrams?
- Is there a notion of logical consequence?
- What properties do equivalence and consequence have?
- What can be expressed using spider diagrams?

Logical equivalences

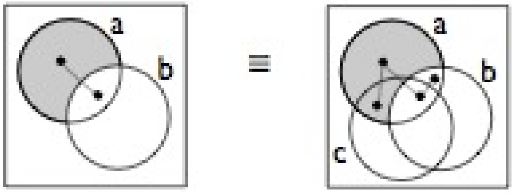
- Splitting spiders: remove the disjunction implicit in a spider, and turn it into an explicit disjunction.
- Adding a contour: a contour can be added, with various adjustments.
- A shaded empty zone can be added to or removed from the diagram.
- etc ...

Splitting spiders



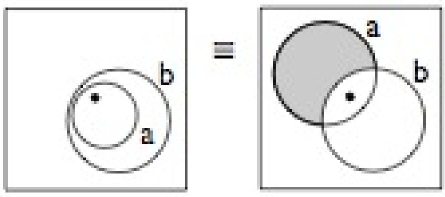
- Splitting spiders: remove the disjunction implicit in a spider, and turn it into an explicit disjunction.

Adding a contour



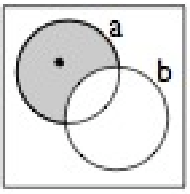
- A contour can be added, with all appropriate intersections.
- Spider feet need to be replicated over any new zones created.

Shaded empty zones



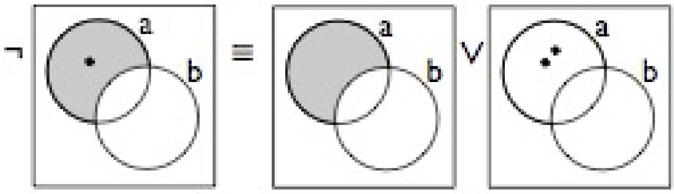
- A non-existent zone can be replaced by a shaded empty zone, and vice versa.

Why not ¬?



- Negation is a derived operator.
- a-b contains exactly one element.
- negation: a-b contains 0, or 2+ ...

Why not ¬?



- For a general spider diagram, split all spiders into disjunctions of α-diagrams, where each spider has one foot.
- Then use de Morgan's laws repeatedly to push all negations inwards to the α-diagrams, and negate as above.

Expressiveness

- What can be expressed by means of spider diagrams?
- Can we find a more traditional, non visual, logical system which has the same expressive power?
- Relate systems by translation from one system to another…
- … or by showing how the models of the formulas of the two systems are related.

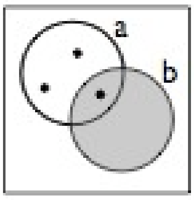
Monadic logic (MLE)

- A monad is a single unit.
- Monadic logic: one-place predicates only.
- Atomic formulas: $P(x)$, $x=y$
- Propositional combinations: $F \wedge G$, $F \vee G$, $F \Rightarrow G$, $\neg F$
- Quantifications: $\exists x.F$, $\forall x.F$

Spider diagrams to MLE

- Each contour is a predicate.
- Enough to translate α-diagrams … the rest is logical combination.
- Missing zones: state that there's no element.
- Spiders: distinct spiders are distinct elements; not the usual rule for $\exists$.
- Shaded areas: express the upper bound here using the $\forall$ quantifier.

Example: SD to MLE



- $\exists x. \exists y. (A(x) \land \neg B(x) \land A(y) \land \neg B(y) \land x \neq y)$
- $\exists x. (A(x) \land B(x) \land \forall y. ((A(y) \land B(y)) \Rightarrow x=y)$
- $\neg \exists x. (\neg A(x) \land B(x))$

SD = MLE ?

- The translation shows that spider diagrams are no more expressive than MLE, so SD $\subseteq$ MLE.
- MLE $\subseteq$ SD?
- One mechanism is to give a translation from MLE to SD … no obvious way of doing this.
- Another route: look at the sets of models for MLE formulas and for spider diagrams.

MLE: first key insight

- N nested quantifiers introduce N names.
  - A sentence of the form $Qx_1Qx_2 ...Qx_N.F$ can talk about at most N individuals: $x_1, x_2, ... x_N$.
  - Using '=' these $x_1, x_2, ... x_N$ can be made distinct.
- Examples:
- $\exists x. \exists y. (A(x) \land \neg B(x) \land A(y) \land \neg B(y) \land x \neq y)$
- $\exists x. (A(x) \land B(x) \land \forall y. ((A(y) \land B(y)) \Rightarrow x=y)$
- $\neg \exists x. (\neg A(x) \land B(x))$

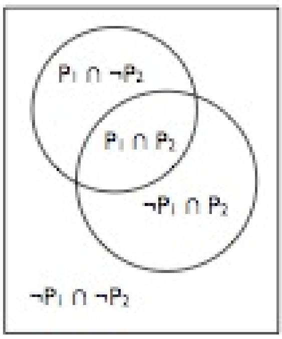
MLE: second key insight

- A structure M for a sentence using $P_1, \dots P_k$ is characterised by the sets $M(X) = \cap\{{}^M P_i \mid i \in X\} \cap$

  $\cap\{U - {}^M P_i \mid i \notin X\}$

  where X ranges over subsets of $\{1,2, \dots k\}$.

  - X ranges over all the possible combinations of P_i / $\neg P_i$.



Hypothesis

- MLE formulas involving $P_1, P_2, \dots, P_k$ have the effect of describing each of the 2^k regions given by a choice of $X \subseteq \{1, \dots, k\}$.
- MLE formulas involving quantifier nesting N can control the size of M(X) to be $0, 1, 2, \dots, N\text{-}1, \geq N$.
- Models of MLE formulas are either *small* or *large*, i.e. larger than $N.2^k$.
- If large, then can add elements to all large regions and remain a model.

Definition

Two structures M_1 and M_2 are *similar with respect to the sentence S*, with quantifier nesting N and using predicates $P_1, P_2, \dots P_k$, iff

for all $X \in P(\{1,2, \dots k\})$,
$M_1(X) = M_2(X)$ or $|M_1(X) \cap M_2(X)| \geq N$

for all $X, Y \in P(\{1,2, \dots k\})$, $X \neq Y$
$M_1(X) \cap M_2(Y) = \varnothing$

Result

If M_1 and M_2 are similar with respect to the sentence S, then for any subformula G of S and assignment to the free variables of G of values in $U_1 \cap U_2$,

M_1 models G if and only if M_2 models G.

Proof: by induction on the structure of G.

Proof sketch

The crucial case: when G has the form $\exists x.H$.

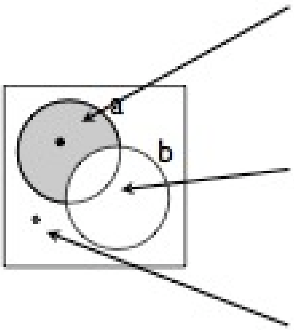
There's witness *a* for H lying in $M_1(X)$.

If $M_1(X)$ is small; *a* is also a witness in M_2.

If $M_1(X)$ is large, and *a* in U_2, we're done.

If a not in U_2, can pick an unmentioned *b* in $M_2(X)$, and argue that *b* is itself a witness in M_2, by automorphism argument.

Spider diagram models



This zone has upper and lower limits on its size.

This zone has no limits on its size.

This zone has a lower limit on its size.

Models of MLE formulas

- Models of MLE formulas are either *small* or *large*, i.e. larger than $N.2^k$.
- If large, then can add elements to all large regions and remain a model.

MLE ⊆ SD

- Represent each small model as a fully shaded unitary SD.
- Represent each (minimal) large model as a unitary SD with shading in small regions.
- Disjunction of these gives the representing spider diagram.

Constructive?

- *Yes*: run through all the potential small and minimal large models, checking whether they are indeed models.
- *No*: not in the sense that there's a direct, syntactic algorithm going from MLE sentences to spider diagrams.

Refinements

- Can make the representing SD substantially smaller by applying spider-creating rules, and coalescing small and large models when possible.

Commentary

- Like a quantifier elimination procedure, but not quite the same.
- QE procedure can be used directly on ML formulas; doesn't seem to generalise to MLE.

Conclusion

- Spider diagrams give an attractive, intuitive representation of properties of sets and their elements.
- Spider diagrams are obviously examples of sentences in monadic first-order logic with equality.
- Less obviously, all MLE sentences can be represented by spider diagrams.
- The proof of this goes via an analysis of the class of models of MLE sentences, and thus to a set of SDs characterising the model class.