Applied Logic in the Undergraduate Mathematics Curriculum

Larry Moss, Indiana University

AMS-MAA-MER subsession on Logic in the Undergraduate Curriculum, January 7, 2011

- The emergence of applied logic as a research trend makes it possible to re-conceptualize and reorganize logic in the undergraduate curriculum.
- I am not sure whether this is more applicable for majors or non-majors.

Applied logic is the application of logical and mathematical methods to foundational matters that go beyond the traditional areas of mathematical logic.

The central domain of application
at the present time is computer science, but it
also has significant applications in other fields.

It extends of the boundaries
of logic to include change, uncertainty,
fallibility, and community.

Its ultimate interest is a concern with human reasoning, so it will ultimately lead to a rapprochement with psychology, artificial intelligence, and cognitive science.

But even before this happens, the development of
tractable logical systems
are the most conspicuous applications of logic in many fields.

Applied logic is an interdisciplinary field, and this has its own set of difficulties and opportunities.

Applied logic is an interdisciplinary field, and this has its own set of difficulties and opportunities.

Mathematics and logic, but not mathematical logic.

For a somewhat contentious overview, one might see my "Applied Logic: A Manifesto".

Why teach applied logic to undergraduates?

My goals

- Explain real mathematics based on applications.
  to college students who are not math majors
- Give some math training and background to students
  who will encounter logic in their further studies.
- ~~Provide a course for math majors on logic.~~

In the rest of this talk, I want to mention my experiences teaching a few classes.

- Modal Logic, aimed at presenting dynamic epistemic logic to students interested in logic, philosophy, cognitive science, AI, etc.
- Mathematics from Language, aimed at connections to linguistics.
- Logic from Language, aimed at math students who want a strong introduction to logic based on reasoning in language
- Computability Theory for Majors and Non-Majors.

I have taught all but Logic from Language in the past few years.

Getting Started in An Applied Logic Version of Modal Logic

a "card scenario"

We have a deck with three cards ♥, ♣, and ♦.

We also have three players: *B*, *C*, and *D*.

We deal the cards out, one to each player.

The deal is face down, and then the players look.

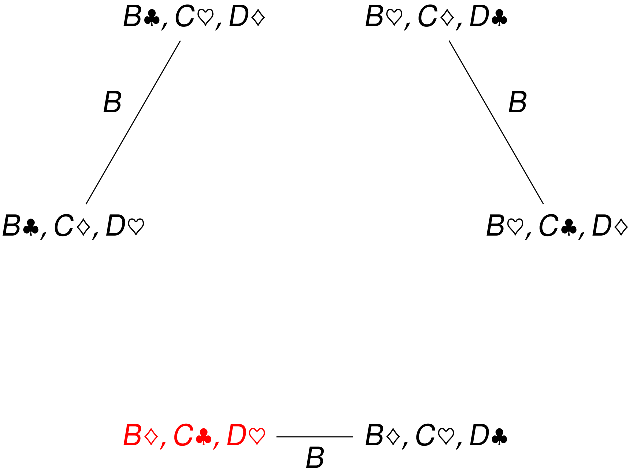
$B\clubsuit, C\heartsuit, D\diamondsuit$ $B\heartsuit, C\clubsuit, D\diamondsuit$

$B\clubsuit, C\diamondsuit, D\heartsuit$ $B\heartsuit, C\diamondsuit, D\clubsuit$

$B\diamondsuit, C\clubsuit, D\heartsuit$ $B\diamondsuit, C\heartsuit, D\clubsuit$

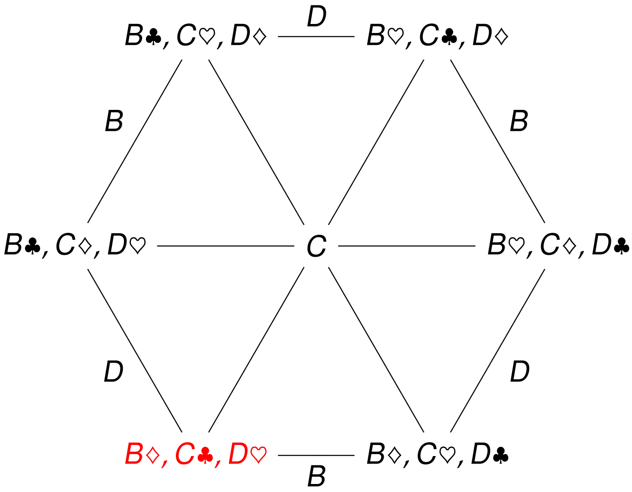
The space of possible deals

with B's indifference relation, and with the real world in red



The space of possible deals

with everyone's indifference relation,
but omitting loops on all six nodes, each labeled B, C, and D



We have a formal language built from atomic sentences

$$\begin{array}{lll} B\heartsuit, & B\clubsuit, & B\diamondsuit, \\ C\heartsuit, & C\clubsuit, & C\diamondsuit, \\ D\heartsuit, & D\clubsuit, & D\diamondsuit \end{array}$$

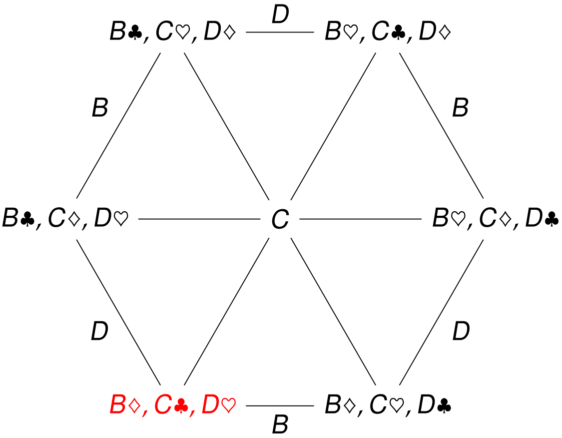
using the usual logical symbols and the knowledge operators K_B, K_C and K_D.

Here is the main formal definition of the semantics of our language.

For x is a node in one of the pictures:

$$\begin{array}{lll} x \models p & \text{iff} & p \text{ is written on } x \ (p \text{ atomic}) \\ \vdots & & \\ x \models K_B\,\varphi & \text{iff} & y \models \varphi \text{ for all } y \text{ such that } x \stackrel{B}{—} y \end{array}$$

Easy examples



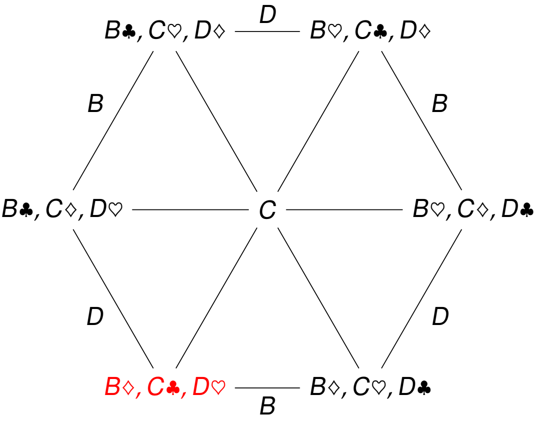
$B\clubsuit, C\diamondsuit, D\heartsuit \models B\clubsuit$

$B\diamondsuit, C\heartsuit, D\clubsuit \models B\diamondsuit \wedge D\clubsuit$

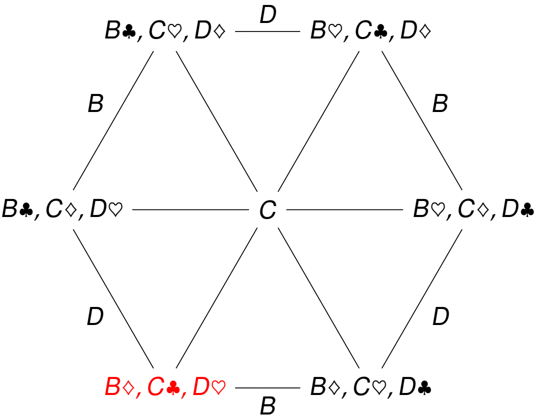
$B\clubsuit, C\heartsuit, D\diamondsuit \models \neg(C\clubsuit \vee C\diamondsuit)$

$B\clubsuit, C\heartsuit, D\diamondsuit \models C\heartsuit \rightarrow D\diamondsuit$

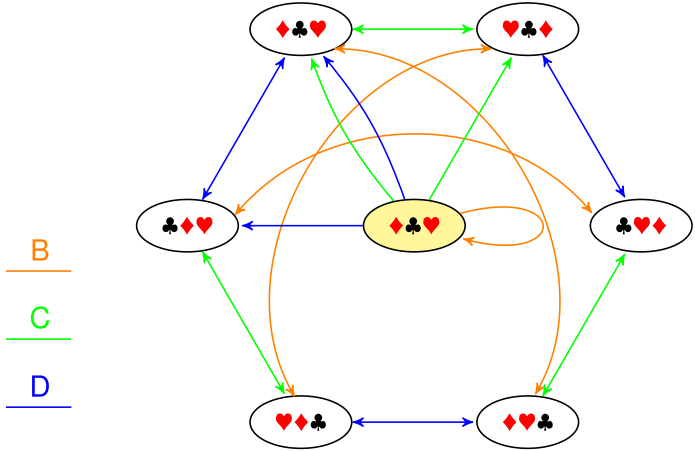
A harder example



B♢, C♣, D♡ $\models K_B$ B♢
because the only y such that B♢, C♣, D♡ is B-connected to y are B♢, C♣, D♡ and B♢, C♡, D♣,
and at both of those, B♢ is true.



$B\diamondsuit, C\clubsuit, D\heartsuit \not\models K_B\ C\clubsuit$



$B\diamondsuit, C\clubsuit, D\heartsuit \models K_B\ (C\clubsuit\ \wedge K_D \neg K_B C\clubsuit)$

- Kripke models for knowledge, belief, common knowledge, time, preferences, conditionals, ...
- Lots of use of representational pictures and connections to formal logical languages.
- Some use of computers (logics workbench).
- Basic math theory: relations, preorders.
- Basic (meta)-logic skills including: logical systems, models, consistent and satisfiable sets of sentences.
  Also, formal proofs in natural deduction style.
- I have a new approach to completeness, using finite versions of canonical models.
  This is only for students with some math background.

I have lecture slides and chapters for an eventual textbook.

A similar book, but with a much broader scope and for more sophisticated students:
Johan van Benthem, Modal Logic for Open Minds,
CSLI Publications, 2010.

- Teach math topics motivated entirely by linguistics.
- Formal grammars, automata, some probability, logic
- One high point: the equivalence of regular languages with one-directional Lambek grammars
- Another high point: mathematics connected to determiners
- I teach this for non-math students, so I omit induction and all hard proofs.
- Theory is motivated by many examples.

- Teach math topics motivated entirely by linguistics.
- Formal grammars, automata, some probability, logic
- One high point: the equivalence of regular languages with one-directional Lambek grammars
- Another high point: mathematics connected to determiners
- I teach this for non-math students, so I omit induction and all hard proofs.
- Theory is motivated by many examples.

A very serious introduction to this may be found in Edward L. Keenan and LM, Mathematical Structure in Language.

- It is possible to teach a class in logical systems based on natural language.
- The class could have connections to model theory, proof theory, computational complexity, algebraic logic, and beyond.
- I have had two math REU students (CMU and U. Alabama) on this topic.

- It is possible to teach a class in logical systems based on natural language.
- The class could have connections to model theory, proof theory, computational complexity,
algebraic logic, and beyond.
- I have had two math REU students (CMU and U. Alabama) on this topic.

$$\dfrac{\textit{All } n \textit{ are } q \qquad \dfrac{\dfrac{\textit{All } n \textit{ are } p \quad \textit{Some } n \textit{ are } n}{\textit{Some } n \textit{ are } p}}{\textit{Some } p \textit{ are } n}}{\textit{Some } p \textit{ are } q}$$

What do you think about this one?

$$\frac{\textit{All skunks are mammals}}{\textit{All who fear all who respect all skunks fear all who respect all mammals}}$$

It *follows*, using an interesting antitonicity principle:

$$\frac{\textit{All skunks are mammals}}{\textit{All who respect all mammals respect all skunks}}$$

It follows, using an interesting antitonicity principle

All skunks are mammals

All who respect all mammals respect all skunks

All who fear all who respect all skunks fear all who respect all mammals

It follows, using an interesting antitonicity principle

$$\dfrac{\dfrac{\textit{All skunks are mammals}}{\textit{All who respect all mammals respect all skunks}}}{\textit{All who fear all who respect all skunks fear all who respect all mammals}}$$

I have taught pieces of this subject for a few years, and also given a short course on it.

I have an ongoing textbook.

The subject is attracting attention these days, so I expect to work on educational materials.

- In content, this is the most traditional course in my list.
- The pedagogic approach is the only (slight) novelty.
- The idea is to continue Barwise and Etchemendy's Turing's World,
  especially their emphasis on learning by programming.
- This also led to the very popular JFLAP applets in CS education,
  and seem close to my course.
- But I use register machines and an ad hoc programming language that allows
  programs to manipulate programs directly (no coding).

Quick summary of my expericencs

My goals, again:

- Explain real mathematics based on applications.
  to college students who are not math majors
  This is partly successful as a course not requiring calculus.

- Give some math training and background to students who will encounter logic in their further studies.
  This is very successful.

- ~~Provide a course for math majors on logic.~~
  This could be done, but it's not what I'm after.

The work is interdisciplinary, so it doesn't even have to be taught in mathematics.

I'd like to think that someday applied logic courses will be the mainstream presentation.

But I can't make this happen myself!