**Reasoning with Uncertainty: Non-Monotonic Reasoning**

## What is reasoning?

* When we require any knowledge system to do something it has not been explicitly told how to do it must reason.
* The system must figure out what it needs to know from what it already knows.

We have seen simple example of reasoning or drawing inferences already. For example if we know: Robins are birds.

All birds have wings. Then if we ask: Do robins have wings?

Some reasoning (albeit very simple) has to go on answer the question.

## How can we reason?

To a certain extent this will depend on the knowledge representation chosen. Although a good knowledge representation scheme has to allow easy, natural and plausible reasoning. Listed below are very broad methods of how we may reason. We will study specific instances of some of these methods in the next few lectures.

**Formal reasoning**

-- Basic rules of inference with logic knowledge representations.

**Procedural reasoning**

-- Uses procedures that specify how to perhaps solve (sub) problems.

**Reasoning by analogy**

-- Humans are good at this, more difficult for AI systems. E.g. If we are asked Can robins fly?. The system might reason that robins are like sparrows and it knows sparrows can fly so ...

**Generalisation and abstraction**

-- Again humans effective at this. This is basically getting towards learning and understanding methods.

**Meta-level reasoning**

-- Once again uses knowledge about about what you know and perhaps ordering it in some kind of importance.

## Uncertain Reasoning?

Unfortunately the world is an uncertain place.

Any AI system that seeks to model and reasoning in such a world must be able to deal with this.

In particular it must be able to deal with:

* Incompleteness -- compensate for lack of knowledge.
* Inconsistencies -- resolve ambiguities and contradictions.
* Change -- it must be able to update its world knowledge base over time.

Clearly in order to deal with this some decision that a made are more likely to be true (or false) than others and we must introduce methods that can cope with this uncertainty.

There are three basic methods that can do this:

* Symbolic methods.
* Statistical methods.
* Fuzzy logic methods.

## Non-Monotonic Reasoning

Predicate logic and the inferences we perform on it is an example of monotonic reasoning.

In monotonic reasoning if we enlarge at set of axioms we cannot retract any existing assertions or axioms.

Humans do not adhere to this monotonic structure when reasoning:

* we need to jump to conclusions in order to plan and, more basically, survive.
  + we cannot anticipate all possible outcomes of our plan.
  + we must make assumptions about things we do not specifically know about.

### Default reasoning

This is a very common from of non-monotonic reasoning. Here We want to draw conclusions based on what is most likely to be true.

We have already seen examples of this and possible ways to represent this knowledge.

We will discuss two approaches to do this:

* Non-Monotonic logic.
* Default logic.

DO NOT get confused about the label Non-Monotonic and Default being applied to reasoning and a particular logic. Non-Monotonic reasoning is generic descriptions of a class of reasoning. Non-Monotonic logic is a specific theory. The same goes for Default reasoning and Default logic.

**Non-Monotonic Logic**

This is basically an extension of first-order predicate logic to include a modal operator, *M*. The purpose of this is to allow for consistency.

For example: tex2html_wrap_inline7154: plays\_instrument(*x*) tex2html_wrap_inline7400 improvises(*x*) tex2html_wrap_inline7156 jazz\_musician(*x*)

states that for all *x* is *x* plays an instrument and if the fact that *x* can improvise is consistent with all other knowledge then we can conclude that *x* is a jazz musician.

How do we define consistency?

One common solution (consistent with PROLOG notation) is

to show that fact *P* is true attempt to prove tex2html_wrap_inline7418. If we fail we may say that *P* is consistent (since tex2html_wrap_inline7418 is false).

However consider the famous set of assertions relating to President Nixon.

: Republican(*x*) tex2html_wrap_inline7428Pacifist(*x*) tex2html_wrap_inline7432Pacifist(*x*)



tex2html_wrap_inline7154: Quaker(*x*) tex2html_wrap_inline7400 Pacifist(*x*) tex2html_wrap_inline7156Pacifist(*x*)

Now this states that Quakers tend to be pacifists and Republicans tend not to be.

BUT Nixon was both a Quaker and a Republican so we could assert:

Quaker(Nixon)

Republican(Nixon)

This now leads to our total knowledge becoming inconsistent.

**Default Logic**

Default logic introduces a new inference rule:

tex2html_wrap_inline7448

which states if A is deducible and it is consistent to assume B then conclude C.

Now this is similar to Non-monotonic logic but there are some distinctions:

* New inference rules are used for computing the set of plausible extensions. So in the Nixon example above Default logic can support both assertions since is does not say anything about how choose between them -- it will depend on the inference being made.
* In Default logic any nonmonotonic expressions are rules of inference rather than expressions.

### Circumscription

Circumscription is a rule of conjecture that allows you to jump to the conclusion that the objects you can show that posses a certain property, *p*, are in fact all the objects that posses that property.

Circumscription can also cope with default reasoning.

Suppose we know: bird(tweety)

tex2html_wrap_inline7154: penguin(*x*) tex2html_wrap_inline7156 bird(*x*)

tex2html_wrap_inline7154: penguin(*x*) tex2html_wrap_inline7432flies(*x*)

and we wish to add the fact that typically, birds fly.

In circumscription this phrase would be stated as:

A bird will fly if it is not abnormal

and can thus be represented by:

tex2html_wrap_inline7154: bird(*x*) tex2html_wrap_inline7472abnormal(*x*) tex2html_wrap_inline7156 flies(*x*).

However, this is not sufficient

We cannot conclude

flies(tweety)

since we cannot prove

tex2html_wrap_inline7182abnormal(tweety).

This is where we apply circumscription and, in this case,

we will assume that those things that are shown to be abnormal are the only things to be abnormal

Thus we can rewrite our default rule as:

tex2html_wrap_inline7154: bird(*x*) tex2html_wrap_inline7472flies(*x*) tex2html_wrap_inline7156 abnormal(*x*)

and add the following

tex2html_wrap_inline7154: tex2html_wrap_inline7182abnormal(*x*)

since there is nothing that cannot be shown to be abnormal.

If we now add the fact:

penguin(tweety)

Clearly we can prove

abnormal(tweety).

If we circumscribe abnormal now we would add the sentence,

a penguin (tweety) is the abnormal thing:

tex2html_wrap_inline7154: abnormal(*x*) tex2html_wrap_inline7156 penguin(*x*).

Note the distinction between Default logic and circumscription:

Defaults are sentences in language itself not additional inference rules.

### Implementations: Truth Maintenance Systems

Due to Lecture time limitation. This topic is not dealt with in any great depth. Please refer to the further reading section.

A variety of Truth Maintenance Systems (TMS) have been developed as a means of implementing Non-Monotonic Reasoning Systems.

Basically TMSs:

* all do some form of dependency directed backtracking
* assertions are connected via a network of dependencies.

**Justification-Based Truth Maintenance Systems (JTMS)**

* This is a simple TMS in that it does not know anything about the structure of the assertions themselves.
* Each supported belief (assertion) in has a justification.
* Each justification has two parts:
  + An IN-List -- which supports beliefs held.
  + An OUT-List -- which supports beliefs not held.
* An assertion is connected to its justification by an arrow.
* One assertion can feed another justification thus creating the network.
* Assertions may be labelled with a belief status.
* An assertion is valid if every assertion in the IN-List is believed and none in the OUT-List are believed.
* An assertion is non-monotonic is the OUT-List is not empty or if any assertion in the IN-List is non-monotonic.

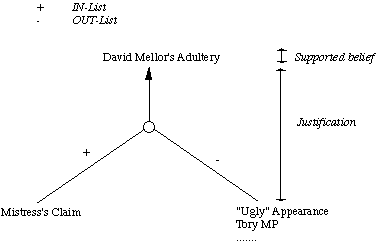


Fig. [20](http://www.cs.cf.ac.uk/Dave/AI2/node81.html#figJTMS) A JTMS Assertion

**Logic-Based Truth Maintenance Systems (LTMS)**

Similar to JTMS except:

* Nodes (assertions) assume no relationships among them except ones explicitly stated in justifications.
* JTMS can represent P and tex2html_wrap_inline7182P simultaneously. An LTMS would throw a contradiction here.
* If this happens network has to be reconstructed.

**Assumption-Based Truth Maintenance Systems (ATMS)**

* JTMS and LTMS pursue a single line of reasoning at a time and backtrack (dependency-directed) when needed -- depth first search.
* ATMS maintain alternative paths in parallel -- breadth-first search
* Backtracking is avoided at the expense of maintaining multiple contexts.
* However as reasoning proceeds contradictions arise and the ATMS can be pruned
  + Simply find assertion with no valid justification.

# Uncertain Reasoning: Statistical Methods

## Symbolic versus statistical reasoning

The (Symbolic) methods basically represent uncertainty belief as being

* True,
* False, or
* Neither True nor False.

Some methods also had problems with

* Incomplete Knowledge
* Contradictions in the knowledge.

Statistical methods provide a method for representing beliefs that are not certain (or uncertain) but for which there may be some supporting (or contradictory) evidence.

Statistical methods offer advantages in two broad scenarios:

**Genuine Randomness**

-- Card games are a good example. We may not be able to predict any outcomes with certainty but we have knowledge about the likelihood of certain items (e.g. like being dealt an ace) and we can exploit this.

**Exceptions**

-- Symbolic methods can represent this. However if the number of exceptions is large such system tend to break down. Many common sense and expert reasoning tasks for example. Statistical techniques can summarise large exceptions without resorting enumeration.

## Basic Statistical methods -- Probability

The basic approach statistical methods adopt to deal with uncertainty is via the axioms of probability:

* Probabilities are (real) numbers in the range 0 to 1.
* A probability of *P*(*A*) = 0 indicates total uncertainty in *A*, *P*(*A*) = 1 total certainty and values in between some degree of (un)certainty.
* Probabilities can be calculated in a number of ways.

Very Simply

Probability = (number of desired outcomes) / (total number of outcomes)

So given a pack of playing cards the probability of being dealt an ace from a full normal deck is 4 (the number of aces) / 52 (number of cards in deck) which is 1/13. Similarly the probability of being dealt a spade suit is 13 / 52 = 1/4.

If you have a choice of number of items *k* from a set of items *n* then the tex2html_wrap_inline7520 formula is applied to find the number of ways of making this choice. (! = factorial).

So the chance of winning the national lottery (choosing 6 from 49) is tex2html_wrap_inline7524 to 1.

* Conditional probability, *P*(*A*|*B*), indicates the probability of of event *A* given that we know event *B* has occurred.

### Bayes Theorem

* This states:

displaymath1496

* + This reads that given some evidence *E* then probability that hypothesis tex2html_wrap_inline7536 is true is equal to the ratio of the probability that *E* will be true given tex2html_wrap_inline7536 times the a priori evidence on the probability of tex2html_wrap_inline7536 and the sum of the probability of *E* over the set of all hypotheses times the probability of these hypotheses.
  + The set of all hypotheses **must** be mutually exclusive and exhaustive.
  + Thus to find if we examine medical evidence to diagnose an illness. We must know all the prior probabilities of find symptom and also the probability of having an illness based on certain symptoms being observed.

Bayesian statistics lie at the heart of most statistical reasoning systems.

How is Bayes theorem exploited?

* The key is to formulate problem correctly:

*P*(*A*|*B*) states the probability of *A* given only *B*'s evidence. If there is other relevant evidence then it **must** also be considered.

Herein lies a problem:

* All events must be mutually exclusive. However in real world problems events are not generally unrelated. For example in diagnosing measles, the symptoms of spots and a fever are related. This means that computing the conditional probabilities gets complex.

In general if a prior evidence, *p* and some new observation, *N* then computing

displaymath7532  
grows exponentially for large sets of *p*

* All events must be exhaustive. This means that in order to compute all probabilities the set of possible events must be closed. Thus if new information arises the set must be created afresh and allprobabilities recalculated.

Thus Simple Bayes rule-based systems are not suitable for uncertain reasoning.

* Knowledge acquisition is very hard.
* Too many probabilities needed -- too large a storage space.
* Computation time is too large.
* Updating new information is difficult and time consuming.
* Exceptions like ``none of the above'' cannot be represented.
* Humans are not very good probability estimators.

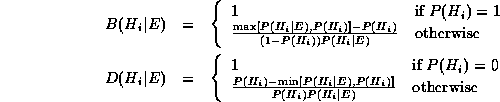
However, Bayesian statistics still provide the core to reasoning in many uncertain reasoning systems with suitable enhancement to overcome the above problems.

## Belief Models and Certainty Factors

This approach has been suggested by Shortliffe and Buchanan and used in their famous medical diagnosis MYCIN system.

MYCIN is essentially and expert system. Here we only concentrate on the probabilistic reasoning aspects of MYCIN.

* MYCIN represents knowledge as a set of rules.
* Associated with each rule is a certainty factor
* A certainty factor is based on measures of belief *B* and disbelief *D* of an hypothesis tex2html_wrap_inline7536 given evidence *E* as follows:



where tex2html_wrap_inline7566 is the standard probability.

* The certainty factor *C* of some hypothesis tex2html_wrap_inline7536 given evidence*E* is defined as:   
  displaymath1566

## Dempster-Shafer Models

This can be regarded as a more general approach to representing uncertainty than the Bayesian approach.

Bayesian methods are sometimes inappropriate:

Let A represent the proposition Demi Moore is attractive.

Then the axioms of probability insist that tex2html_wrap_inline7580

Now suppose that Andrew does not even know who Demi Moore is.

Then

* We cannot say that Andrew believes the proposition if he has no idea what it means.
* Also, It is not fair to say that he disbelieves the proposition.
* It would therefore be meaningful to denote Andrew's belief of *B*(*A*) and tex2html_wrap_inline7584 as both being 0.
* Certainty factors do not allow this.

### Dempster-Shafer Calculus

The basic idea in representing uncertainty in this model is:

* Set up a confidence interval -- an interval of probabilities within which the true probability lies with a certain confidence -- based on the Belief *B* and plausibility *PL* provided by some evidence *E* for a proposition *P*.
* The belief brings together all the evidence that would lead us to believe in *P* with some certainty.
* The plausibility brings together the evidence that is compatible with *P* and is not inconsistent with it.
* This method allows for further additions to the set of knowledge and does not assume disjoint outcomes.

If tex2html_wrap_inline7598 is the set of possible outcomes, then a mass probability, *M*, is defined for each member of the set tex2html_wrap_inline7602 and takes values in the range [0,1].

The Null set, tex2html_wrap_inline7606, is also a member of tex2html_wrap_inline7602.

**NOTE:** This deals wit set theory terminology that will be dealt with in a tutorial shortly. Also see exercises to get experience of problem solving in this important subject matter.

M is a probability density function defined not just for tex2html_wrap_inline7598 but for **em all** subsets.

So if tex2html_wrap_inline7598 is the set { Flu (F), Cold (C), Pneumonia (P) } then tex2html_wrap_inline7602 is the set { tex2html_wrap_inline7606, {*F*}, {*C*}, {*P*}, {*F*, *C*}, {*F*, *P*}, {*C*, *P*}, {*F*, *C*, *P*} }

* The confidence interval is then defined as [*B*(*E*),*PL*(*E*)]   
  where   
  displaymath1649  
  where tex2html_wrap_inline7640 i.e. all the evidence that makes us believe in the correctness of *P*, and   
  eqnarray1652  
  where tex2html_wrap_inline7644 i.e. all the evidence that contradicts *P*.

## Bayesian networks

These are also called Belief Networks or Probabilistic Inference Networks. Initially developed by Pearl (1988).

The basic idea is:

* Knowledge in the world is modular -- most events are conditionally independent of most other events.
* Adopt a model that can use a more local representation to allow interactions between events that only affect each other.
* Some events may only be unidirectional others may be bidirectional -- make a distinction between these in model.
* Events may be causal and thus get chained together in a network.

### Implementation

* A Bayesian Network is a directed acyclic graph:
  + A graph where the directions are links which indicate dependencies that exist between nodes.
  + Nodes represent propositions about events or events themselves.
  + Conditional probabilities quantify the strength of dependencies.

Consider the following example:

* The probability, tex2html_wrap_inline7674 that my car won't start.
* If my car won't start then it is likely that
  + The battery is flat or
  + The staring motor is broken.

In order to decide whether to fix the car myself or send it to the garage I make the following decision:

* If the headlights do not work then the battery is likely to be flat so i fix it myself.
* If the starting motor is defective then send car to garage.
* If battery and starting motor both gone send car to garage.

The network to represent this is as follows:

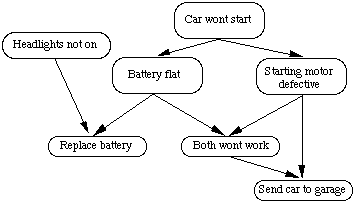


Fig. [21](http://www.cs.cf.ac.uk/Dave/AI2/node95.html#figbnet1) A simple Bayesian network

### Reasoning in Bayesian nets

* Probabilities in links obey standard conditional probability axioms.
* Therefore follow links in reaching hypothesis and update beliefs accordingly.
* A few broad classes of algorithms have bee used to help with this:
  + Pearls's message passing method.
  + Clique triangulation.
  + Stochastic methods.
  + Basically they all take advantage of clusters in the network and use their limits on the influence to constrain the search through net.
  + They also ensure that probabilities are updated correctly.
* Since information is local information can be readily added and deleted with minimum effect on the whole network. **ONLY** affected nodes need updating.

### A Practical Example

Here we describe a practical example from research based here in Cardiff.

We have used Bayesian Nets in a Computer Vision application. Details of the visual processes involved will be discussed later in the course so the contest will become clearer later.

Here we attempt to describe the Bayesian reasoning behind the process.

The goal is to perform a task called data fusion to obtain a segmentation -- a description of an object (viewed from a set of images) detailing its surface properties. In the example given here we deal with a simple cube. So the final description will hopefully list its edges and its faces and how they are connected together.

The input to the fusion process is three preprocessing stages that have extracted out edge information and planar surface information from 2D grey scale (monochrome) images and 3D range data.

So from these three pre-processes we have a list of all lines, curved or straight, a list of all line intersections (two or three line intersections) and a list of all the surface equations extracted from both image types. We can now build the network from these lists of features. As mentioned above, we hypothesise about extracted surfaces intersecting. For us to evaluate these hypotheses we need to have evidence to support or contradict them. The evidence that we use is :

* straight lines extracted from light image.
* curves extracted from light image.
* `areas of uncertainty' extracted from depth map.

The two lines lists are generated as described above. The areas of uncertainty are found when we are attempting to find the surface equations of each surface type. Errors are found in the depth map where the mask to find the general surface shape overlaps two or more surfaces, the error tends to be enlarged therefore, giving us a clue that a surface intersection exists in that general area. So we are using evidence from more than one source of data.

We proceed by taking each of the surfaces in the surface list and a node is generated to represent it. We then take a pair of surfaces and attempt to intersect them. If they are possibly intersecting then a `feature group' node is generated referencing the surfaces and connected to the children surface nodes. This process is repeated for each pair surfaces that we have extracted. We now want to attach a conditional probability to each of our new nodes. So we now know the surfaces that could *possibly* interact in the object. We now attach a probability to these connections. We do this by finding the equation of intersection, this will be a three dimensional line for two planes or an ellipse for a plane and a sphere, and project this onto our focal plane. Now we have our hypothesised intersections in the same dimension as our extracted lines from the preliminary stage. So we now find, for each intersecting line a closest match line from our line list. Once we have found the closest matching line we generate a probability from the error. So a line that closely matches our intersection line then we have a high probability whereas two surfaces that don't intersect in the object are unlikely to coincide with a line from the line list therefore giving us a low probability. The line that is found is also checked to see if it lies in an area of uncertainty. If it does then that is another strong clue that the line that we have found is actually where surfaces are joined.

So once we have generated this network with all the necessary links *etc.* any more information that is provided to the system can be added and the network will propagate this information throughout the network in the form of probability updating. So for example say a new image was provided from say a colour image and this image increased the likelihood of some edges and corners being present in the image then this would increase the probability of those features that are linked to those edges and corners which would propagate throughout the network. Figure [22](http://www.cs.cf.ac.uk/Dave/AI2/node97.html#figbnet2) shows us a simple example of the network that would be generated from the input data of edges and planar faces of the cube. As can be seen, the feature group nodes can represent groups that vary from single features such as line segments, surfaces or corners or the whole object is represented in the lower nodes which includes three surfaces, three line segments, three crosses and one corner.

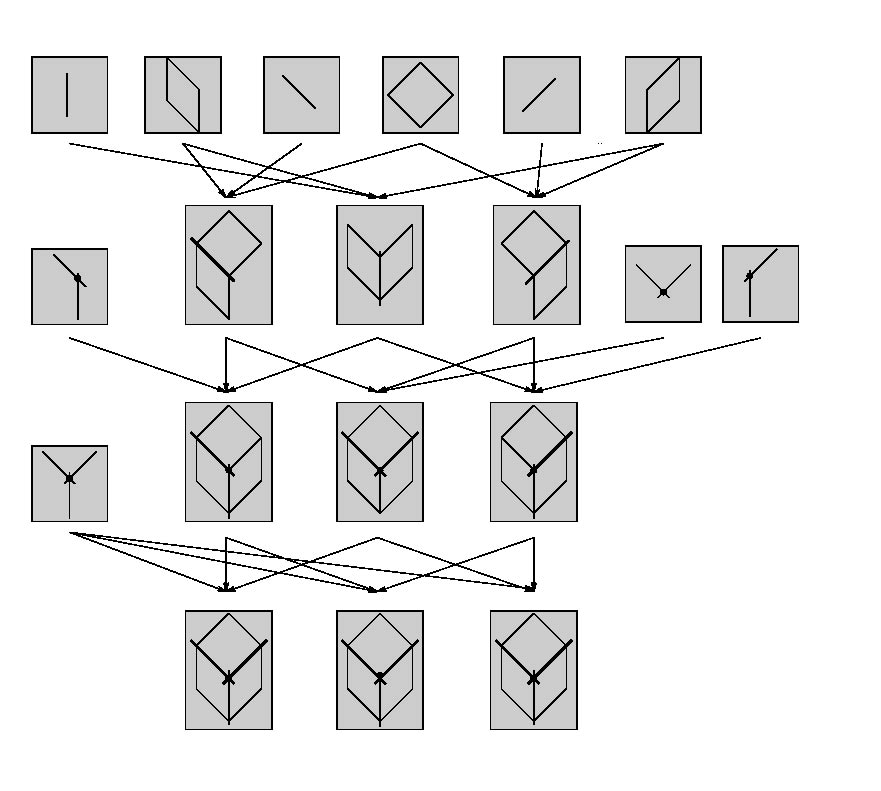


Fig. [22](http://www.cs.cf.ac.uk/Dave/AI2/node97.html#figbnet2) A Bayesian Network for Segmentation of a Cube

## Fuzzy Logic

This topic is treated more formally in other courses. Here we summarise the main points for the sake completeness.

Fuzzy logic is a totally different approach to representing uncertainty:

* It focuses on ambiguities in describing events rather the uncertainty about the occurrence of an event.
* Changes the definitions of set theory and logic to allow this.
* Traditional set theory defines set memberships as a boolean predicate.

### Fuzzy Set Theory

* Fuzzy set theory defines set membership as a possibility distribution.

The general rule for this can expressed as:

displaymath1725

where *n* some number of possibilities.

This basically states that we can take *n* possible events and us *f* to generate as single possible outcome.

This extends set membership since we could have varying definitions of, say, hot curries. One person might declare that only curries of Vindaloo strength or above are hot whilst another might say madras and above are hot. We could allow for these variations definition by allowing both possibilities in fuzzy definitions.

* Once set membership has been redefined we can develop new logics based on combining of sets etc. and reason effectively.