# Dependency Graphs

Oliver Richardson, oli@cs.cornell.edu

December 8, 2019

**Abstract**

We introduce Probabilistic Dependency Graphs, which can be regarded both as a probabilistic graphical model, and as a collection of soft, local, constraints. The additional representational flexibility allows us to represent both under and over-constrained belief states, as well as a modularity that makes the models easier to modify. They also have a clean theoretical backing, and reduce appropriately to existing, commonly used representations such as Bayesian Networks and constraint graphs. Some algorithms, notably including belief propagation, lift to the more general setting.

## Contents

**1 Introduction** **2**
- 1.1 Factor Graphs . . . 4
- 1.2 Benefits . . . 5

**2 Related Work** **5**

**3 Worlds** **5**

**4 Equivalent Definitions and Variants** **6**
- 4.1 Alternate Presentations . . . 8
  - 4.1.1 Random Variables . . . 8
- 4.2 Sub-stochastic Transitions . . . 8
  - 4.2.1 Relation to Partial Functions of $W$ . . . 10
  - 4.2.2 Reduction to Lower Probability Measures . . . 10

**5 Semantics** **10**
- 5.1 As Sets of Distributions . . . 10
- 5.2 As Weighted Distributions . . . 10
- 5.3 As Distributions . . . 10

**6 Relations to Other Graphical Models** **11**
- 6.1 Bayesian Networks . . . 11
- 6.2 Factor Graphs . . . 11

**7 Relations to Other Representations of Uncertainty** **11**
- 7.1 Sets of Probability Measures . . . 11

**8 Using Inconsistency** **11**

**9 Algorithms** **12**
- 9.1 Belief Propagation . . . 12

**10 Discussion** **12**
- 10.1 Inconsistency . . . 12

# 1 Introduction

Being a perfect Bayesian agent, while admirable, is hard. At the beginning of life, you must conceptualize the set of all worlds you could ever consider possible at any point in your existence. Then, every time you acquire knowledge, you need to know exactly how much you trust it, and do substantial book-keeping just to maintain a valid mental state in the form of a prior over these possible worlds. Such hurdles illustrate a very serious *lack of modularity*: there is no way to conceptualize new worlds, and it is only possible to combine observations with a full prior (that you trust). Combining partial observations, looking at only part of your knowledge, or adopting beliefs without considering its impact on all possible worlds all count as cheating, and worrying about a fault in your prior is unnecessary. Why endure such tribulations? The big payoff is *guaranteed consistency*. Having inconsistent beliefs leads to arbitrarily bad conclusions, and leave you vulnerable to unbounded pain in the form of dutch book arguments; the engineering ethos suggests it is best to avoid such things by design.

their

Why is it cheating? What rule is broken

Why is it unnecessary? I doubt a Bayesian would agree.

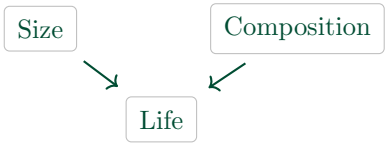
Why is it a stunt? This stunt is too difficult for humans, but many argue that we should approximate it as best we can, especially given the freedom to construct arbitrary artificial agents. Fortunately, there are many tools to make the problem tractable; of particular relevance are graphical models, which offer a compact, implicit representation of possible worlds as settings of random variables, and of a distribution over them in terms of local information. Graphical models even offer modularity (to various extents), but are still beholden to the restrictions above: updating is still restrictive and expensive, and consistency must still be enforced at all times, in a way which cripples promises of modularity.[1]

We have a graphical representation that doesn't suffer from these restrictions! So these restrictions are clearly not inherent to graphical models.

While we share the distaste for inconsistency, awareness of it can be a useful resource for seeking truth, and knowledge that there isn't even though there could have been, is a valuable safeguard. Worse than merely missing this opportunity, disallowing inconsistency by construction can sweep problems under the rug (see section 10.1 for the full argument). In light of this, we introduce a more general representation of knowledge and uncertainty in the form of a graphical model capable of representing inconsistent belief states, which we call Probabilistic Dependency Graphs (PDGs). Special cases of PDGs include Bayesian Networks and they can also mostly emulate (but are better behaved than) factor graphs. The extra flexibility makes them easier to use, and surprisingly, they enjoy additional properties which make them more useful than some specific variants, which we list in section 1.2. To get a feel for what PDGs are and how they differ from other graphical models, it is helpful to see examples.

**Example 1.1.** Suppose we have a belief about how size and composition affect the habitability of a planet: say we're astrobiologists, and we have some sense of how likely we are to find life on a given planet, supposing we knew its size (big or small) and its composition (mostly made of rocks vs gas). That is to say, we have a conditional probability table:

$$\Pr(\text{Life} \mid \text{Size}, \text{Comp}) = \begin{array}{c} \begin{matrix} \text{life} & \text{no life} \end{matrix} \\ \begin{bmatrix} .1 & .9 \\ .2 & .8 \\ .05 & 0.95 \\ 0.00001 & 0.99999 \end{bmatrix} \end{array} \begin{array}{l} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gasseous} \end{array}$$




This one conditional probability table cannot constitute a full Bayesian network. It is indeed a fragment of the Bayesian Network on the right, but in order to interpret the picture as a BN, it would be impossible not to also have a prior distribution over the variables 'Size' and 'Composition'—things we may not know

We can certainly have qualitative Bayesian networks that have no numbers at all. Now it's true that people don't tend to look at hybrids, where there are some cpts, but not all, but it's easy to make sense of this.

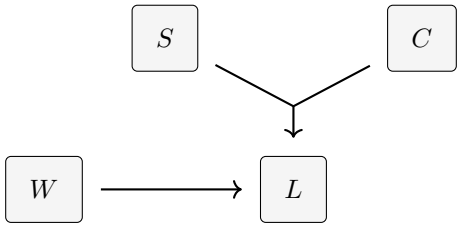
Premature. In any case, this is an important point that shouldn't be relegated to a footnote.

[1] This avoidance of inconsistency is actually the reason that adding arbitrary nodes to graphical models is problematic: to add a node to a network, it has to satisfy the appropriate independence properties, lest the network actually represent two distinct beliefs about one thing. Of course, knowing this for certain is a very strong assumption, and hence you cannot safely add nodes to graphical models without already knowing a lot about the variable—which begs the question: if your mental state is only what's in the graphical model, how did you get this knowledge of a new variable?

Let's avoid the philosophical question of what the graphical model represents.

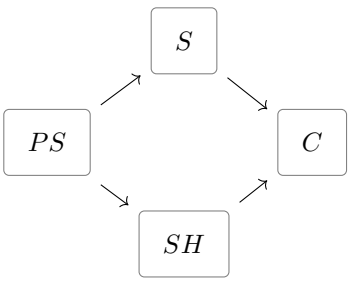
anything about.[2] While such a mental state is under-constrained from the perspective of probability, it is certainly falsifiable and *seems like knowledge*. This is perhaps the most common challenge to the idea that belief states are probability distributions, and by extension to the universality of the Bayesian Networks which represent them.

The opposite problem, though less appreciated, also occurs: just as it is impossible to represent an under-constrained distribution with a Bayesian Network, so too is it impossible to represent an over-constrained one. Patching this requires the more radical fix we present in the form of a PDG. Suppose our biologist friend now reminds us that life requires water, and gives us a probability estimate for the existence of life on a planet, with and without water. We trust this friend completely, and totally believe these probabilities. Unfortunately, but there's no way to place it as the parent of the $L$ node, because we don't know what the correlations are between water, size, and composition; neither are we prepared to give a probability of live given a full description of the three, and may not even have the space to keep such a thing.[3] Let $S, C, W, L$ be shortenings of Size, Composition, Water, and Life, respectively. Intuitively, we want to draw instead a picture that looks more like this:

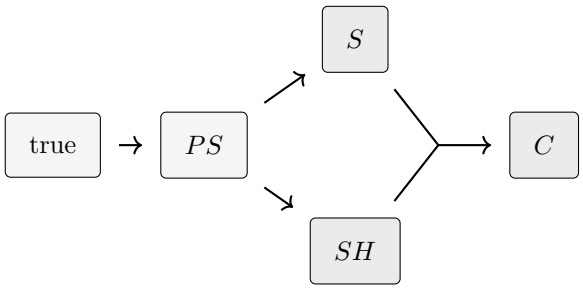



which represents having two conditional probability tables on $L$: one from $S \times C$ and the other from $W$. This would allow us to combine the two facts that we know, without also providing much more information than we're prepared to equivocate on. It is worth noting that there is now a possibility of being inconsistent, in the sense that is possible to specify the conditional distributions in the links in such a way that no joint distribution on all variables will marginalize out to them — for instance, if all estimates of $L$ from the $W$ are strictly smaller than any probability estimate of $L$ from $S \times C$. △

**Example 1.2.** Consider the classic example used to introduce Bayesian nets, in which the four variables are interest are booleans indicating whether a person ($C$) develops cancer, ($S$) smokes, ($SH$) is exposed to 2nd hand smoke, and ($PS$) has parents who smoke, presented graphically as follows:




(a) Bayesian Network




(b) Corresponding PDG

This is a compact representation of a joint distribution over all four variables, which achieves compactness by taking advantage of independence between variables. It encodes an assumption that every node is independent

[2] Models like these are well known to hardcore probabilists as well. In statistical learning theory, an 'incomplete' model like this is a discriminative (or conditional) model, as compared to a generative one.

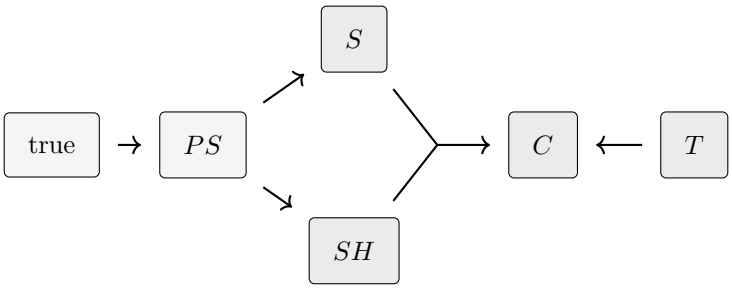
[3] If *Size* and *Composition* each had $\approx \sqrt{N}$ elements, and *Water* had $\approx N$ elements, it would be $O(N^2)$ to store a full joint table, compared to $O(N)$ for the two individual ones.

of its non-descendants given its parents.

Most of the time, we do not make the independence assumption because we know for certain that the variables are independent; rather, we just suspect that the identified links are by much more important than the others. Determining for sure that smoking and second hand smoke are independent, controlling for parents' smoking habits, would extremely difficult, and to do properly would require much more empiricism to validate. Why even bother jumping through this hoop? Because we wanted a BN to be shorthand for a probability distribution. But we have freed ourselves from these shackles, and make no such assumption.

The node on the far left is a special node which only takes one value, and allows us to represent unconditional distributions as arrows, visually making clear the difference between a lack of distribution, and an unconditional one.

Now, suppose you read a very thorough empirical study which demonstrates that people who use tanning beds have a 10% incidence of cancer, compared with 1% in the control. Just as in the previous example, this cannot be encoded directly into the Bayesian Network. The PDG on the other hand, has no trouble, and is simply the union of the two pieces of information:



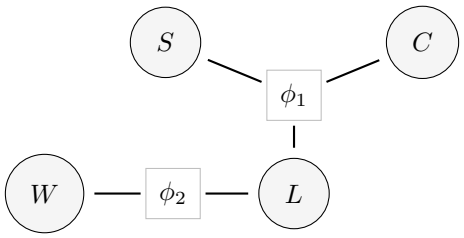

Note that the right half of the diagram (shaded slightly darker) has the same topology as in example 1.1. △

## 1.1 Factor Graphs

**Example 1.1** (continuing from p. 2)**.** The probabilists in us might not be willing to so easily give up the notion that this data ought to define a probability distribution, at least implicitly. Maybe there's something simple we can do to turn it into one? It turns out that we can (almost always) simultaneously get a distribution and commit to preserving the relative ratios of the specified probabilities within the links, while also more clearly exposing our independence assumptions. This can be done by treating the conditional distributions $p(L = l \mid S = s, C = c)$ and $p(L = l \mid W = w)$ as *factors*, which multiplied together give the relative probability density of any setting of variables $S \times C \times W \times L$

$$\Pr(s, c, w, l) \propto \phi_1(s, c, l)\phi_2(w, l)$$

where $\phi_1(s, c, l) = p(L = l \mid S = s, C = c)$, $\phi_2(w, l) = p(L = l \mid W = w)$. This can also be represented graphically, with a *factor graph*—a commonly used graphical model hailed as a strict generalization of Bayesian Networks and Markov networks.




In this diagram, circles represent variables, and the boxes represent factors that depend on variables they connect to. This is a lot more modular (we can add and remove factors as we like). We now have a distribution

that combines our beliefs, but this is not really what we were thinking of earlier. Beyond simply the inevitable effects of representing our knowledge as a distribution, such as forcing us to implicitly adopt marginal distributions over the variables $S, C$, and $W$, a product of factors has additional undesirable properties that are not shared by PDGs:

1. We can't weight the pieces of information differently. Though the scale of each factor $\phi_i$ gives us a degree of freedom in which to encode this information, it cannot be used, as $(a\phi_1)(b\phi_2) = (ab)(\phi_1\phi_2)$, and the aggregate coefficient $ab$ too is normalized out to form the distribution.
2. The resulting picture does not encode conditional probabilities in quite the way that we had wanted: now updating on $S$ does not preserve $L \mid C, S$, bringing $L$ along as required, but rather does something unclear and very global: we've lost the dependency structure we had in the first few pictures. Relatedly, we have lost the directedness of the edges, and with it, hope that the edges represent anything causal. Furthermore, the addition of new factors can dramatically change the meanings of existing ones. For all of these reasons, it is incredibly difficult to interpret part of the graph by itself.For instance, knowing the joint distribution does not determine the values of the factors.
3. If at least one factor is zero for every setting of $S, C, W, L$, no distribution is defined — in the face of inconsistency, the entire formalism ceases to work at all.

△

---

While factor graphs offer a solution of great generality, they sacrifice interpretability and important internal features of our original belief representation, so that they can represent distributions. This is not good, but much worse is the way that they sweep under the rug issues wherever possible. In example 1.1,

Akin to a in some ways destroy inconsistency that may have been a big deal had it come to the surface.

### 1.2 Benefits

1. We can represent both over-constrained and under-constrained mental states, both of which we argue are an important component of an agent's state.
2. Over-constrained models may be inconsistent; such inconsistencies provide a natural way of prescribing changes in mental state. Moreover, many standard algorithms, such as belief updating via Jeffrey's rule, as well as marginalization algorithms such as belief propagation, can be regarded as special cases of consistency reduction.
3. PDGs can emulate the functionality of not only other graphical models (such as Bayesian Networks, and to a large extent, factor graphs), but also other non-probabilistic notions of uncertainty.
4. The local interpretation of arrows makes it much less invasive to add, remove, and partially interpret parts of the model, compared to other graphical models.
5. In conjunction with agents to merge, split, and compress variables, we also make it possible for agents to design their own representations. With these tools, in conjunction with consistency.
6. The modularity makes it possible to add explicit rules to embed logic within the model.
7. In contrast with a simple collection of constraints, inconsistencies are local, and individual mistakes have limited impact on expectations.

## 2 Related Work

## 3 Worlds

Other than allowing for inconsistency, the biggest difference between Probabilistic Dependency Graphs and other graphical models is their interaction with the underlying space: a PDG .

The standard approach to probabilistic modeling is to start by selecting a measurable space of possible outcomes $\Omega$, and then put a normalized measure on it and compute desired quantities with it. Before you can begin to think about random variables, which are defined as set $X_i : \Omega \to V_i$ from outcomes to the set of values $V_i$ that $X_i$ can take on, you have to specify $\Omega$. This construction works well, so long as $\Omega$ is large enough to express everything you ever cared to conceptualize. Because agents are expected to have probability distributions over $\Omega$, the set of worlds that they consider possible must effectively stay constant over time, to use mechanisms such as conditioning as a sole way of changing a mental state.

Still, a we might wiggle our way around this using only classical probability: we could say that $\Omega$ is some very large set of outcomes that is guaranteed to be expressive enough to capture anything we care about, and then

To be clear, we've already given up on the possibility of being a Bayesian, because we clearly don't have priors on arbitrary concepts we haven't considered yet if we don't even know the extent of the space, but we might be able to do this with some under-constrained mechanism.

This strategy, works so long as you are an omniscient modeler. If you are modeling a system in which you know the set of all possible outcomes, either implicitly or explicitly, you can just collect them and use this to be $\Omega$, marginalize out appropriately, and let agents figure out their own distributions on subspaces of $\Omega$. Still, this is not entirely satisfying, for several reasons.

[todo: *most of these are unfinished thoughts, some need to be deleted*]

1. While it is true that the there will be subspaces of $\Omega$ which are isomorphic to the sets of worlds $W_i$ that the agents are modeling in their heads, the embedding is not at all clear [todo: ]
2. There may not be an omniscient modeler, and even if there were one, it seems very strange for an agent to have any access to it. Suppose you are using probabilities to describe real uncertainties in your life. To do this the standard way, you need to chose the subspace of the one true $\Omega$ [todo: *why is this problematic? for reasons other than inability to change?*]
3. Agents can never gain access via any standard mechanism to new worlds. There's no principled way to add worlds from $\Omega$ to $W_i$. Effectively, they can never learn new concepts.
4. Any updates must be done on the entire space of things you consider possible [todo: *response: of course, this is all handled implicitly, so it's taken care of*].
5. While agents are free to merge multiple states of $\Omega$ into a single state in $W_i$, they cannot do the reverse: an agent cannot have a finer granularity than $\Omega$ for discerning events. This would . This also implies that agents are logically omniscient.

For all of these reasons, we take the view that probability should be thought of subjectively,

At the same time, starting with a set of random variables $\{X_i\}$, and setting $\Omega = \prod_i V_i$ to be every possible assignment of variables is also an abuse of the word "possible".

# 4 Equivalent Definitions and Variants

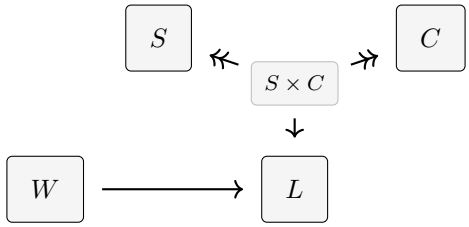
**Definition 4.1.** A *strict Probabilistic Dependency Graph* is a tuple $(N, L, \boldsymbol{\mu}, \mathcal{V})$ where

- $\mathcal{N} : \mathbf{Set}$ is a finite collection of nodes
- $\mathcal{L} \subseteq N \times N$ is a set of directed links between nodes
- $\mathcal{V} : \mathcal{N} \to \mathbf{MeasSet}$ is an $N$-indexed family of measurable sets, representing the values that a node can take
- $\boldsymbol{\mu} : ((A, B) : \mathcal{L}) \to \mathcal{V}(A) \to \Delta(\mathcal{V}(B))$ is a family of conditional probability distributions on $\mathcal{V}(B)$ indexed by the values of $A$ for every link $(A, B) \in \mathcal{L}$

The definition of $\boldsymbol{\mu}$ is probably more familiar than it looks. If every $\mathcal{V}(N)$ is finite with all subsets measurable, then $\boldsymbol{\mu}_{A,B}$ is just a conditional probability table, just as in a Bayesian Network. For those more familiar with stochastic processes, this is a stochastic matrix.

The definition $\boldsymbol{\mu}$ is slightly over-simplified if not everything is measurable. More generally if $\mathcal{V}(A) = (X, \mathcal{A})$ and $\mathcal{V}(B) = (Y, \mathcal{B})$, then for any link $L \in \mathcal{L}$, we're really referring to a function $\boldsymbol{\mu}_L : X \times \mathcal{B} \to [0, 1]$ such that $\boldsymbol{\mu}_L(x, -) : \mathcal{B} \to [0, 1]$ is a probability distribution and $\boldsymbol{\mu}_L(-, S)^{-1}(R) \in \mathcal{A}$ for any $S \in \mathcal{B}$, and measurable subset $R \subseteq [0, 1]$, technically making $\boldsymbol{\mu}_{A,B}$ a *Markov Kernel* from $A$ to $B$.

**Example 4.1.** In example 1.1, we displayed the arrow from $S$ and $C$ to $L$ as a directed hyper-edge. While we would like to maintain this intuition, it turns out that we can simplify our formalism by de-sugaring this picture into the following:




The double headed arrows are for degenerate conditional distributions, which are fully deterministic, but for now this is not terribly relevant. We can now present this PDG formally with the elements specified in definition 4.1; below we assume everything is measurable and omit this part of the formalism.

$$\begin{aligned}
\mathcal{N} &= \{S,\ C,\ L,\ W,\ S \times C\} \\
\mathcal{L} &= \{(S \times C, L), (W, L), (S \times C, S), (S \times C, C)\} \\
\mathcal{V} &= \begin{cases}
\mathcal{V}(S) &= \{\mathit{big}, \mathit{small}\} \\
\mathcal{V}(C) &= \{\mathit{rocky}, \mathit{gasseous}\} \\
\mathcal{V}(L) &= \{l, \neg l\} \\
\mathcal{V}(W) &= \{\mathit{none}, \mathit{some}, \mathit{mostly}\} \\
\mathcal{V}(S \times C) &= \mathcal{V}(S) \times \mathcal{V}(C)
\end{cases}
\end{aligned}$$

$$\boldsymbol{\mu} = \begin{cases}
\boldsymbol{\mu}[S \times C, L] = \begin{array}{c} \begin{matrix} l & \neg l \end{matrix} \\ \begin{bmatrix} .1 & .9 \\ .2 & .8 \\ .05 & 0.95 \\ 0.00001 & 0.99999 \end{bmatrix} \end{array} \begin{array}{l} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gasseous} \end{array} \qquad \boldsymbol{\mu}[W, L] = \begin{array}{c} \begin{matrix} l & \neg l \end{matrix} \\ \begin{bmatrix} 0 & 1 \\ .005 & .995 \\ .05 & 0.95 \end{bmatrix} \end{array} \begin{array}{l} \text{none} \\ \text{some} \\ \text{mostly} \end{array} \\
\boldsymbol{\mu}[S \times C, C] = \begin{array}{c} \begin{matrix} \text{rocky} & \text{gasseous} \end{matrix} \\ \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \end{array} \begin{array}{l} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gasseous} \end{array} \qquad \boldsymbol{\mu}[S \times C, S] = \begin{array}{c} \begin{matrix} \text{small} & \text{big} \end{matrix} \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \end{array} \begin{array}{l} \text{big, rocky} \\ \text{small,rocky} \\ \text{big, gasseous} \\ \text{small,gasseous} \end{array}
\end{cases}$$

△

---

This works pretty well for the two edges that we described before, but the structural overhead of the additional de-sugaring: the $\boldsymbol{\mu}[S \times C \to S]$ and $\boldsymbol{\mu}[S \times C \to C]$ tables, as well as the set $\mathcal{V}(S \times C)$ seem like they didn't need to be specified, and one might even feel that it would be a mistake to allow any other table. Some reasons for this design decision include:

- It is easier to prove things about graphs than directed hyper-graphs. Similarly, defining directed paths is a lot simpler.

- We can eliminate the clunkiness by fusing the model with an algebra, as in **??** — which will give us a lot more than modeling the hyper-edges directly.
- We will eventually also want to allow for the possibility of keeping only a relaxed, approximate representation of $\mathcal{V}$ and $\boldsymbol{\mu}$, and in particular, of the ones constructed logically in this way. By specifying them explicitly for now, we will have to do less work to regain manual control in **??**

## 4.1 Alternate Presentations

### 4.1.1 Random Variables

If $\mathcal{W} = (W, \mathcal{F}, \mu)$ is a measure space, and $\mathcal{X} = \{X_i : W \to \mathcal{V}(X_i)\}_{i \in I}$ is a collection of measurable random variables on $W$,[4] and $\mathcal{L} \subseteq I \times I$ is a collection of pairs of variables such that the agent [todo: *what is a way of phrasing this that doesn't sound like it's shoehorned in? $\mathcal{L}$ really can represent anything an agent knows. Any subjective conditional probability distribution $\mu'$ such that the only measurable subsets are "axis aligned", in that they involve queries on only one variable, can be represented by $\mathcal{L}$, and for other queries we can simply change variables.*], we call $(\mathcal{W}, \mathcal{X})$ an *ensemble*.

**Proposition 4.1.** *There is a natural correspondence between strict PDGs as defined in definition 4.1, and ensembles such that* [todo: *spell this out explicitly to avoid vague categorical intuition*] *. . . $\mu$'s are defined on same set and produce same values.*

*Proof. /outline:* On the one hand, $(\prod_{N \in \mathcal{N}} \mathcal{V}(N).\text{set}, \bigotimes_{N \in \mathcal{N}} \mathcal{V}(N).\text{algebra}, \boldsymbol{\mu})$ is a measure space, with $\{X_N = \pi_N : (\prod \mathcal{V}(N')) \to \mathcal{V}(N)\}_{N \in \mathcal{N}}$ a set of random variables

and on the other, $(I, \mathcal{L}, \mathcal{X}', \mu|_{\mathcal{L}})$ is a strict PDG. □

This is the technical underpinning of our flippant, noncommittal treatment of possible worlds: any time we are thinking in terms of random variables or probability distributions on a fixed set $W$, we can instead reduce

The complexity of the representation is $O(XV + LV^2)$, compared to $O(XW)$

## 4.2 Sub-stochastic Transitions

In this section we will see why we called the object in definition 4.1 a *strict* PDG. Sometimes an otherwise very useful variable might not apply in a small percentage of cases; in this case, we want a way of putting all of the extra probability mass in a "something else happened" bucket, giving us effectively a sub-stochastic matrix, or a a lower probability on singletons. For instance, the variable describing whether or not your answer is correct doesn't make sense if you weren't solving problems; the amount of money in your wallet doesn't make sense if you don't own one, and so forth. So now, when you're trying to predict the probability of certain amounts of money in your wallet, some of the probability mass needs to go into the "not applicable / something else" bucket.

There are several closely concepts that we will be able to employ with our framework after integrating them

1. Allowing random variables to be partial, rather than total functions of $W$.
2. Relaxing the requirement $\int_W \mu \mathrm{d}w = 1$ to $\int_W \mu \mathrm{d}w \leq 1$
3. Requiring that matrices be sub-stochastic, rather than stochastic
4. Replacing probabilities, with the more general class of lower probability measures.

[4] that is: $\mathcal{V}(X_i)$ is a measurable space, taking the form $(D, \mathcal{D})$, and $X_i : W \to D$ is a set function such that for every $B \in \mathcal{D}$, the set $X_i^{-1}(B) \in \mathcal{F}$

This generalization is useful, but our primary motivation for this generalization is so that we can represent implication, and thus a weakening of knowledge as it travels through our graph, in a way that is not just entropy (which might not be distinguishable from certain knowledge of a high entropy distribution otherwise).

At first glance, though, it might not be clear why this particular weakening buys us anything at all, because we can always just add the "something else" bucket $\bullet$, to $\mathcal{V}(X)$ for each $X$, and come up with a new strict PDG. A variable which might not make sense can always take a `null` value, and so now the set of possible is once again exhaustive. From the perspective of providing conditional distributions, however, this resolution poses a problem: our marginals now require us to estimate distributions from a null value— this is problematic, as a big part of the reason we've been using links to avoid assigning probabilities to everything. Suppose you are trying to represent the belief that you're happier when you get the right answer as a marginal link $L[\text{RightAns} \to ☺]$. We now need a distribution on happiness when you get the right answer, when you get the wrong answer, and also for when $\bullet$. Why might it not be applicable? Are you not solving problems because you're skiing? Because you've been injured? Maybe you are solving problems but there are multiple right answers? You can't just answer with a prior over happiness if you want to have consistent beliefs, because solving problems and happiness might be correlated. One *could* have such a thing but it seems unreasonable not to be able to express a belief about "does the right answer make you happy?" without also answering the much more difficult question, "how happy are you when 'the right answer' is not applicable to your current situation?"

To see how this increases our expressive power, suppose $A, B$ are binary variables (taking values $a, \bar{a}$ and $b, \bar{b}$ respectively). While we can easily easily represent $A = B$, $A = \neg B$ as stochastic matrices,

$$p(B \mid A) = \begin{matrix} b & \bar{b} \end{matrix} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{matrix} a \\ \bar{a} \end{matrix} \qquad \text{and} \qquad p(B \mid A) = \begin{matrix} b & \bar{b} \end{matrix} \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{matrix} a \\ \bar{a} \end{matrix}$$

we cannot (via stochastic matrices) represent an assertion that $A \Rightarrow B$ without also giving a distribution over $B$ given $\bar{a}$. One strategy is a uniform prior (used in [**logicalinduction**]), but this can easily lead to avoidable inconsistencies — perhaps for totally different reasons you have very good reason to believe that the true distribution of $B$ is true in 90% of cases; you don't want an arbitrary assumption of a prior competing with actual knowledge.

For this reason, we drop the requirement that our null element, $\bullet$, indexes a distribution in marginals. Below is an example of transition matrix $A \to B$ including the extra element. As mentioned, the last row is not something we are keeping track of.

$$\begin{matrix} b_0 & b_1 & \bullet \end{matrix} \\ \left[\begin{array}{ccc} .2 & .1 & 0.7 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ \hline .2 & .6 & 0 \end{array}\right] \begin{matrix} a_0 \\ a_1 \\ a_2 \\ \bullet \end{matrix}$$

Furthermore, because the final column is just whatever is necessary to make the rows sum to 1, we don't need to keep that either; as a result, it is sufficient to keep a smaller matrix without any $\bullet$-indices; the only price that we pay is that this matrix is *sub*-stochastic rather than stochastic: its row entries sum to at most 1, rather than exactly 1. Composition works just as before; the product of sub-stochastic matrices is sub-stochastic. A probability distribution alone, and by extension a standard Bayesian network cannot do this — because we require the look-up tables to exactly match all possible values, we can't drop any without totally giving up on any world which looks like that.

#### 4.2.1 Relation to Partial Functions of $W$

#### 4.2.2 Reduction to Lower Probability Measures

# 5 Semantics

These graphs admit multiple semantics. As discussed in section **??**, we think of Probabilistic Dependency Graphs as being a representation of knowledge in and of themselves, rather than a compression of something more fundamental such as a probability distribution. Still, we will find it useful to interpret them in various ways: doing so will make it possible to compare them more directly with existing graphical models, which one thinks of as really just being compressed distributions. In this section, we would like to highlight three important semantics.

## 5.1 As Sets of Distributions

If the focus is on under-constrained models, then just as a BN represents a distribution on joint space, a PDG might might thought of as representing the set of all distributions that marginalize out to it exactly.

**Definition 5.1.** If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG, let $[\![M]\!]_{\mathbf{Set}}$ be the set of distributions over the variables in $M$ consistent with $\boldsymbol{\mu}$ on every marginal. Formally,

$$[\![M]\!]_{\mathbf{Set}} := \left\{ \mu \in \Delta \left[ \prod_{N \in \mathcal{N}} \mathcal{V}(N) \right] \;\middle|\; \begin{array}{r} \mu(B = b \mid A = a) = \boldsymbol{\mu}[A, B](b \mid a) \\ \text{for all } A, B \in \mathcal{L},\ a \in \mathcal{V}(A), \text{ and } b \in \mathcal{V}(B) \end{array} \right\}$$

## 5.2 As Weighted Distributions

**Definition 5.2.** If $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is a PDG with joint worlds $W$, and $\ell : \Delta W \to \mathbb{D}$ is a scoring function, let $[\![M]\!]^{\ell}$ be the set of distributions that dominate. Formally,

$$[\![M]\!]^{\ell}(\mu) := \left\{ \mu \in \Delta \left[ \prod_{N \in \mathcal{N}} \mathcal{V}(N) \right] \;\middle|\; \begin{array}{r} \ell(\mu) \succeq \ell(\mu') \\ \text{for all } \mu' \in \Delta \prod \mathcal{V} \end{array} \right\}$$

## 5.3 As Distributions

To satisfy any lingering desire to compress all of the information to a single distribution, we also offer a way of interpreting a PDG as a single distribution.

**Definition 5.3.** If $M$ is a PDG, $[\![-]\!]_{\mathbf{S}}$ is a semantics, $(D, \preceq)$ is an ordered set, and $\ell : \Delta W_M \to (D, \leq)$ is a scoring function for probabilities, let *the upper $\preceq$-frontier under $\ell$*, denoted $[\![M]\!]^{\ell}_{\mathbf{S}}$, be the set of distributions that are not strictly dominated by any others. More formally,

$$[\![M]\!]^{\ell}_{\mathbf{S}} = \left\{ \mu \in [\![M]\!]_{\mathbf{S}} \;\middle|\; \forall \mu' \in [\![M]\!]_{\mathbf{S}}.\ \ell(\mu') \preceq \ell(\mu) \right\}$$

**Fact 5.1.**

$$[\![M]\!]^{\ell}_{\boldsymbol{Set}} = \left\{ \mu \in \Delta \prod \mathcal{V} : [\![M]\!]^{\ell}(\mu) \right\}$$

One particularly useful one for emulating Bayesians is the following one, maximizing entropy:

$$\left[\!\left[M\right]\!\right]_{\substack{\textbf{Max}\\\textbf{Ent}}} := \left[\!\left[M\right]\!\right]_{\textbf{Set}}^{-H(\cdot)}$$

This corresponds to a lexicographical ordering on distributions

Similarly, we can define

$$\left[\!\left[M\right]\!\right]_{\substack{\textbf{Max}\\\textbf{Ent}}} := \left[\!\left[M\right]\!\right]_{\textbf{Set}}^{-H(\cdot)}$$

# 6 Relations to Other Graphical Models

## 6.1 Bayesian Networks

## 6.2 Factor Graphs

# 7 Relations to Other Representations of Uncertainty

Probabilistic Dependency Graphs are far from the first formalism to provide a weaker notion of uncertainty than probability. Belief functions, inner measures, sets of probabilities, lower probabilities, weighted sets of probabilities, and plausibility measures have all been studied extensively in the past. One feature that each of these has in common is that they are under-specified, from the perspective of wanting probabilities for everything.

The natural question now becomes: to what do these under-constrained representations of belief correspond to under-constrained bits of a Probabilistic Dependency Graph?

## 7.1 Sets of Probability Measures

As we discuss in section 5.1

# 8 Using Inconsistency

Given a distribution $\mu$, and a

**Definition 8.1** (consistency)**.** A Probabilistic Dependency Graph $M = (\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})$ is *consistent* if there exists some joint probability Pr on all of the variables, or equivalently, if $[\![M]\!]_{\textbf{Set}} \neq \emptyset$

$$\zeta(\mathcal{N}, \mathcal{L}, \mathcal{V}, \boldsymbol{\mu})) := \inf_{p \in \Delta(W^{\mathcal{V}})} \sum_{L[A,B] \in \mathcal{L}} \mathop{\mathbb{E}}_{a \sim p(A)} \left[ D_{\mathrm{KL}}\Big( L(a) \,\Big\|\, p(B \mid a) \Big) \right] \tag{1}$$

# 9 Algorithms

## 9.1 Belief Propagation

# 10 Discussion

## 10.1 Inconsistency

Inconsistency is bad. Believing a logically inconsistent formula can lead you to arbitrarily bad conclusions, having an infeasible set of constraints makes all answers you could give wrong, and having inconsistent preferences can lose you infinite money. We don't want to build inconsistent systems or agents with incoherent views of the world, and so, where possible, we design them so they cannot possibly be broken in this way. Suppose, for example, that we are trying to represent some quantity that must be a point on the unit circle. We could do it with an $x$ and $y$ coordinate, but this could be problematic because $x^2 + y^2$ might not be $1$ — it would be safer and harder to go awry if we parameterize it by an angle $\theta \in [0, 2\pi)$ instead. In the absence of performance benefits (like needing to regularly use the $y$-coordinate and not wanting to compute a sine), why would we take the first approach, introducing a potentially complex data-invariant, when we could avoid it?

This line of thought, though common and defensible, is flawed if we are not perfectly confident in the design of both our system and the ways it can interact with the outside world. Using similar logic, we might ask ourselves: Why ask programmers for type annotations when all instructions are operationally well-defined at run-time? Why use extra training data if there's already enough there to specify a function? Why estimate a quantity in two ways when they will yield different answers? Why repeat and rephrase your ideas when this could make you contradict yourself? Why write test cases when they could fail and make the project inconsistent? Why conduct an experiment if it could just end up contradicting your current knowledge?

These questions may seem silly, but there is a satisfying information theoretic answer to all of them: redundancy, though costly, is the primary tool that we use to combat the possibility of being wrong. Maintaining data invariants can be expensive but provides diagnostic information; in the example above, settings of $x$ and $y$ that don't lie on the unit circle provide diagnostic information that something has gone wrong. In many cases, it is also possible to paper over problems by forcibly re-instating local data invariants: for instance, we could re-normalize any values of $x$ and $y$ (so long as $xy \neq 0$; we can chose an arbitrary point otherwise) at every step. While this would reduce inconsistency, it also hides red flags.

Using a Bayesian Network to represent a probability distribution is like representing a circle with $\theta \in [0, 2\pi)$. By construction, the result must be a distribution, and nothing can possibly go wrong so long as we can always decide on exactly one distribution which is sufficient for our purposes.

The process of mechanistically forcing invariants is homologous to the standard practice for factor graphs: practitioners will often just assume that the density it defines is normalizable, and either forcibly re-normalize or cleverly avoid computing the normalization constant while still assuming that one exists; behavior is usually left unspecified in the unlikely event that it is not defined or zero.