Probabilistic Dependency Graphs

Oliver Richardson, Joseph Y. Halpern
Computer Science Department
Cornell University
Ithaca, NY 14853
{oli, halpern}@cs.cornell.edu

Abstract

We introduce Probabilistic Dependency Graphs (PDGs), a new class of directed graphical models. PDGs can capture inconsistent beliefs in a natural way and are more modular than Bayesian Networks (BNs), in that they make it easier to incorporate new information and restructure the representation. We show by example how PDGs are an especially natural modeling tool. We provide three semantics for PDGs, each of which can be derived from a scoring function (on joint distributions over the variables in the network) that can be viewed as representing a distribution's incompatibility with the PDG. For the PDG corresponding to a BN, this function is uniquely minimized by the distribution the BN represents, showing that PDG semantics extend BN semantics. We show further that factor graphs and their exponential families can also be faithfully represented as PDGs, while there are significant barriers to modeling a PDG with a factor graph.

1 Introduction

In this paper we introduce yet another graphical tool for modeling beliefs, *Probabilistic Dependency Graphs* (PDGs). There are already many such models in the literature, including Bayesian networks (BNs) and factor graphs. (For an overview, see (Koller and Friedman 2009).) Why does the world need one more?

Our original motivation for introducing PDGs was to be able capture inconsistency. We want to be able to model the process of resolving inconsistency; to do so, we have to model the inconsistency itself. But our approach to modeling inconsistency has many other advantages. In particular, PDGs are significantly more modular than other directed graphical models: operations like restriction and union that are easily done with PDGs are difficult or impossible to do with other representations. The following examples motivate PDGs and illustrate some of their advantages.

Example 1. Grok is visiting a neighboring district. From prior reading, she thinks it likely (probability .95) that guns are illegal here. Some brief conversations with locals lead her to believe that, with probility .1, the law prohibits floomps.

The obvious way to represent this as a BN is to use two random variables F and G (respectively taking values $\{f, \overline{f}\}$ and $g, \overline{g}$), indicating whether floomps and guns are prohibited. The semantics of a BN offer her two choices: either assume that F and G to be independent and give (unconditional) probabilities of F and G, or choose a direction of dependency, and give one of the two unconditional probabilities and a conditional probability distribution. As there is no reason to choose either direction of dependence, the natural choice is to assume independence, giving her the BN on the left of Figure 1.

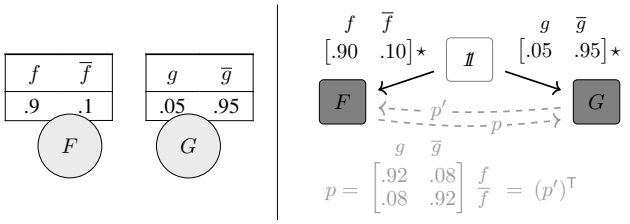


Figure 1: A BN (left) and corresponding PDG (right), which can include more cpds; p or p' make it inconsistent.

A traumatic experience a few hours later leaves Grok believing that "floomp" is likely (probability .92) to be another word for gun. Let $p(G \mid F)$ be the *c*onditional *p*robability *d*istribution (cpd) that describes the belief that if floomps are legal (resp., illegal), then with probability .92, guns are as well, and $p'(F \mid G)$ be the reverse. Starting with p, Grok's first instinct is to simply incorporate the conditional information by adding F as a parent of G, and then associating the cpd p with G. But then what should she do with the original probability she had for G? Should she just discard it? It is easy to check that there is no joint distribution that is consistent with both the two original priors on F and G and also p. So if she is to represent the information with a BN, which always represents a consistent distribution, she must resolve the inconsistency.

However, sorting this out immediately may not be ideal. For instance, if the inconsistency arises from a conflation between two definitions of "gun", a resolution will have destroyed the original cpds. A better use of computation may be to notice the inconsistency and look up the actual law.

By way of contrast, consider the corresponding PDG. In a PDG, the cpds are attached to edges, rather than nodes of the graph. In order to represent unconditional probabilities,

we introduce a *unit variable* $\mathbb{1}$ which takes only one value, denoted $\star$. This leads Grok to the PDG depicted in Figure 1, where the edges from $\mathbb{1}$ to F and G are associated with the unconditional probabilities of F and G, and the edges between F and G are associated with p and p'.

The original state of knowledge consists of all three nodes and the two solid edges from $\mathbb{1}$. This is like Bayes Net that we considered above, except that we no longer explicitly take F and G to be independent; we merely record the constraints imposed by the given probabilities.

The key point is that we can incorporate the new information into our original representation (the graph in Figure 1 without the edge from F to G) simply by adding the edge from F to G and the associated cpd p (the new infromation is shown in blue). Doing so does not change the meaning of the original edges. Unlike a Bayesian update, the operation is even reversible: all we need to do recover our original belief state is delete the new edge, making it possible to mull over and then reject an observation. □

The ability of PDGs to model inconsistency, as illustrated in Example 1, appears to have come at a significant cost. We seem to have lost a key benefit of BNs: the ease with which they can capture (conditional) independencies, which, as Pearl (1988) has argued forcefully, are omnipresent.

Example 2 (emulating a BN)**.** We now consider the classic (quantitative) Bayesian network $\mathcal{B}$, which has four binary variables indicating whether a person (C) develops cancer, (S) smokes, (SH) is exposed to second-hand smoke, and (PS) has parents who smoke, presented graphically in Figure 2(a). We now walk through what is required to represent $\mathcal{B}$ as a PDG, which we call $\mathcal{M}_{\mathcal{B}}$, shown as the solid nodes and edges in Figure 2(b).

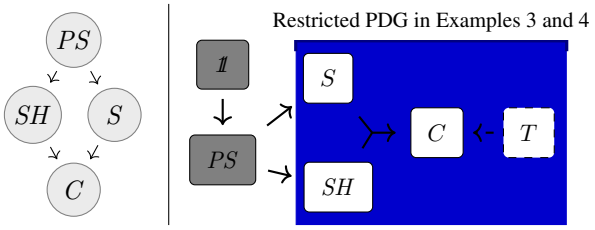


Figure 2: (a) The Bayesian Network $\mathcal{B}$ in Example 2 (left), and (b) $\mathcal{M}_{\mathcal{B}}$, its corresponding PDG (right). The shaded box indicates a restriction of $\mathcal{M}_{\mathcal{B}}$ to only the nodes and edges it contains, and the dashed node T and its arrow to C can be added in the PDG, without taking into account S and SH.

We start with the nodes corresponding to the variables in $\mathcal{B}$, together with the special node $\mathbb{1}$ from Example 1; we add an edge from $\mathbb{1}$ to PS, to which we associate the unconditional probability given by the cpd for PS in $\mathcal{B}$. We can also reuse the cpds for S and SH, assigning them, respectively, to the edges $PS \to S$ and $PS \to SH$ in $\mathcal{M}_{\mathcal{B}}$. There are two remaining problems: (1) modeling the remaining table in $\mathcal{B}$, which corresponds to the conditional probability of C given S and SH; and (2) recovering the additional conditional independence assumptions in the BN.

For (1), we cannot just add the edges $S \to C$ and $SH \to C$ that are present in $\mathcal{B}$. As we saw in Example 1, this would mean supplying two *separate* tables, one indicating the probability of C given S, and the other indicating the probability of C given SH. We would lose significant information that is present in $\mathcal{B}$ about how C depends jointly on S and SH. To distinguish the joint dependence on S and SH, for now, we draw an edge with two tails—a (directed) *hyperedge*—that completes the diagram in Figure 2(b). With regard to (2), there are many distributions consistent with the conditional marginal probabilities in the cpds, and the independences presumed by $\mathcal{B}$ need not hold for them. Rather than trying to distinguish between them with additional constraints, we develop a a scoring-function semantics for PDGs which is in this case uniquely minimized by the distribution specified by $\mathcal{B}$ (Theorem 4.1). This allows us to recover the semantics of Bayesian networks without requiring the independencies that they assume.

Next suppose that we get information beyond that captured by the original BN. Specifically, we read a thorough empirical study demonstrating that people who use tanning beds have a 10% incidence of cancer, compared with 1% in the control group (call the cpd for this p); we would like to add this information to $\mathcal{B}$. The first step is clearly to add a new node labeled T, for "tanning bed use". But simply making T a parent of C (as clearly seems appropriate, given that the incidence of cancer depends on tanning bed use) requires a substantial expansion of the cpd; in particular, it requires us to make assumptions about the interactions between tanning beds and smoking. The corresponding PDG, $\mathcal{M}_{\mathcal{B}}$, on the other hand, has no trouble: We can simply add the node T with an edge to C that is associated with $\mathbf{p}$. But note that doing this makes it possible for our knowledge to be inconsistent. To take a simple example, if the distribution on C given S and H encoded in the original cpd was always deterministically "has cancer" for every possible value of S and H, but the distribution according to the new cpd from T was deterministically "no cancer", the resulting PDG would be inconsistent. □

We have seen that we can easily add information to PDGs; removing information is equally painless.

Example 3 (restriction)**.** After the Communist party came to power, children were raised communally, and so parents' smoking habits no longer had any impact on them. Grok is reading her favorite book on graphical models, and she realizes that while the node PS in Figure 2(a) has lost its usefulness, and nodes S and SH no longer ought to have PS as a parent, the other half of the diagram—that is, the node C and its dependence on S and SH—should apply as before. Grok has identified two obstacles to modeling deletion of information from a BN by simply deleting nodes and their associated cpds. First, this restricted model is technically no longer a BN (which in this case would require unconditional distributions on S and SH), but rather a *conditional* BN (Koller and Friedman 2009), which allows for these nodes to be marked as observations; observation nodes do not have associated beliefs. Second, even regarded as a conditional BN, the result of deleting a node may introduce *new* indepen-

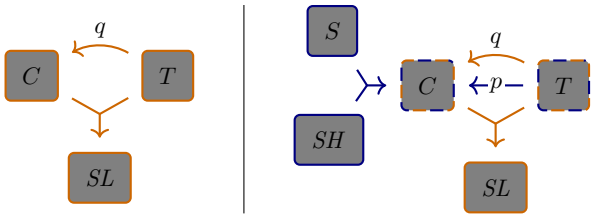


Figure 3: Grok's prior (left) and combined (right) knowledge.

dence information, incompatible with the original BN. For instance, by deleting the node B in a chain $A \to B \to C$, one concludes that A and C are independent, a conclusion incompatible with the original BN containing all three nodes. PDGs do not suffer from either problem. We can easily delete the nodes labeled 1 and PS in Figure 2(b) to get the restricted PDG shown in the figure, which captures Grok's updated information. The resulting PDG has no edges leading to S or SH, and hence no distributions specified on them; no special modeling distinction between observation nodes and other nodes are required. Because PDGs do not directly make independence assumptions, the information in this fragment is truly a subset of the information in the whole PDG. □

Being able to form a well-behaved local picture and restrict knowledge is useful, but an even more compelling reason to use PDGs is their ability to aggregate information.

Example 4. Grok dreams of becoming Supreme Leader (SL), and has come up with a plan. She has noticed that people who use tanning beds have significantly more power than those who don't. Unfortunately, her mom has always told her that tanning beds cause cancer; specifically, that 15% of people who use tanning beds get it, compared to the baseline of 2%. Call this cpd q. Grok thinks people will make fun of her if she uses a tanning bed and gets cancer, making becoming Supreme Leader impossible. This mental state is depicted as a PDG on the left of Figure 3.

Grok is reading about graphical models because she vaguely remembers that the variables in Example 2 match the ones she already knows about. When she finishes reading the statistics on smoking and the original study on tanning beds (associated to a cpd $\mathbf{p}$ in Example 2), but before she has time to reflect, we can represent her (conflicted) knowledge state as the union of the two graphs, depicted graphically on the right of Figure 3.

The union of the two PDGs, even with overlapping nodes, is still a PDG. This is not the case in general for BNs. Note that the PDG that Grok used to represent her two different sources of information (the mother's wisdom and the study) regarding the distribution of C is a *multigraph*: there are two edges from T to C, with inconsistent information. Had we not allowed multigraphs, we would have needed to choose between the two edges, or represent the information some other (arguably less natural) way. As we are already allowing inconsistency, merely recording both is much more in keeping with the way we have handled other types of uncertainty. □

Not all inconsistencies are equally egregious. For example, even though the cpds p and q are different, they are numerically close, so, intuitively, the PDG on the right in Figure 3 is not very inconsistent. Making this precise is the focus of Section 3.2.

These examples give a taste of the power of PDGs. In the coming sections, we formalize PDGs and relate them to other approaches.

2 Syntax

We now provide formal definitions for PDGs. Although it is possible to formalize PDGS with hyperedges directly, we opt for a different approach here, in which PDGs have only regular edges, and hyperedges are captured using a simple construction that involves adding an extra node.

Definition 2.1. A *Probabilistic Dependency Graph* is a tuple $\mathcal{M} = (\mathcal{N}, \mathcal{E}, \mathcal{V}, \mathbf{p}, \alpha, \beta)$, where

- $\mathcal{N}$ is a finite set of nodes, corresponding to variables;
- $\mathcal{E}$ is a set of labeled edges $\{X \xrightarrow{L} Y\}$, each with a source X and target Y in $\mathcal{N}$;
- $\mathcal{V}$ associates each variable $N \in \mathcal{N}$ with a set $\mathcal{V}(N)$ of values that the variable N can take;
- $\mathbf{p}$ associates to each edge $X \xrightarrow{L} Y \in \mathcal{E}$ a distribution $\mathbf{p}_L(x)$ on Y for each $x \in \mathcal{V}(X)$;
- α associates to each edge $X \xrightarrow{L} Y$ a non-negative number α_L which, roughly speaking, is the modeler's confidence in the functional dependence of Y on X implicit in L;
- β associates to each edge L a positive real number β_L, the modeler's subjective confidence in the reliability of $\mathbf{p}_L$.

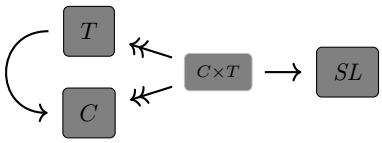
Note that we allow multiple edges in $\mathcal{E}$ with the same source and target; thus $(\mathcal{N}, \mathcal{E})$ is a multigraph. We occasionally write a PDG as $\mathcal{M} = (\mathcal{G}, \mathbf{p}, \alpha, \beta)$, where $\mathcal{G} = (\mathcal{N}, \mathcal{E}, \mathcal{V})$, and abuse terminology by referring to $\mathcal{G}$ as a multigraph. We refer to $\mathcal{N} = (\mathcal{G}, \mathbf{p})$ as an *unweighted* PDG, and give it semantics as though it were the (weighted) PDG $(\mathcal{G}, \mathbf{p}, \mathbf{1}, \mathbf{1})$, where $\mathbf{1}$ is the constant function (i.e., so that $\alpha_L = \beta_L = 1$ for all L). In this paper, with the exception of Section 4.3, we implicitly take $\alpha = \mathbf{1}$ and omit α, writing $\mathcal{M} = (\mathcal{G}, \mathbf{p}, \beta)$.[1] □

If $\mathcal{M}$ is a PDG, we reserve the names $\mathcal{N}^{\mathcal{M}}, \mathcal{E}^{\mathcal{M}}, \ldots$, for the components of $\mathcal{M}$, so that we may reference one without naming them all explicitly. We write $\mathcal{V}(S)$ for the set of possible joint settings of a set S of variables, and write $\mathcal{V}(\mathcal{M})$ for all settings of the variables in $\mathcal{N}^{\mathcal{M}}$; we refer to these settings as "worlds". While the definition above is sufficient to represent the class of all legal PDGs, we often use two additional bits of syntax to indicate common constraints: the special variable $\mathbb{1}$ such that $\mathcal{V}(\mathbb{1}) = \{\star\}$ from Examples 1 and 2, and double-headed arrows, $A \twoheadrightarrow B$, which visually indicate that the corresponding cpd is degenerate, effectively representing a deterministic function $f : \mathcal{V}(A) \to \mathcal{V}(B)$.

Construction 2.2. We can now explain how we capture the multi-tailed edges that were used in Examples 2 to 4. That notation can be viewed as shorthand for the graph that results by adding a new node at the junction representing the joint value of the nodes at the tails, with projections going back. For instance, the diagram displaying Grok's prior knowledge

[1] The full paper gives results for arbitrary α.

in Example 4, on the left of Figure 3 is really shorthand for the following PDG, where where we insert a node labeled $C \times T$ at the junction:



As the notation suggests, $\mathcal{V}(C \times T) = \mathcal{V}(C) \times \mathcal{V}(T)$. For any joint setting $(c, t) \in \mathcal{V}(C \times T)$ of both variables, the cpd for the edge from $C \times T$ to C gives probability 1 to c; similarly, the cpd for the edge from $C \times T$ to T gives probability 1 to t. □

3 Semantics

Although the meaning of an individual cpd is clear, we have not yet given PDGs a "global" semantics. We discuss three related approaches to doing so. The first is the simplest: we associate with a PDG the set of distributions that are consistent with it. This set will be empty if the PDG is inconsistent. The second approach associates a PDG with a scoring function, indicating the fit of an arbitrary distribution μ, and can be thought of as a *weighted* set of distributions (Halpern and Leung 2015). This approach allows us to distinguish inconsistent PDGs, while the first approach does not. The third approach chooses the distributions with the best score, typically associating with a PDG a unique distribution.

3.1 PDGs As Sets Of Distributions

We have been thinking of a PDG as a collection of constraints on distributions, specified by matching cpds. From this perspective, it is natural to consider the set of all distributions that are consistent with the constraints.

Definition 3.1. If $\mathcal{M}$ is a PDG (weighted or unweighted) with edges $\mathcal{E}$ and cpds $\mathbf{p}$, let $[\![\mathcal{M}]\!]_{\text{sd}}$ be the *s*et of *d*istributions over the variables in $\mathcal{M}$ whose conditional marginals are exactly those given by $\mathbf{p}$. That is, $\mu \in [\![\mathcal{M}]\!]_{\text{sd}}$ iff, for all edges $L \in \mathcal{E}$ from X to Y, $x \in \mathcal{V}(X)$, and $y \in \mathcal{V}(Y)$, we have that $\mu(Y = y \mid X = x) = \mathbf{p}_L(x)$. $\mathcal{M}$ is *inconsistent* if $[\![\mathcal{M}]\!]_{\text{sd}} = \emptyset$, and *consistent* otherwise. □

Note that $[\![\mathcal{M}]\!]_{\text{sd}}$ is independent of the weights α and β.

3.2 PDGs As Distribution Scoring Functions

We now generalize the previous semantics by viewing a PDG $\mathcal{M}$ as a *scoring function* that, given an arbitrary distribution μ on $\mathcal{V}(\mathcal{M})$, returns a real-valued score indicating how well μ fits $\mathcal{M}$. Distributions with the lowest (best) scores are those that most closely match the cpds in $\mathcal{M}$, and contain the fewest unspecified correlations.

We start with the first component of the score, which assigns higher scores to distributions that require a larger perturbation in order to be consistent with $\mathcal{M}$. We measure the magnitude of this perturbation with relative entropy. In particular, for an edge $X \xrightarrow{L} Y$ and $x \in \mathcal{V}(X)$, we measure the relative entropy from $\mathbf{p}_L(x)$ to $\mu(Y = \cdot \mid X = x)$, and take the expectation over μ_X (that is, the marginal of μ on X). We then sum over all the edges L in the PDG, weighted by their reliability.

Definition 3.2. For a PDG $\mathcal{M}$, the *incompatibility* of a a joint distribution μ over $\mathcal{V}(\mathcal{M})$, is given by

$$Inc_{\mathcal{M}}(\mu) := \sum_{X \xrightarrow{L} Y \in \mathcal{E}^m} \beta_L^{\mathcal{M}} \mathop{\mathbb{E}}_{x \sim \mu_X} \left[\boldsymbol{D}\Big(\mu(Y \mid X = x) \;\Big\|\; \mathbf{p}_L^{\mathcal{M}}(x)\Big)\right],$$

where $\boldsymbol{D}(\mu \parallel \nu) = \sum_{w \in \text{Supp}(\mu)} \mu(w) \log \frac{\mu(w)}{\nu(w)}$ is the relative entropy from ν to μ. □

$[\![\mathcal{M}]\!]_{\text{sd}}$ and $Inc_{\mathcal{M}}$ distinguish between distributions based on their compatibility with $\mathcal{M}$, but even among distributions that match the marginals, some more closely match the qualitative structure of the graph than others. We think of each edge $X \xrightarrow{L} Y$ as representing a qualitative claim (with confidence α_L) that the value of Y can be computed from X alone. To formalize this, we require only the multigraph $\mathcal{G}^{\mathcal{M}}$.

Given a multigraph G and distribution μ on its variables, contrast the amount of information required to

(a) directly describe a joint outcome $\mathbf{w}$ drawn from μ, and

(b) separately specify, for each edge $X \xrightarrow{L} Y$, the value $\mathbf{w}_Y$ (of Y in world $\mathbf{w}$) given the value $\mathbf{w}_X$, in expectation.

If (a) = (b), a specification of (b) has exactly the same length as a full desciption of the world. If (b) > (a), then there are correlations in μ that allow for a more compact representation than G provides. The larger the difference, the more information is needed to determine targets Y beyond the conditional probabilities associated with the edges $X \to Y$ leading to Y (which according to G should be sufficient to compute them), and the poorer the qualitative fit of μ to G. Finally, if (a) > (b), then μ requires additional information to specify, beyond what is necessary to determine outcomes of the marginals selected by G.

Definition 3.3. For a multigraph $G = (\mathcal{N}, \mathcal{E}, \mathcal{V})$ over a set $\mathcal{N}$ of variables, define the *G-information deficiency* of distribution μ, denoted $IDef_G(\mu)$, by considering the difference between (a) and (b), where we measure the amount of information needed for a description using entropy:

$$IDef_G(\mu) := \sum_{(X,Y) \in \mathcal{E}} \mathrm{H}_\mu(Y \mid X) - \mathrm{H}(\mu). \qquad (1)$$

(Recall that $H_\mu(Y \mid X)$, the (μ-)*conditional entropy of Y given X*, is defined as $-\sum_{x,y \in \mathcal{V}(X,Y)} \mu(x, y) \log \mu(y \mid x)$.) For a PDG $\mathcal{M}$, we take $IDef_{\mathcal{M}} = IDef_{\mathcal{G}^{\mathcal{M}}}$. □

We illustrate $IDef_{\mathcal{M}}$ with some simple examples. Suppose that $\mathcal{M}$ has two nodes, X and Y. If $\mathcal{M}$ has no edges, the $IDef_{\mathcal{M}}(\mu) = -\mathrm{H}(\mu)$. There is no information required to specify, for each edge in $\mathcal{M}$ from X to Y, the value $\mathbf{w}_Y$ given $\mathbf{w}_X$, since there are no edges. Since we view smaller numbers as representing a better fit, $IDef_{\mathcal{M}}$ in this case will prefer the distribution that maximizes entropy. If $\mathcal{M}$ has one edge from X to Y, then since $\mathrm{H}(\mu) = \mathrm{H}_\mu(Y \mid X) + \mathrm{H}_\mu(X)$ by the well known *entropy chain rule* (MacKay 2003), $IDef_{\mathcal{M}}(\mu) = -\mathrm{H}_\mu(X)$. Intuitively, while knowing the conditional probability $\mu(Y \mid X)$ is helpful, to completely specify μ we also need $\mu(X)$. Thus, in this case, $IDef_{\mathcal{M}}$ prefers distributions that maximize the entropy of the marginal on X. If $\mathcal{M}$ has sufficiently many parallel edges from X to Y

and $\mathrm{H}_\mu(Y \mid X) > 0$ (so that Y is not totally determined by X) then we have $IDef_{\mathcal{M}}(\mu) > 0$, because the redundant edges add no information, but there is still a cost to specifying them. In this case, $IDef_{\mathcal{M}}$ prefers distributions that make Y a deterministic function of X will maximizing the entropy of the marginal on X. Finally, if $\mathcal{M}$ has an edge from X to Y and another from Y to X, then a distribution μ minimizes $IDef_{\mathcal{M}}$ when X and Y vary together (so that $\mathrm{H}_\mu(Y \mid X) = \mathrm{H}_\mu(X \mid Y) = 0$) while maximizing $\mathrm{H}(\mu)$, for example, by taking $\mu(0,0) = \mu(1,1) = 1/2$.

$Inc_{\mathcal{M}}(\mu)$ and $IDef_{\mathcal{M}}(\mu)$ give us two measures of compatibility between $\mathcal{M}$ and a distribution μ. We take the score of interest to be their sum, with the tradeoff controlled by a parameter $\gamma \geq 0$:

$$[\![\mathcal{M}]\!]_\gamma(\mu) := Inc_{\mathcal{M}}(\mu) + \gamma IDef_{\mathcal{M}}(\mu) \tag{2}$$

The following just makes precise that the scoring semantics generalizes the first semantics.

Proposition 3.1. *$[\![\mathcal{M}]\!]_{sd} = \{\mu : [\![\mathcal{M}]\!]_0(\mu) = 0\}$ for all $\mathcal{M}$.*

While we focus on this particular scoring function in the paper, in part because it has deep connections to the free energy of a factor graph (Koller and Friedman 2009), other scoring functions may well end up being of interest.

3.3 PDGs As Unique Distributions

Finally, we provide an interpretation of a PDG as a probability distribution. Before we provide this semantics, we stress that this distribution does *not* capture all of the important information in the PDG—for example, a PDG can represent inconsistent knowledge states. Still, by giving a distribution, we enable comparisons with other graphical models, and show that PDGs are a surprisingly flexible tool for specifying distributions. The idea is to select the distributions with the best score. We thus define

$$[\![\mathcal{M}]\!]^*_\gamma = \operatorname*{arg\,min}_{\mu \in \Delta\mathcal{V}(\mathcal{M})} [\![\mathcal{M}]\!]_\gamma(\mu). \tag{3}$$

In general, $[\![\mathcal{M}]\!]^*_\gamma$ does not give a unique distribution. But if γ is sufficiently small, then it does:

Proposition 3.2. *If $\mathcal{M}$ is a PDG and $0 < \gamma \leq \min_L \beta^{\mathcal{M}}_L$, then $[\![\mathcal{M}]\!]^*_\gamma$ is a singleton.*

In this paper, we are interested in the case where γ is small; this amounts to emphasizing the accuracy of the probability distribution as a description of probabilistic information, rather than the graphical structure of the PDG. This motivates us to consider what happens as γ goes to 0. If S_γ is a set of probability distributions for all $\gamma \in [0,1]$, we define $\lim_{\gamma\to 0} S_\gamma$ to consist of all distributions μ such that there is a sequence $(\gamma_i, \mu_i)_{i\in\mathbb{N}}$ with $\gamma_i \to 0$ and $\mu_i \to \mu$ such that $\mu_i \in S_{\gamma_i}$ for all i. It can be further shown that

Proposition 3.3. *For all $\mathcal{M}$, $\lim_{\gamma\to 0}[\![\mathcal{M}]\!]^*_\gamma$ is a singleton.*

Let $[\![\mathcal{M}]\!]^*$ be the unique element of $\lim_{\gamma\to 0}[\![\mathcal{M}]\!]^*_\gamma$. The semantics has an important property:

Proposition 3.4. *$[\![\mathcal{M}]\!]^* \in [\![\mathcal{M}]\!]^*_0$, so if $\mathcal{M}$ is consistent, then $[\![\mathcal{M}]\!]^* \in [\![\mathcal{M}]\!]_{sd}$.*

4 Relationships to Other Graphical Models

We start by relating PDGs to two of the most popular graphical models: BNs and factor graphs. PDGs are strictly more general than BNs, and can emulate factor graphs for a particular value of γ.

4.1 Bayesian Networks

Construction 2.2 can be generalized to convert arbitrary Bayesian Networks into PDGs. Given a BN $\mathcal{B}$ and a positive confidence β_X for the cpd of each variable X of $\mathcal{B}$, let $\mathcal{M}_{\mathcal{B},\beta}$ be the PDG comprising the cpds of $\mathcal{B}$ in this way; we defer the straightforward formal details to the full paper.

Theorem 4.1. *If $\mathcal{B}$ is a Bayesian network and $\Pr_{\mathcal{B}}$ is the distribution it specifies, then for all $\gamma > 0$ and all vectors β such that $\beta_L > 0$ for all edges L, $[\![\mathcal{M}_{\mathcal{B},\beta}]\!]^*_\gamma = \{\Pr_{\mathcal{B}}\}$, and thus $[\![\mathcal{M}_{\mathcal{B},\beta}]\!]^* = \Pr_{\mathcal{B}}$.*

Theorem 4.1 is quite robust to parameter choices: it holds for every weight vector β and all $\gamma > 0$. However, it does lean heavily on our assumption that $\alpha = \mathbf{1}$, making it our only result that does not have a natural analog for general α.

4.2 Factor Graphs

Factor graphs (Kschischang, Frey, and Loeliger 2001), like PDGs, generalize BNs. In this section, we consider the relationship between factor graphs (FGs) and PDGs.

Definition 4.1. A *factor graph* Φ is a set of random variables $\mathcal{X} = \{X_i\}$ and *factors* $\{\phi_J \colon \mathcal{V}(X_J) \to \mathbb{R}_{\geq 0}\}_{J\in\mathcal{J}}$, where $X_J \subseteq \mathcal{X}$. More precisely, each factor ϕ_J is associated with a subset $X_J \subseteq \mathcal{X}$ of variables, and maps joint settings of X_J to non-negative real numbers. Φ specifies a distribution

$$\Pr_\Phi(\vec{x}) = \frac{1}{Z_\Phi} \prod_{J\in\mathcal{J}} \phi_J(\vec{x}_J),$$

where $\vec{x}$ is a joint setting of all of the variables, $\vec{x}_J$ is the restriction of $\vec{x}$ to only the variables X_J, and Z_Φ is the constant required to normalize the distribution. □

The cpds of a PDG naturally constitute a collection of factors, so it natural to wonder how the semantics of a PDG compares to simply treating the cpds as factors in a factor graph. To answer this, we start by making the translation precise.

Definition 4.2 (unweighted PDG to factor graph)**.** If $\mathcal{N} = (\mathcal{G}, \mathbf{p})$ is an unweighted PDG, define the associated FG $\Phi_{\mathcal{N}}$ on the variables $(\mathcal{N}, \mathcal{V})$ by taking $\mathcal{J}$ to be the set of edges, and for an edge L from Z to Y, taking $X_L = \{Z, Y\}$, and $\phi_L(z,y)$ to be $\mathbf{p}^{\mathcal{M}}_L(y \mid z)$ (i.e., $(\mathbf{p}^{\mathcal{M}}_L(z))(y)$). □

It turns out we can also do the reverse. Using essentially the same idea as in Construction 2.2, we can encode a factor graph as an assertion about the unconditional probability distribution over the variables associated to each factor.

Definition 4.3 (factor graph to unweighted PDG)**.** For a FG Φ, let $\mathcal{N}_\Phi$ be the unweighted PDG consisting of

- the variables in Φ together with $\mathbb{1}$ and a variable $X_J := \prod_{j\in J} X_j$ for every factor $J \in \mathcal{J}$, and
- edges $\mathbb{1} \to X_J$ for each J and $X_J \twoheadrightarrow X_j$ for each $X_j \in \mathbf{X}_J$,

where the edges $X_J \twoheadrightarrow X_j$ are associated with the appropriate projections, and each $\mathbb{1} \to X_J$ is associated with the unconditional joint distribution on X_J obtained by normalizing ϕ_J. The process is illustrated in Figure 4. □

PDGs are directed graphs, while factors graphs are undirected. The map from PDGs to factor graphs thus loses some important structure. As shown in Figure 4, this mapping can change the graphical structure significantly. Nevertheless,

Theorem 4.2. $\Pr_\Phi = [\![\mathcal{N}_\Phi]\!]_1^*$ *for all factor graphs* Φ.[2]

Theorem 4.3. $[\![\mathcal{N}]\!]_1^* = \Pr_{\Phi_\mathcal{N}}$ *for all unweighted PDGs* $\mathcal{N}$.

The correspondence hinges on the fact that we take $\gamma = 1$, so that *Inc* and *IDef* are weighted equally. Because the user of a PDG gets to choose γ, the fact that the translation from factor graphs to PDGs preserves semantics only for $\gamma = 1$ poses no problem. Conversely, the fact that the reverse correspondence requires $\gamma = 1$ suggests that factor graphs are less flexible than PDGs.

What about weighted PDGs $(\mathcal{G}, \mathbf{p}, \beta)$ where $\beta \neq \mathbf{1}$? There is also a standard notion of weighted factor graph, but as long as we stick with our convention of taking $\alpha = \mathbf{1}$, we cannot relate them to weighted PDGs. As we are about to see, once we drop this convention, we can do much more.

4.3 Factored Exponential Families

A *weighted factor graph (WFG)* Ψ is a pair (Φ, θ) consisting of a factor graph Φ together with a vector of non-negative weights $\{\theta_J\}_{J \in \mathcal{J}}$. Ψ specifies a canonical scoring function

$$\mathit{GFE}_\Psi(\mu) := \mathop{\mathbb{E}}_{\vec{x} \sim \mu} \left[\sum_{J \in \mathcal{J}} \theta_J \log \frac{1}{\phi_J(\vec{x}_J)} \right] - \mathrm{H}(\mu), \qquad (4)$$

called the *variational Gibbs free energy* (Mezard and Montanari 2009). GFE_Ψ is uniquely minimized by the distribution $\Pr_\Psi(\vec{x}) = \frac{1}{Z_\Psi} \prod_{J \in \mathcal{J}} \phi_J(\vec{x}_J)^{\theta_J}$, which matches the unweighted case when every $\theta_J = 1$. The mapping $\theta \mapsto \Pr_{(\Phi,\theta)}$ is known as Φ's *exponential family* and is a central tool in the analysis and development of many algorithms for graphical models (Wainwright, Jordan et al. 2008).

PDGs can in fact capture the full exponential family of a factor graph, but only by allowing values of α other than $\mathbf{1}$. In this case, the only definition that requires alteration is *IDef*, which now depends on the *weighted multigraph* $(\mathcal{G}^m, \alpha^m)$, and is given by

$$\mathit{IDef}_\mathcal{M}(\mu) := \sum_{X \xrightarrow{L} Y \in \mathcal{E}} \alpha_L \, \mathrm{H}_\mu(Y \mid X) - \mathrm{H}(\mu). \qquad (5)$$

Thus, the conditional entropy $\mathrm{H}_\mu(Y \mid X)$ associated with the edge $X \xrightarrow{L} Y$ is multiplied by the weight α_L of that edge.

One key benefit of using α is that we can capture arbitrary WFGs, not just ones with a constant weight vector. All we have to do is to ensure that in our translation from factor graphs to PDGs, the ratio α_L/β_L is a constant. (Of course, if we allow arbitrary weights, we cannot hope to do this if $\alpha_L = 1$ for all edges L.) We therefore define a family of translations, parameterized by the ratio of α_L to β_L.

[2]Recall that we identify the unweighted PDG $(\mathcal{G}, \mathbf{p})$ with the weighted PDG $(\mathcal{G}, \mathbf{p}, \mathbf{1}, \mathbf{1})$.

Definition 4.4 (WFG to PDG)**.** Given a WFG $\Psi = (\Phi, \theta)$, and postive number k, we define the corresponding PDG $\mathcal{M}_{\Psi,k} = (\mathcal{N}_\Phi, \alpha_\theta, \beta_\theta)$ by taking $\beta_J = k\theta_J$ and $\alpha_J = \theta_J$ for the edge $\mathbb{1} \to X_J$, and taking $\beta_L = k$ and $\alpha_L = 1$ for the projections $X_J \twoheadrightarrow X_j$. □

We now extend Definitions 4.2 and 4.3 to (weighted) PDGs and WFGs. In translating a PDG to a WFG, there will necessarily be some loss of information: PDGs have two sets, while WFGs have only have one. Here we throw out α and keep β, though in its role here as a left inverse of Definition 4.4, either choice would suffice.

Definition 4.5 (PDG to WFG)**.** Given a (weighted) PDG $\mathcal{M} = (\mathcal{N}, \beta)$, we take its corresponding WFG to be $\Psi_\mathcal{M} := (\Phi_\mathcal{N}, \beta)$; that is, $\theta_L := \beta_L$ for all edges L. □

We now show that we can capture the entire exponential family of a factor graph, and even its associated free energy, but only for γ equal to the constant k used in the translation.

Theorem 4.4. *For all WFGs* $\Psi = (\Phi, \theta)$ *and all* $\gamma > 0$*, we have that* $\mathit{GFE}_\Psi = {}^1\!/\!_\gamma [\![\mathcal{M}_{\Psi,\gamma}]\!]_\gamma + C$ *for some constant* C*, so* $\Pr_\Psi$ *is the unique element of* $[\![\mathcal{M}_{\Psi,\gamma}]\!]_\gamma^*$.

In particular, for $k = 1$, so that θ is used for both the functions α and β of the resulting PDG, Theorem 4.4 strictly generalizes Theorem 4.2.

Corollary 4.4.1. *For all weighted factor graphs* (Φ, θ)*, we have that* $\Pr_{(\Phi,\theta)} = [\![(\mathcal{N}_\Phi, \theta, \theta)]\!]_1^*$

Conversely, as long as the ratio of α_L to β_L is constant, the reverse translation also preserves semantics.

Theorem 4.5. *For all unweighted PDGs* $\mathcal{N}$ *and non-negative vectors* $\mathbf{v}$ *over* $\mathcal{E}^n$*, and all* $\gamma > 0$*, we have that* $[\![(\mathcal{N}, \mathbf{v}, \gamma\mathbf{v})]\!]_\gamma = \gamma \, \mathit{GFE}_{(\Phi_\mathcal{N}, \mathbf{v})}$*; consequently,* $[\![(\mathcal{N}, \mathbf{v}, \gamma\mathbf{v})]\!]_\gamma^* = \{\Pr_{(\Phi_\mathcal{N}, \mathbf{v})}\}$.

The key step in proving Theorems 4.4 and 4.5 (and in the proofs of a number of other results) involves rewriting $[\![\mathcal{M}]\!]_\gamma$ as follows:

Proposition 4.6. *Letting* $x^\mathbf{w}$ *and* $y^\mathbf{w}$ *denote the values of* X *and* Y*, respectively, in* $\mathbf{w} \in \mathcal{V}(\mathcal{M})$*, we have*

$$[\![\mathcal{M}]\!](\mu) = \mathop{\mathbb{E}}_{\mathbf{w} \sim \mu} \left\{ \sum_{X \xrightarrow{L} Y} \Bigg[\overbrace{\beta_L \log \frac{1}{\mathbf{p}_L(y^\mathbf{w} | x^\mathbf{w})}}^{\textit{log likelihood / cross entropy}} + \underbrace{(\alpha_L\gamma - \beta_L) \log \frac{1}{\mu(y^\mathbf{w} | x^\mathbf{w})}}_{\textit{local regularization (if } \beta_L > \alpha_L\gamma\textit{)}} \Bigg] - \underbrace{\gamma \log \frac{1}{\mu(\mathbf{w})}}_{\textit{global regularization}} \right\}. \qquad (6)$$

For a fixed γ, the first and last terms of (6) are equal to a scaled version of the free energy, $\gamma \, \mathit{GFE}_\Phi$, if we set $\phi_J := \mathbf{p}_L$ and $\theta_J := {}^{\beta_L}\!/\!_\gamma$. If, in addition, $\beta_L = \alpha_L\gamma$ for all edges L, then the local regularization term disappears, giving us the desired correspondence.

Equation (6) also makes it clear that taking $\beta_L = \alpha_L\gamma$ for all edges L is essentially necessary to get Theorems 4.2 and 4.3. Of course, fixed γ precludes taking the limit as γ goes to 0, so Proposition 3.4 does apply. This is reflected in some strange behavior in factor graphs trying to capture the same phenomena as PDGs, as the following example shows.

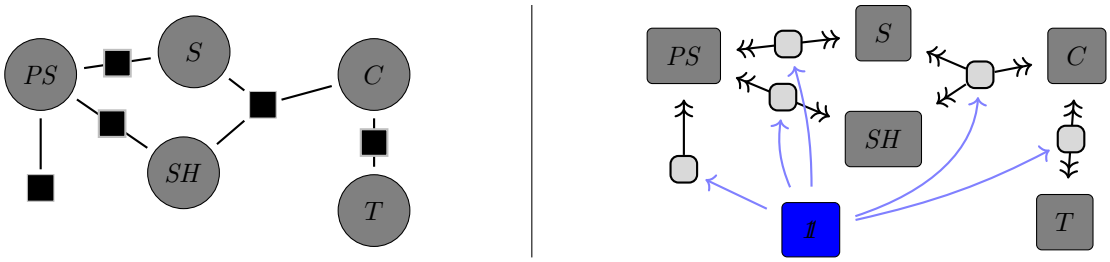


Figure 4: Conversion of the PDG in Example 2 to a factor graph according to Definition 4.2 (left), and from that factor graph back to a PDG by Definition 4.3 (right). In the latter, for each J we introduce a new variable X_J (displayed as a smaller darker rectangle), whose values are joint settings of the variables connected it, and also an edge $1 \to X_J$ (shown in blue), to which we associate the unconditional distribution given by normalizing ϕ_J.

Example 5. Consider the PDG $\mathcal{M}$ containing just X and 1, and two edges $p, q : 1 \to X$. (Recall that such a PDG can arise if we get different information about the probability of X from two different sources; this is a situation we certainly want to be able to capture!) Consider the simplest situation, where p and q are both associated with the same distribution on X; further suppose that the agent is certain about the distribution, so $\beta_p = \beta_q = 1$. For definiteness, suppose that $\mathcal{V}(X) = \{x_1, x_2\}$, and that the distribution associated with both edges is $\mu_{.7}$, which ascribes probability .7 to x_1. Then, as we would hope $[\![\mathcal{M}]\!]^* = \{\mu_{.7}\}$; after all, both sources agree on the information. However, it can be shown that $\Pr_{\Psi_m} = \mu_{.85}$, so $[\![\mathcal{M}]\!]^*_1 = \{\mu_{.85}\}$. □

Although both θ and β are measures of confidence, the way that the Gibbs free energy varies with θ is quite different from the way that the score of a PDG varies with β. The scoring function that we use for PDGs can be viewed as extending $GFE_{\Phi,\theta}$ by including the local regularization term. As γ approaches zero, the importance of the global regularization terms decreases relative to that of the local regularization term, so the PDG scoring function becomes quite different from Gibbs free energy.

5 Discussion

We have introduced PDGs, a powerful tool for representing probabilistic information. They have a number of advantages over other probablisitic graphical models.

- They allow us to capture inconsistency, including conflicting information from multiple sources with varying degrees of reliability.
- They are much more modular than other representations; for example, we can combine information from two sources by simply taking the union of two PDGs, and it is easy to add new information (edges) and features (nodes) without affecting previously-received information.
- They allow for a clean separation between quantitiatve information (the cpds and weights β) and more qualitative information contained by the graph structure (and the weights α); this is captured by the terms Inc and $IDef$ in our scoring function.
- PDGs have (several) natural semantics; one of them allows us to pick out a unique distribution. Using this distrbution, PDGs can capture BNs and factor graphs. In the latter case, a simple parameter shift in the corresponding PDG eliminates arguably problematic behavior of a factor graph.

We have only scratched the surface of what can be done with PDGs here. Two major issues that need to be tackled are inference and dynamics. How should we query a PDG for probabilistic information? How should we modify a PDG in light of new information or to make it more consistent? These issues turn out to be closely related. Due to space limitations, we just briefly give some intuitions and examples here.

Suppose that we want to compute the probability of Y given X in a PDG $\mathcal{M}$. For a cpd $p(Y|X)$, let $\mathcal{M}^{+p}$ be the PDG obtained by associating p with a new edge in $\mathcal{M}$ from X to Y, with $\alpha_p = 0$. We judge the quality of a candidate answer p by the best possible score that $\mathcal{M}^{+p}$ gives to any distribution (which we call the *degree of inconsistency* of $\mathcal{M}^{+p}$). It can be shown that the deegree of inconsistency is minimized by $[\![\mathcal{M}]\!]^*(Y \mid X)$. Since the degree of inconsistency of $\mathcal{M}^{+p}$ is smooth and strongly convex as a function of p, we can compute its optimum values by standard gradient methods. This approach is inefficient as written (since it involves computing the full joint distribution $[\![\mathcal{M}^{+p}]\!]^*$), but we believe that standard approximation techniques will allow us to draw inferences efficiently.

To take another example, conditioning can be understood in terms of resolving inconsistencies in a PDG. To condition on an observation $Y = y$, given a situation described by a PDG $\mathcal{M}$, we can add an edge from $\mathbb{1}$ to Y in $\mathcal{M}$, annoted with the cpd that gives probability 1 to y, to get the (possibly inconsistent) PDG $\mathcal{M}^{+(Y=y)}$. The distribution $[\![\mathcal{M}^{+(Y=y)}]\!]^*$ turns out to be the result of conditioning $[\![\mathcal{M}]\!]^*$ on $Y = y$. This account of conditioning generalizes without modification to give Jeffrey's Rule (Jeffrey 1968), a more general approach to belief updating.

Issues of updating and inconsistency also arise in variational inference. A variational autoencoder (Kingma and Welling 2013), for instance, is essentially three cpds: a prior $p(Z)$, a decoder $p(X \mid Z)$, and an encoder $q(Z \mid X)$. Because two cpds target Z (and the cpds are inconsistent until fully trained), this situation can be represented by PDGs but not by other graphical models. We hope to report further on the deep connection between inference, updating, and the resolution of inconsistency in PDGs in future work.

References

Halpern, J. Y.; and Leung, S. 2015. Weighted sets of probabilities and minimax weighted expected regret: new approaches for representing uncertainty and making decisions. *Theory and Decision* 79(3): 415–450.

Jeffrey, R. C. 1968. Probable knowledge. In Lakatos, I., ed., *International Colloquium in the Philosophy of Science: The Problem of Inductive Logic*, 157–185. Amsterdam: North-Holland.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes.

Koller, D.; and Friedman, N. 2009. *Probabilistic Graphical Models*. Cambridge, MA: MIT Press.

Kschischang, F. R.; Frey, B. J.; and Loeliger, H. . 2001. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory* 47(2): 498–519.

MacKay, D. J. C. 2003. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press.

Mezard, M.; and Montanari, A. 2009. *Information, physics, and computation*. Oxford University Press.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems*. San Francisco: Morgan Kaufmann.

Wainwright, M. J.; Jordan, M. I.; et al. 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning* 1(1–2): 1–305.

Ethics Statement

Because PDGs are a recent theoretical development, there is a lot of guesswork in evaluating the impact. Here are two views of opposite polarity.

5.1 Positive Impacts

One can imagine many applications of enabling simple and coherent aggregation of (possibly inconsistent) information. In particular we can imagine using PDGs to build and interpret a communal and global database of statistical models, in a way that may not only enable more accurate predictions, but also highlights conflicts between information.

This could have many benefits. Suppose, for instance, that two researchers train models, but use datasets with different racial makeups. Rather than trying to get an uninterpretable model to "get it right" the first time, we could simply highlight any such clashes and flag them for review.

Rather than trying to ensure fairness by design, which is both tricky and costly, we envision an alternative: simply aggregate (conflicting) statistically optimal results, and allow existing social structure to resolve conflicts, rather than sending researchers to fiddle with loss functions until they look fair.

5.2 Negative Impacts

We can also imagine less rosy outcomes. To the extent that PDGs can model and reason with inconsistency, if we adopt the attitude that a PDG need not wait until it is consistent to be used, it is not hard to imagine a world where a PDG gives biased and poorly-thought out conclusions. It is clear that PDGs need a great deal more vetting before they can be used for such important purposes as aggregating the world's statistical knowledge.

PDGs are powerful statistical models, but are by necessity semantically more complicated than many existing methods. This will likely restrict their accessibility. To mitigate this, we commit to making sure our work is widely accessible to researchers of different backgrounds.