Conditional Gaussian Probabilistic Decision Graphs

Jens D. Nielsen
Instituto de Investigación en Informática de Albacete
Universidad de Castilla-La Mancha
Campus Universitario s/n
02071 Albacete, Spain
dalgaard@dsi.uclm.es

Antonio Salmerón
Dpto. de Estadística y Matemática Aplicada
Universidad de Almería
La Cañada de San Urbano s/n
04120 Almería, Spain
antonio.salmeron@ual.es

Abstract

Probabilistic decision graphs (PDGs) are probabilistic graphical models that represent a factorisation of a discrete joint probability distribution using a "decision graph"-like structure over local marginal parameters. The structure of a PDG enables the model to capture some context specific independence relations that are not representable in the structure of more commonly used graphical models such as Bayesian networks and Markov networks. This sometimes makes operations in PDGs more efficient than in alternative models. PDGs have previously been defined only in the discrete case, assuming a multinomial joint distribution over the variables in the model. We extend PDGs to incorporate continuous variables, by assuming a Conditional Gaussian (CG) joint distribution. The CG model can be factorised as a product of conditionals. The conditional distribution of each discrete variable is multinomial while for each continuous variable it is Gaussian.

Introduction

The Probabilistic Decision Graph (PDG) model was introduced by (Bozga and Maler 1999) as an efficient representation of probabilistic transition systems. In this study, we consider the more general version of PDGs proposed in (Jaeger 2004).

PDGs are probabilistic graphical models that can represent some context specific independencies that are not efficiently captured by conventional graphical models, such as Markov Network or Bayesian Network (BN) models. Furthermore, probabilistic inference can be carried out directly in the PDG structure and has a time complexity linear in the size of the PDG model.

So far, PDGs have only been studied as representations of joint distributions over *discrete categorical* random variables, showing a competitive performance when compared to BN or Naïve BN models (Nielsen and Jaeger 2006). The PDG model has also been successfully applied to supervised classification problems (Nielsen, Rumí, and Salmerón 2009) and unsupervised clustering (Flores, Gámez, and Nielsen 2009).

In this paper, we introduce an extension of PDG models that incorporates *continuous* variables, and therefore expands the class of problems that can be handled by these models. More precisely, we define a new class of PDG models, called *conditional Gaussian PDGs* and show how they represent a joint distribution over a set of discrete and continuous variables, of class conditional Gaussian. We also show how probabilistic inference can be carried out over this new structure, taking advantage of the efficiency already shown for discrete PDGs.

The Conditional Gaussian model (CG)

We will use uppercase letters to denote random variables, and boldfaced uppercase letters to denote random vectors, e.g. $\mathbf{X} = \{X_0, X_1, \ldots, X_n\}$. By $R(X)$ we denote the set of possible states of variable X, and similarily for random vectors, $R(\mathbf{X}) = \times_{X_i \in \mathbf{X}} R(X_i)$. By lowercase letters x (or $\mathbf{x}$) we denote some element of $R(X)$ (or $R(\mathbf{X})$). When $\mathbf{x} \in R(\mathbf{X})$ and $\mathbf{Y} \subseteq \mathbf{X}$, we denote by $\mathbf{x}[\mathbf{Y}]$ the projection of $\mathbf{x}$ onto coordinates $\mathbf{Y}$. Throughout this document we will consider a set $\mathbf{W}$ of discrete variables and a set $\mathbf{Z}$ of continuous variables, and we will use $\mathbf{X} = \mathbf{W} \cup \mathbf{Z}$.

The Conditional Gaussian (CG) model (Lauritzen 1992; Lauritzen and Wermuth 1989) allows a factorised representation of a joint probability distribution over discrete and continuous variables, and that factorisation can be encoded by a Bayesian network with the restriction that discrete variables are not allowed to have continuous parents.

In the CG model, the conditional distribution of each discrete variable $W \in \mathbf{W}$ given their parents is a multinomial, whilst the conditional distribution of each continuous variable $Z \in \mathbf{Z}$ with discrete parents $\mathbf{E} \subseteq \mathbf{W}$ and continuous parents $\mathbf{V} \subseteq \mathbf{Z}$, is given by

$$f(z|\mathbf{E}=\mathbf{e}, \mathbf{V}=\mathbf{v}) = \mathcal{N}(z; \alpha(\mathbf{e}) + \beta(\mathbf{e})^T \mathbf{v}, \sigma^2(\mathbf{e})), \quad (1)$$

for all $\mathbf{e} \in R(\mathbf{E})$ and $\mathbf{v} \in R(\mathbf{V})$, where α and β are the coefficients of a linear regression model of Z given its continuous parents which could be a different model for each configuration of the discrete variables $\mathbf{E}$.

Discrete Multinomial PDGs

PDGs were introduced in (Jaeger 2004) as probabilistic graphical models of joint distributions over discrete variables.

In order to introduce the structure of a PDG model, we need to establish some graph-related notation. Let G be a

directed graph over nodes $\mathbf{V}$. Let $V \in \mathbf{V}$, we then denote by $pa_G(V)$ the set of parents of node V in G, by $ch_G(V)$ the set of children of V in G, by $de_G(V)$ the set of descendants of V in G and we use as shorthand notation $de^*_G(V) = de_G(V) \cup V$. By $pa^*_G(V)$ we understand the set of predecessors of V in G. The structure is formally defined as follows:

Definition 1 (The PDG Structure (Jaeger 2004)) *Let F be a forest of directed tree structures over a set of discrete random variables $\mathbf{W}$. A* PDG-structure *$G = \langle \mathbf{V}, \mathbf{E} \rangle$ for $\mathbf{W}$ w.r.t. F is a set of* rooted *DAGs, such that:*

1. *Each node $\nu \in \mathbf{V}$ is labelled with exactly one $W \in \mathbf{W}$. By $\mathbf{V}_W$, we will refer to the set of all nodes in a PDG structure labelled with the same variable W. For every variable W, $\mathbf{V}_W \neq \emptyset$, and we will say that ν represents W when $\nu \in \mathbf{V}_W$.*
2. *For each node $\nu \in \mathbf{V}_W$, each possible state $w \in R(W)$ and each successor $Y \in ch_F(W)$ there exists exactly* one *edge labelled with w from ν to some node ν' representing Y. Let $U \in ch_F(W)$, $\nu \in \mathbf{V}_W$ and $w \in R(W)$. By $succ(\nu, U, w)$ we will then refer to the unique node $\nu' \in \mathbf{V}_U$ that is reached from ν by an edge with label w.*

A PDG-structure is instantiated by assigning a real function f^ν to every node ν in the structure. The function must have the signature $f^\nu : R(W_i) \to \mathbb{R}$, where $\nu \in \mathbf{V}_{W_i}$.

An instantiated PDG structure G over the discrete variables $\mathbf{W}$ is called a Real Function Graph (RFG). It defines the (global) real function f_G with the signature $f_G : R(\mathbf{W}) \to \mathbb{R}$, by the following recursive definition:

Definition 2 *Let G be an RFG over discrete variables $\mathbf{W}$, and let $\nu \in \mathbf{V}_W$. We then define the local recursive functions:*

$$f^\nu_G(\mathbf{w}) := f^\nu(\mathbf{w}[W]) \prod_{Y \in ch_F(W)} f^{succ(\nu, Y, \mathbf{w}[W])}_G(\mathbf{w}), \quad (2)$$

for all $\mathbf{w} \in R(\mathbf{W})$. f_G is then defined on $R(\mathbf{W})$ as:

$$f_G(\mathbf{w}) := \prod_{\nu : \nu \text{ is a root}} f^\nu_G(\mathbf{w}). \quad (3)$$

The recursive function of Eq. (2) defines a factorisation that includes exactly one factor f^ν for each $W \in \mathbf{W}$. It will sometimes be convenient to be able to directly refer to the factor that is associated with a given element $\mathbf{w} \in \mathbf{W}$. The function *reach* defines exactly this association:

Definition 3 (Reach) *A node ν representing variable W_i in G is* reached *by $\mathbf{w} \in R(\mathbf{W})$ if*

1. *ν is a root in G, or*
2. *$W_j = \text{pa}_F(W_i)$, ν' representing W_j is* reached *by $\mathbf{w}$ and $\nu = \text{succ}(\nu', W_i, \mathbf{w}[W_j])$.*

By $reach_G(W_i, \mathbf{w})$ we denote the unique node representing W_i reached by $\mathbf{w}$ in PDG-structure G.

Using Def. 3, we can given an alternative definition of f_G:

$$f_G(\mathbf{w}) := \prod_{W_i \in \mathbf{W}} f^{reach_G(W_i, \mathbf{w})}(\mathbf{w}[W_i])\,. \quad (4)$$

When all the local functions f^ν in an RFG G over $\mathbf{W}$ define probability distributions, the function f_G (Def. 2) defines a joint multinomial probability distribution over $\mathbf{W}$ (see (Jaeger 2004)). In fact, f^ν_G in Eq. (2) defines a multinomial distribution over variables $W \cup de^*_F(W)$. We will refer to such RFGs as PDG models:

Definition 4 (The PDG model (Jaeger 2004)) *A PDG model $\mathcal{G}$ is a pair $\mathcal{G} = \langle G, \theta \rangle$, where $G = \langle \mathbf{V}, \mathbf{E} \rangle$ is a valid PDG-structure (Def. 1) over some set $\mathbf{W}$ of discrete random variables and $\theta = \{f^\nu : \nu \in \mathbf{V}\}$ is a set of real functions, each of which defines a discrete probability distribution.*

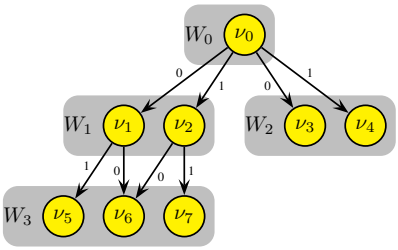

Figure 1: A PDG structure for variables W_0, W_1, W_2 and W_3.

Example 1 *Consider the PDG structure in figure 1. It encodes a factorisation of the joint distribution of $\mathbf{W} = \{W_0, W_1, W_2, W_3\}$, with*

$$\begin{aligned} f^{\nu_0} &= P(W_0), & f^{\nu_4} &= P(W_2|W_0 = 1), \\ f^{\nu_1} &= P(W_1|W_0 = 0), & f^{\nu_5} &= P(W_3|W_0 = 0, W_1 = 1), \\ f^{\nu_2} &= P(W_1|W_0 = 1), & f^{\nu_6} &= P(W_3|W_1 = 0), \\ f^{\nu_3} &= P(W_2|W_0 = 0), & f^{\nu_7} &= P(W_3|W_0 = 1, W_1 = 1). \end{aligned}$$

Conditional Gaussian PDGs

In this section we introduce an extension of the discrete multinomial PDG model defined in the previous section. The extension includes continuous variables in the model, and we will show afterwards that the factorisation now is a conditional Gaussian probability function.

Definition 5 (CG-PDG model) *A Conditional Gaussian PDG (CG-PDG) model is constructed over variables $\mathbf{X}$ by first constructing a forest F of trees such that* no continuous variable has a discrete variable as child. *Next, the discrete part of the structure is initialised as a PDG model according to Def. 4. The continuous part is subject to the following constraints:*

1. *For each node ν representing a continuous variable Z and for each variable $Z_c \in ch_F(Z)$, exactly* one *unique edge from ν to some $\nu' \in \mathbf{V}_{Z_c}$ exists, and ν' has no other parents than ν.*
2. *A node ν representing a continuous variable Z for which $pa^*_G(Z) \cap \mathbf{Z} = \mathbf{U}$, is parameterised with a vector $(\alpha_\nu, \beta_\nu, \sigma^2_\nu)$, meaning that $f^\nu(z) = f(z|\mathbf{x}) = \mathcal{N}(z; \alpha_\nu + \beta^T_\nu \mathbf{u}, \sigma^2_\nu)$. So f^ν is a Gaussian density with mean $\mu_Z = \alpha_\nu + \beta^T_\nu \mathbf{u}$ and variance $\sigma^2_Z = \sigma^2_\nu$, where β_ν is a vector*

of $|\mathbf{U}|$ *real values, and* $\mathbf{u}$ *is a vector of observations of variables* $\mathbf{U}$.

It is clear that when $\mathbf{Z} = \emptyset$, a CG-PDG model reduces to a PDG model. Also, observe that in a CG-PDG, for all ν's representing some $Z \in \mathbf{Z}$ where $Z_i \in ch_F(Z)$, the relation $succ(\nu, Z_i, z)$ is "constant" or invariant in z. We can therefore leave out the z argument and unambiguously write $succ(\nu, Z_i)$. Moreover, we have that $reach_G(Z, \mathbf{x}) = reach_G(Z, \mathbf{x}[\mathbf{W}])$ for any $X \in \mathbf{X}$, $\mathbf{x} \in R(\mathbf{X})$ and $Z \in \mathbf{Z}$.

We will extend the meaning of an RFG to include any graph with the structural syntax of Def. 5 and where nodes contain any real-valued function with the appropriate domain. The definition of the global function f_G in Def. 2 is still valid for such general RFGs and in particular for CG-PDG models.

The following proposition establishes that when $\mathcal{G}$ *is* a CG-PDG model, then f_G as defined in Def. 2 represents a CG distribution.

Proposition 1 *Let* $\mathcal{G}$ *be a CG-PDG model with structure* G *over variables* $\mathbf{X} = (\mathbf{W}, \mathbf{Z})$ *w.r.t. variable forest* F. *Function* f_G *defines a Conditional Gaussian density over* $\mathbf{X}$.

Proof: We have to show:

1. $f(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in R(\mathbf{X})$.
2. $$\sum_{\mathbf{w} \in R(\mathbf{W})} \int_{R(\mathbf{Z})} f_G(\mathbf{w}, \mathbf{z}) d\mathbf{z} = 1.$$
3. $\int_{R(\mathbf{Z})} f_G(\mathbf{x}) d\mathbf{z}$ is a multinomial distribution.
4. For each $\mathbf{w} \in R(\mathbf{W})$, $f_G(\mathbf{w}, \mathbf{z})$ is a multivariate Gaussian over $\mathbf{Z}$.

1. Trivially, $f_G(x) \geq 0$, since it is a product of non-negative terms.
2. We will show this by induction. First, as G is a CG-PDG over discrete variables $\mathbf{W}$ and continuous variables $\mathbf{Z}$, it is clear that $G \setminus \mathbf{Z}$ is a valid PDG over discrete variables $\mathbf{W}$. Therefore, by proposition 3.3 in [3], we know that:
$$\sum_{\mathbf{w} \in R(\mathbf{W})} f_{G \setminus \mathbf{Z}}(\mathbf{w}) \quad = \quad 1 \, .$$
Then, adding a single continuous variable $Z \in \mathbf{Z}$ as a leaf, we get:
$$\sum_{\mathbf{w} \in R(\mathbf{W})} \int_{R(Z)} f_{G \setminus \{\mathbf{Z} \setminus Z\}}(\mathbf{w}, z) dz =$$
$$\sum_{\mathbf{w} \in R(\mathbf{W})} f_{G \setminus \mathbf{Z}}(\mathbf{w}) \int_{R(Z)} f^{reach(Z, \mathbf{w})}(z) dz = 1 \, .$$
This addition of continuous variables can be repeated until we get G, and thus
$$\sum_{\mathbf{w} \in R(\mathbf{W})} \int_{R(\mathbf{Z})} f_G(\mathbf{w}, \mathbf{z}) d\mathbf{z} = 1.$$

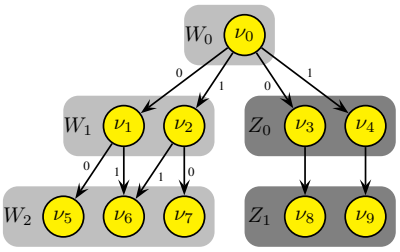

Figure 2: Sturcture of a CG-PDG with three discrete and two continuous variables.

3. If we fix a configuration $\mathbf{w} \in R(\mathbf{W})$, then f_G is just a product of functions of the form f^{ν}, where ν is a continuous parameter node, and therefore, f_G is a product of conditional Gaussians in each branch of the trees in the forest of variables restricted to $\mathbf{w}$, and therefore f_G is a multivariate Normal over the continuous variables $\mathbf{Z}$. Thus,
$$\int_{R(\mathbf{Z})} f_G(\mathbf{w}, \mathbf{z}) d\mathbf{z} = f_{G_{\mathbf{W}}}(\mathbf{w}) \prod_{\nu' \in V} \int_{R(\mathbf{Z})} f_G^{\nu'}(\mathbf{z}) d\mathbf{z} = f_{G_{\mathbf{W}}}(\mathbf{w}),$$
where V is the set of parameter nodes representing a continuous variable with either no parent or a discrete parent in the variable structure and $G_{\mathbf{W}}$ is the PDG obtained from structure G by keeping only the variables in $\mathbf{W}$.
4. It was shown in the previous step.

■

Before going further, we will give an example of how a CG-PDG model naturally captures the structure of a problem domain.

Example 2 *A newspaper delivery van has two possible delivery routes, one of them covering only city* A *and the other covering city* B *as well. A 70% of the days, the selected route is the one including only city* A. *Let us denote by* W_0 *the delivery route (*$0 = A, 1 = A - B$*). Cities* A *and* B *are connected by a pay motorway, with a toll fee of 3 Euro. City* B *is known to be a busy city traffic much more dense than* A, *so that the probability of suffering a traffic jam (denoted as* W_1, *with values 0=no and 1=yes) when the selected route includes* B *is* 0.05, *and* 0.01 *otherwise. If the van suffers a traffic jam, the probability of completing the delivery on time (*W_2, *with values 0=no, 1=yes) is only* 0.5 *regardless of the selected route. If there are no traffic jams, the probability of completing the job on time is* 0.95 *for route* A *and* 0.8 *for route* $A - B$. *The cost of the delivery (*Z_1*) depends on the selected route and on the gas consumption (*Z_0*). The gas consumption follows a Gaussian distribution with mean equal to 5 litres and variance of 1 litre*^{2} *for route* A, *whilst the mean is 10 and the variance 1.2 for the other route. The cost also follows a Gaussian distribution, with mean equal to 1.1 times the consumed litres and variance 0.5 when the route is* A, *and if the route is* $A - B$, *the mean is increased by the toll fee. The structure in figure 2 represents the dependence structure described in this example. A parameterisation of that structure, according to definition 5 and the information given*

above is as follows: $f^{\nu_0} = P(W_0) = (0.7, 0.3)$, $f^{\nu_1} = P(W_1|W_0 = 0) = (0.99, 0.01)$, $f^{\nu_2} = P(W_1|W_0 = 1) = (0.95, 0.05)$, $f^{\nu_3} = f(z_0|W_0 = 0) = \mathcal{N}(z_0; 5, 1^2)$, $f^{\nu_4} = f(z_0|W_0 = 1) = \mathcal{N}(z_0; 10, 1.2^2)$, $f^{\nu_5} = P(W_2|W_0 = 0, W_1 = 0) = (0.05, 0.95)$, $f^{\nu_6} = P(W_2|W_1 = 1) = (0.5, 0.5)$, $f^{\nu_7} = P(W_2|W_0 = 1, W_1 = 0) = (0.2, 0.8)$, $f^{\nu_8} = f(z_1|z_0, W_0 = 0) = \mathcal{N}(z_1; 1.1z_0, 0.5^2)$, $f^{\nu_9} = f(z_1|z_0, W_0 = 1) = \mathcal{N}(z_1; 3 + 1.1z_0, 0.5^2)$.

The efficiency of the PDG model over exclusively discrete domains stems from their structure which is a special kind of decision graph only containing chance nodes. The first PDG version presented by (Bozga and Maler 1999) extends Binary Decision Diagrams (BDDs) and thereby inherits the efficiency of BDDs, which lies in compact representation and efficient manipulation of boolean functions.

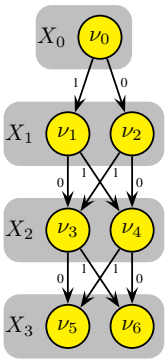

Figure 3: PDG-representation of the parity function.

In Fig. 3, a PDG over 4 binary variables is depicted. The structure encodes the model where X_4 is determined by the parity function over the remaining 3 variables that are marginally independent. Adding more variables to the parity function only makes the model grow in size by a small linear factor. Modelling the parity function using a BN model would yield a model that grows by an exponential factor when adding more variables to the function.[1]

The efficiency of the discrete PDG, exemplified by the representation of the parity function (Fig. 3) is inherited by the CG-PDG model. The addition of continuous variables does not restrict the discrete part of the CG-PDG in any way, and the properties of this part of the model stays intact.

Operations over CG-PDGs

One of the main advantages of the PDG model is that efficient algorithms for exact inference that operate directly on the PDG structure are known. In this section we will show how the original algorithm for exact inference in discrete PDGs by (Jaeger 2004) can be almost directly applied to CG-PDGs.

We will first consider the problem of computing the probability of some set of variables $\mathbf{Y} \subset \mathbf{X}$ being in the joint state $\mathbf{y} \in R(\mathbf{Y})$ when the joint distribution $P(\mathbf{X})$ is represented by a CG-PDG model $\mathcal{G}$ with structure G. The computation that we wish to perform is what is usually called marginalisation or restriction:

$$P\{\mathbf{Y} = \mathbf{y}\} = \sum_{\mathbf{w} \in R(\mathbf{W} \setminus \mathbf{Y})} \int_{R(\mathbf{Z} \setminus \mathbf{Y})} f_G(\mathbf{w}, \mathbf{z}, \mathbf{y}) d\mathbf{z} \,. \quad (5)$$

[1] By including suitable artificial latent variables in the domain, there exists an efficient transformation of any PDG into an equivalent BN model (Jaeger 2004).

The next definition is the first step towards efficient computation of Eq. (5).

Definition 6 (Restriction) *Let $\mathcal{G}$ be a CG-PDG over variables $\mathbf{X}$, let $\mathbf{Y} \subseteq \mathbf{X}$ and let $\mathbf{y} \in R(\mathbf{Y})$. The restriction of $\mathcal{G}$ to $\mathbf{Y} = \mathbf{y}$, denoted as $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$ is an RFG obtained from $\mathcal{G}$ such that*

1. *$\mathcal{G}$ and $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$ has equivalent structure.*
2. *For all ν representing some variable $X \in \mathbf{X} \setminus \mathbf{Y}$, f^ν remains unchanged.*
3. *For every discrete variable $W \in \mathbf{Y} \cap \mathbf{W}$ and each node $\nu \in \mathbf{V}_W$, the function $f^\nu(w)$ in $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$ is unchanged from $\mathcal{G}$ for $w = \mathbf{y}[W]$ and for any $w \neq \mathbf{y}[Y]$ we set $f^\nu(w) = 0$.*
4. *For every continuous variable Z and every node $\nu \in \mathbf{V}_Z$ a real vector $\mathbf{u}_\nu$ is constructed. $\mathbf{u}_\nu$ is indexed by the variables $\mathbf{U} = pa^*_F(Z) \cap \mathbf{Z}$ and with values $\mathbf{u}[U] = \mathbf{y}[U]$ if $U \in \mathbf{Y}$ and $\mathbf{u}[U] = \alpha_{\nu_U}$ (where ν_U is the unique predecessor node of ν representing U). Then a (conditional) mean μ_ν is computed as $\mu_\nu = \alpha_\nu + \beta_\nu^T \mathbf{u}$. Once μ_ν has been computed, in all nodes ν representing $Z \in \mathbf{Y} \cap \mathbf{Z}$, we replace f^ν with the function value $f^\nu(\mathbf{y}[Z])$.*

We call the resulting model a restricted CG-PDG.

From a *restricted* CG-PDG $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$ we can compute the probability of the evidence $P\{\mathbf{Y} = \mathbf{y}\}$ as:

$$P\{\mathbf{Y} = \mathbf{y}\} = \sum_{\mathbf{w} \in R(\mathbf{W})} \int_{R(\mathbf{Z})} f_{G_{\mathbf{Y}=\mathbf{y}}}(\mathbf{w}, \mathbf{z}) d\mathbf{z} \,. \quad (6)$$

In the first part of this section we will show how (6) is computed by local computations in the nodes.

We define the *outflow* as the accumulated function value of the real function f^ν_G defined recursively at ν by Eq. (2) over its full domain.

Definition 7 *Let $\mathcal{G}$ be a (possibly restricted) CG-PDG with structure G over variables $\mathbf{X}$ w.r.t. forest F. The* outflow *of ν is defined as:*

$$ofl(\nu) := \sum_{\mathbf{w} \in R(\mathbf{W} \cap de^*_F(X_i))} \int_{R(\mathbf{Z} \cap de^*_F(X_i))} f^\nu_G(\mathbf{w}, \mathbf{z}) d\mathbf{z}. \quad (7)$$

Notice that in an unrestricted CG-PDG the outflow of all nodes is 1. Also notice that Eq. (6) is equal to the product of outflows of all root nodes in the structure.

The next proposition is central in the efficient computation of *outflow*:

Proposition 2 *Let $\mathcal{G}$ be a (possibly restricted) CG-PDG with structure G w.r.t. forest F over variables $\mathbf{X}$. The outflow is recursively computed as follows:*

1. *If ν is a parameter node of a discrete variable W:*

$$ofl(\nu) = \sum_{w \in R(W)} f^\nu(w) \prod_{Y \in ch_F(W)} ofl(succ(\nu, Y, w)) \,. \quad (8)$$

2. *If ν is a parameter node of a continuous variable Z:*

$$ofl(\nu) = \int_{R(Z)} f^\nu(z) \prod_{Y \in ch_F(Z)} ofl(succ(\nu, Y)) dz \,. \quad (9)$$

Proof: Item 1 is shown in (Jaeger 2004, Lemma 4.3). To prove item 2 we just have to remember that, in a RFG containing continuous variables, all the variables below any continuous variable are continuous as well. Therefore, we have to instantiate Eq. (7) to the case in which there are no discrete variables involved and hence the summation disappears and we are left with only the integration of function f_G^ν. Expanding f_G^ν using Eq. (2) we get Eq. (9). ■

Extending previous results of (Jaeger 2004, Theorem 4.4), Proposition 2 and the fact that Eq. (6) equals the product of outflows of root nodes, yields an efficient computation of $P\{\mathbf{Y} = \mathbf{y}\}$.

We will now turn to the computation of posterior probability distribution $P(W|\mathbf{Y} = \mathbf{y})$ and posterior densities $f(z|\mathbf{Y} = \mathbf{y})$. We will need to be able to talk about parts of a domain $R(\mathbf{U})$, $\mathbf{U} \subseteq \mathbf{X}$, that reach a specific node, so we define a *Path*-relation as follows:

Definition 8 (Path) *Let $\mathcal{G}$ be a (possibly restricted) CG-PDG model with structure G w.r.t. forest F over variables $\mathbf{X}$ and let ν represent $X \in \mathbf{X}$ and let $pa_F^*(X) \subseteq \mathbf{Y} \subseteq \mathbf{X}$. Then*

$$Path_G(\nu, \mathbf{Y}) := \{\mathbf{y} \in R(\mathbf{Y}) \text{ such that } \exists \mathbf{x} \in R(\mathbf{X}) : (reach_G(\mathbf{x}, X) = \nu \text{ and } \mathbf{x}[\mathbf{Y}] = \mathbf{y})\}. \quad (10)$$

The *inflow* of a node ν is the accumulation of values of f_G over the part of the domain that reaches ν, and we define it formally as follows:

Definition 9 *Let $\mathcal{G}$ be a CG-PDG model with structure G over variables $\mathbf{X} = (\mathbf{W}, \mathbf{Z})$ and forest F. Let $\nu \in \mathbf{V}_{X_i}$, $G \setminus X_i$ be the structure obtained from G by removing every node labelled with X_i and their descendants, $\mathbf{W}' = \mathbf{W} \setminus de_F^*(X_i)$ and $\mathbf{Z}' = \mathbf{Z} \setminus de_F^*(X_i)$. The* inflow *of ν is defined as:*

$$ifl(\nu) := \sum_{\mathbf{w} \in Path_G(\nu, \mathbf{W}')} \int_{R(\mathbf{Z}')} f_{G \setminus X_i}(\mathbf{w}, \mathbf{z}) d\mathbf{z}. \quad (11)$$

When $\{\mathbf{W}' \cup \mathbf{Z}'\} = \emptyset$, we define $ifl(\nu) = 1$.

For a node ν in a CG-PDG with structure G over $\mathbf{X}$, the set $Path_G(\nu, \mathbf{X})$ is the part of the domain in which the local function f^ν is included as a factor in the global function f_G. The *inflow* and *outflow* of a node ν factorises the accumulated function value of f_G over $Path_G(\nu, \mathbf{X})$ in two independent factors.

Lemma 1 *Let $\mathcal{G}$ be a (possibly restricted) CG-PDG with structure G over variables $\mathbf{X}$. For any node ν in G, it holds that*

$$ifl(\nu) ofl(\nu) = \sum_{\mathbf{w} \in Path_G(\nu, \mathbf{W})} \int_{Path_G(\nu, \mathbf{Z})} f_G(\mathbf{w}, \mathbf{z}) d\mathbf{z}. \quad (12)$$

Proof: We wish to compute the product $ifl(\nu) ofl(\nu)$ for an arbitrary node ν in a CG-PDG. Let node ν represent variable X_i, then $Path_G(\nu, \mathbf{W})$ can be decomposed as $Path_G(\nu, \mathbf{W}) = Path_G(\nu, \mathbf{W} \setminus de_F^*(X_i)) \times R(\mathbf{W} \cap de_F^*(X_i))$, and obviously $R(\mathbf{Z})$ can be decomposed as $R(\mathbf{Z}) = R(\mathbf{Z} \setminus de_F^*(X_i)) \times R(\mathbf{Z} \cap de_F^*(X_i))$. Then:

$$ifl(\nu) ofl(\nu) = \sum_{\substack{\mathbf{w} \in \\ Path_G(\nu, \mathbf{W})}} \int_{R(\mathbf{Z})} f_{G \setminus X_i}(\mathbf{w}', \mathbf{z}') f_G^\nu(\mathbf{w}'', \mathbf{z}'') d\mathbf{z},$$

where $\mathbf{w}'$ (and $\mathbf{z}'$) are projections of $\mathbf{w}$ (and $\mathbf{z}$) onto $\mathbf{X} \setminus de_F^*(X_i)$, while $\mathbf{w}''$ (and $\mathbf{z}''$) are projections onto $de_F^*(X_i)$. Finally, from Def. 2 we have that the product $f_{G \setminus X_i}(\mathbf{w}', \mathbf{z}') f_G^\nu(\mathbf{w}'', \mathbf{z}'')$ equals $f_G(\mathbf{w}, \mathbf{z})$. ■

The next proposition establishes the basis for probabilistic inference in CG-PDGs.

Proposition 3 *Let $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$ be a CG-PDG model restricted to evidence $\mathbf{Y} = \mathbf{y}$. When* ifl *and* ofl *values have been computed for all nodes in $\mathcal{G}_{\mathbf{Y}=\mathbf{y}}$, the following holds. For any discrete variable $W \in \mathbf{W}$ where $W \notin \mathbf{Y}$,*

$$P\{W = w|\mathbf{Y} = \mathbf{y}\} = \gamma \sum_{\nu \in \mathbf{V}_W} f^\nu(w) ifl(\nu) \prod_{U \in ch_F(W)} ofl(succ(\nu, U, w)). \quad (13)$$

For any continuous variable $Z \in \mathbf{Z}$, $Z \notin \mathbf{Y}$, it holds that

$$f(z|\mathbf{Y} = \mathbf{y}) = \gamma \sum_{\nu \in \mathbf{V}_Z} f^\nu(z) ifl(\nu) \prod_{U \in ch_F(Z)} ofl(succ(\nu, U)). \quad (14)$$

Furthermore,

$$\mathrm{E}[Z|\mathbf{Y} = \mathbf{y}] = \gamma \sum_{\nu \in \mathbf{V}_Z} \mu_\nu ifl(\nu) \prod_{U \in ch_F(Z)} ofl(succ(\nu, U)), \quad (15)$$

and

$$\mathrm{Var}(Z|\mathbf{Y} = \mathbf{y}) = \gamma \sum_{\nu \in \mathbf{V}_Z} \sigma_\nu^2 ifl(\nu)^2 \prod_{U \in ch_F(Z)} ofl(succ(\nu, U))^2. \quad (16)$$

In all equations γ is the normalising factor $\frac{1}{P\{\mathbf{Y}=\mathbf{y}\}}$. In Eq. (15), μ_ν is computed during restriction (see Def. 6).

Proof: Equations (13) and (14) are a direct consequence of lemma 1. Also, note that $f(z|\mathbf{Y} = \mathbf{y})$ in equation (14) is a mixture of Gaussian densities, and therefore the expectation of Z is trivially the one in equation (15) and its variance is the one in equation (16). ■

The next proposition is central in the efficient computation of *inflow*.

Proposition 4 *Let $\mathcal{G}$ be a (possibly restricted) CG-PDG with structure G w.r.t. forest F over variables $\mathbf{X}$. The inflow is recursively computed as follows:*

1. *If ν is a root,*

$$ifl(\nu) = \prod_{\nu' \neq \nu, \nu' \text{ is root}} ofl(\nu'). \quad (17)$$

2. *If ν is not a root, and $X_p = pa_F(X_i)$, and X_p is discrete:*

$$ifl(\nu) = \sum_{x \in R(X_p)} \sum_{\substack{\nu': \\ \nu = succ(\nu', X_i, x)}} [ifl(\nu') f^{\nu'}(x) \prod_{Y \in ch_F(X_p) \setminus X_i} ofl(succ(\nu', Y, x))] . \quad (18)$$

3. *If ν is representing continuous variable X_i, ν is not a root, $X_p = pa_F(X_i)$, X_p is continuous and ν' is the parent of ν:*

$$ifl(\nu) = ifl(\nu') \prod_{Y \in ch_F(X_p) \setminus X_i} ofl(succ(\nu', Y)) . \quad (19)$$

Proof: Items 1 and 2 are shown in (Jaeger 2004, Lemma 4.3). Item 3 follows by realizing that continuous parameter nodes only have one outgoing arc, at most, towards each child variable. ■

Proposition 5 *Computing* inflow *and* outflow *for all nodes in a (possibly restricted) CG-PDG can be done in time linear in the number of edges of the model.*

Proof: The proof is a simple extension of the proof of the result (Jaeger 2004, Theorem 4.4). ■

Propositions 3 and 5 demonstrate that typical probabilistic queries can be answered in time linear in the size of the CG-PDG. The main concern in achieving efficient inference can therefore be directly focused on constructing a small model, which of course may be difficult or even impossible. The size may be exponential in the number of discrete variables in the domain. However, it is considered an advantage to be able to determine complexity of inference directly in the model, as opposed to BN models where inference complexity depends on the size of a secondary Junction Tree model as opposed to the BN model itself.

Example 3 (CG-PDG belief updating) *Consider Ex. 2. Assume we have evidence that the route was not finished in time ($W_2 = 0$), and we then want to update our beliefs of the remaining unknown variables. Restricting the model to evidence $\{W_2 = 0\}$ results in the following changes: $f^{\nu_5}(1) = f^{\nu_6}(1) = f^{\nu_7}(1) = 0$, $\mu_{\nu_8} = 5.5$ and $\mu_{\nu_9} = 8.5$. After restricting our model, we compute outflows, here we list values consecutively as $\{ofl(\nu_0), ofl(\nu_1) \dots ofl(\nu_9)\}$: $\{0.10265, 0.0545, 0.215, 1, 1, 0.05, 0.5, 0.2, 1, 1\}$. Once outflows are computed, inflows can be computed: $\{1, 0.7, 0.3, 0.03815, 0.0645, 0.693, 0.022, 0.285, 0.03815, 0.0645\}$.*

First, as mentioned earlier, the probability of evidence is just the product of outflows of root nodes which in this example means just $ofl(\nu_0) = P\{W_2 = 0\} = 0.10265$. Next, computing the posterior expectations of the continuous variables is done top down from the root to the leaves using Eq. (15) with $\gamma = \frac{1}{P\{W_2=0\}}$, and we get $\mathrm{E}[Z_0|W_2 = 0] \approx 8.14$ and $\mathrm{E}[Z_1|W_2 = 0] \approx 7.39$.

Posterior variances are computed as a weighted average of the variances stored in nodes representing the given variable using Eq. (16), which yields: $\mathrm{Var}[Z_0|W_2 = 0] \approx 0.07$ and $\mathrm{Var}[Z_1|W_2 = 0] \approx 0.01$.

Finally, computing the marginal distributions for the two unobserved categorical variables W_0 and W_1 we use Eq. (13) and get: $P\{W_0|W_2 = 0\} \approx \{0.37, 0.63\}$ and $P\{W_1|W_2 = 0\} \approx \{0.89, 0.11\}$.

Concluding remarks

In this paper we have introduced the CG-PDG model, an extension of PDSs able to represent hybrid probabilistic models with joint conditional Gaussian distribution. The new model keeps the expression power and representational efficiency of its predecessor in what concerns the discrete part, and the continuous part is also compactly represented with a number of parameter linear on the number of continuous variables once the discrete part is fixed.

We have shown how probabilistic inference can be carried out efficiently by using the concepts of inflow and outflow of nodes, and taking advantage of the recursive computations of both quantities.

In the near future we plan to extend the PDGs to another hybrid model, namely the MTE (mixture of truncated exponentials) model (Moral, Rumí, and Salmerón 2001), in which no structural restrictions, regarding arrangement of discrete and continuous variables, are imposed.

Acknowledgements

This work has been supported by the Spanish Ministry of Science and Innovation, through projects TIN2007-67418-C03-01,02 by EFRD funds.

References

Bozga, M., and Maler, O. 1999. On the Representation of Probabilities over Structured Domains. In *Proceedings of the 11th CAV Conference*, 261–273. Springer.

Flores, M. J.; Gámez, J. A.; and Nielsen, J. D. 2009. The pdg-mixture model for clustering. In *Proceedings of DaWaK09 Conference*, 378–389.

Jaeger, M. 2004. Probabilistic Decision Graphs - Combining Verification and AI Techniques for Probabilistic Inference. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 12:19–42.

Lauritzen, S., and Wermuth, N. 1989. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The Annals of Statistics* 17:31–57.

Lauritzen, S. 1992. Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association* 87:1098–1108.

Moral, S.; Rumí, R.; and Salmerón, A. 2001. Mixtures of truncated exponentials in hybrid Bayesian networks. In *ECSQARU'01. LNAI*, volume 2143, 135–143.

Nielsen, J. D., and Jaeger, M. 2006. An empirical study of efficiency and accuracy of probabilistic graphical models. In *Proceedings of the PGM06 Workshop*, 215–222.

Nielsen, J. D.; Rumí, R.; and Salmerón, A. 2009. Supervised classification using probabilistic decision graphs. *Comp. Statistics and Data Analysis* 53:1299–1311.