Synthesis of Function-Described Graphs

René Alquézar[1], Alberto Sanfeliu[2] and Francesc Serratosa[3]

[1] Departament de Llenguatges i Sistemes Informàtics, Univ. Politècnica de Catalunya, Jordi Girona Salgado 1-3, Mòdul C6, 08034 Barcelona, Spain
[2] Institut de Robòtica i Informàtica Industrial, UPC - CSIC, Barcelona
[3] Departament d'Enginyeria Informàtica, Univ. Rovira i Virgili, Tarragona
alquezar@lsi.upc.es, asanfeliu@iri.upc.es, fserrato@etse.urv.es

Abstract. Function-described graphs (FDGs) have been introduced very recently as a representation of an ensemble of attributed relational graphs (ARGs) for structural pattern recognition [1, 2]. In this paper, the relationship between FDGs and Random Graphs [3] is analysed and the synthesis process of FDGs is studied, whereas the matching process between FDGs and ARGs is discussed elsewhere [4]. Two procedures are described to synthesize an FDG from a set of commonly labelled ARGs or FDGs, respectively. Then, a general incremental method to synthesize an FDG from a sequence of unlabelled ARGs is proposed. Specific methods are obtained when an ARG-to-FDG matching algorithm is selected to find optimal morphisms.

Keywords : graph synthesis, function-described graphs, random graphs, attributed relational graphs, structural pattern recognition.

1 Introduction

Attributed relational graphs (ARGs) are used in some pattern recognition tasks to represent both structural and semantic information of complex objects [5, 6]. *Random graphs* were proposed by Wong *et al.* [3, 7] for modelling classes of patterns described by ARGs, through a high-order joint probability space of random variables. However, to apply them in practical problems, the dimensionality of this space must be reduced and random graphs simplified by making some strong assumptions about the statistical independence of random nodes and edges [3, 8]. Since relevant structural information, such as the presence or absence of nodes and edges, is embedded in the probabilities of the random variables, the independence assumptions may lead to improper structural generalizations when synthesizing a random graph from a set of ARGs.

To overcome this problem, function-described graphs (FDGs) have been introduced recently as a new representation for a set of ARGs [1, 2]. It is shown here that FDGs can be seen as a different type of simplification of the general random graphs in which some structural constraints are recorded. On the other hand, both incremental [8] and non-incremental [3] clustering methods have been proposed to build a hierarchy of random graphs, that are synthesized from a given set of ARGs and represent the different classes and sub-classes of patterns included in the input data. In this paper, the synthesis process of FDGs is studied and a general incremental algorithm is given to synthesize an FDG from a sequence of ARGs belonging to a common class.

2 Attributed relational graphs and random graphs

The definitions given in this section come from several previous works [3],[5]-[8], and they are rewritten here (using a convenient notation) as necessary background for the definition of function-described graphs in the next section.

Let $\Sigma_v = \{v_k \mid k = 1, \ldots, n\}$ be a set of vertices (or nodes), and let $\Sigma_e = \{e_{ij} \mid i, j \in \{1, \ldots, n\}\}$ be a set of edges (or arcs), where the arc e_{ij} connects vertices v_i and v_j. Let $Z_v = \{z_i \mid i = 1, \ldots, t\}$ be a nonempty finite set of attribute names for the attributes in a vertex, and for each z_i in Z_v let D_{vi} denote the corresponding domain of attribute values. Similarly, let $Z_e = \{z'_i \mid i = 1, \ldots, s\}$ be a nonempty finite set of attribute names for the attributes in an arc, and let D_{ei} denote the domain of attribute values for z'_i. An attributed vertex v_k is associated with an attribute set in which each element is an attribute pair (z_i, a_i) consisting of an attribute name $z_i \in Z_v$ and its attribute value $a_i \in D_{vi}$. Likewise, an attributed arc e_{pq} is associated with an attribute set containing pairs (z'_i, b_i), $z'_i \in Z_e$, $b_i \in D_{ei}$. An attributed graph contains a set of attributed vertices, that typically represent primitives, and a set of attributed arcs, that represent relations between these primitives [5].

Let us assume in the following that attributed graphs are *homogeneous*, i.e. all vertices have the same attributes Z_v and all arcs have the same attributes Z_e. In such a case, the pairs in an attribute set can be listed always in the same order and the attribute names supressed, and thus an attributed vertex v_k (respectively arc e_{pq}) can be associated with a t-tuple (respectively s-tuple) of vertex (arc) attribute values. Hence, let $\Delta_v = \{(a_1, \ldots, a_i, \ldots, a_t) \mid a_i \in D_{vi}, 1 \le i \le t\}$ be the global domain for attributed vertices and let $\Delta_e = \{(b_1, \ldots, b_i, \ldots, b_s) \mid b_i \in D_{ei}, 1 \le i \le s\}$ be the global domain for attributed arcs. An *attributed relational graph* (ARG) is formally defined to be a graph $G = (\Sigma_v, \Sigma_e, \Delta_v, \Delta_e, \gamma_v, \gamma_e)$, where (Σ_v, Σ_e) corresponds to the underlying graph structure, $\gamma_v : \Sigma_v \to \Delta_v$ and $\gamma_e : \Sigma_e \to \Delta_e$ are two mappings that assign attribute values to nodes and edges, respectively. For each vertex v_k, let $\mathbf{a}^k = \gamma_v(v_k) = (a_1^k, \ldots, a_t^k)$, and for each arc e_{ij}, let $\mathbf{b}^{ij} = \gamma_e(e_{ij}) = (b_1^{ij}, \ldots, b_s^{ij})$.

A random graph is a graph structure with randomly varying vertex and arc attribute values [3, 7]. Any ARG obtained by instantiating all random vertices and random arcs is called an outcome graph of the random graph. Hence, a random graph represents the set of all possible ARGs that can be outcome graphs of it. Next, a more precise definition is given.

Let ϕ denote a null value for attributed vertices and arcs. Let $\Delta_\omega = \Delta_v \cup \{\phi\}$ and $\Delta_\varepsilon = \Delta_e \cup \{\phi\}$. A (general) *random graph* is defined to be a tuple $R = (\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon, P)$, where $(\Sigma_\omega, \Sigma_\varepsilon)$ is the underlying graph structure, $\gamma_\omega : \Sigma_\omega \to \Omega_\omega$ where Ω_ω is a set of random variables with values in Δ_ω (random vertices), $\gamma_\varepsilon : \Sigma_\varepsilon \to \Omega_\varepsilon$ where Ω_ε is a set of random variables with values in Δ_ε (random arcs), and P is a joint probability distribution $P(\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_m)$ of all random vertices $\{\alpha_i \mid \alpha_i = \gamma_\omega(\omega_i), 1 \le i \le n\}$ and random arcs $\{\beta_j \mid \beta_j = \gamma_\varepsilon(\varepsilon_{kl}), 1 \le j \le m\}$.

Let $C(R)$ be the set of all outcome graphs of a random graph R. For each $G \in C(R)$, a probability measure $P_R(G)$ is given by the sum of the joint probabilities of random vertices and arcs over all instantiations which produce G, and any such instantiation is associated with a structural isomorphism $\mu : G' \to R$, where G' is the extension of G to the order of R obtained by adding null vertices and arcs appropriately[4]. Let $G = (\Sigma_v, \Sigma_e, \Delta_v, \Delta_e, \gamma_v, \gamma_e)$ be an ARG in $C(R)$ for a random graph $R = (\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon, P)$, and let G be oriented with respect to R by isomorphism $\mu : G' \to R$. For each vertex ω_i of R, let $v_k = \mu^{-1}(\omega_i)$ and then define $\mathbf{a}^i = \gamma_v(v_k)$ if v_k is non-null and $\mathbf{a}^i = \phi$ otherwise. Similarly, for each arc ε_j of R, let $e_{kl} = \mu^{-1}(\varepsilon_j)$ and then define $\mathbf{b}^j = \gamma_e(e_{kl})$ if e_{kl} is non-null and $\mathbf{b}^j = \phi$ otherwise. Then, the probability of G to be an outcome of R with orientation μ is given by

$$P_R(G', \mu) = Pr\left(\bigwedge_{i=1}^{n} (\alpha_i = \mathbf{a}^i) \wedge \bigwedge_{j=1}^{m} (\beta_j = \mathbf{b}^j) \right) \ . \tag{1}$$

General random graphs are absolutely impractical due to the difficulty in estimating and handling the high-order joint probability distribution P, where all primitives and their relations are taken jointly. Consequently, a strong simplification must be made to allow their use in practical cases. Wong *et al.* [3, 7] proposed the class of *first-order random graphs* for real applications, in which the following assumptions are made:

(i) the random vertices $\{\alpha_i\}$ are mutually independent;
(ii) given values for the random vertices $\{\alpha_i\}$, the random arcs $\{\beta_j\}$ are independent;
(iii) a random arc β_j is independent of any random vertex other than its endpoints α_k and α_l (where β_j is the random variable assigned to the edge ε_{kl} connecting vertices ω_k and ω_l).

Based on the above assumptions, (1) becomes

$$P_R(G', \mu) = \prod_{i=1}^{n} p_i(\mathbf{a}^i) \prod_{j=1}^{m} p_j(\mathbf{b}^j \mid \mathbf{a}^{j1}, \mathbf{a}^{j2}), \tag{2}$$

where $p_i(\mathbf{a}) \stackrel{\Delta}{=} Pr(\alpha_i = \mathbf{a})$, $p_j(\mathbf{b} \mid \mathbf{a}^{j1}, \mathbf{a}^{j2}) \stackrel{\Delta}{=} Pr(\beta_j = \mathbf{b} \mid \alpha_{j1} = \mathbf{a}^{ji}, \alpha_{j2} = \mathbf{a}^{j2})$, and α_{j1}, α_{j2} refer to the random vertices for the endpoints of the random arc β_j. Therefore, only the probability density functions $p_i(\mathbf{a})$, $1 \le i \le n$, and $p_j(\mathbf{b} \mid \mathbf{a}^{j1}, \mathbf{a}^{j2})$, $1 \le j \le m$, must be estimated. Wong *et al.* also introduced an entropy measure for a first-order random graph and used it in the definition of distance and similarity measures for ARGs and random graphs [3, 7].

[4] Without the extension of G to G', the mapping $\mu : G \to R$ would be a monomorphism.

3 Function-described graphs (FDGs)

An important drawback of first-order random graphs, which is due to assumptions (i) and (ii), is that the structural information in a set of sample ARGs is not well preserved in the graph synthesized from them. For example, if C is a set of ARGs describing different perspective views of an object O, many of the outcome graphs of the random graph synthesized from C will represent impossible views of O (just from the topological point of view, without further consideration of the attributes of primitives and relations).

The function-described graphs introduced by Serratosa and Sanfeliu [1, 2] (and redefined next) can be seen as a different type of simplification of the general random graphs, in which both random vertices and arcs are not assumed to be mutually independent, at least with regards to the structural information. On the other hand, the conditional probabilities of the random arcs $p_j(\mathbf{b} \mid \mathbf{a}^{j1}, \mathbf{a}^{j2})$ included in first-order random graphs are hard to estimate, and they are converted into marginal probability density functions $p_j(\mathbf{b})$ in FDGs, but taking into account that an edge cannot occur if any of its two endpoints does not occur, i.e. $p_j(\mathbf{b}) \stackrel{\Delta}{=} Pr(\beta_j = \mathbf{b} \mid \alpha_{j1} \neq \phi, \alpha_{j2} \neq \phi)$. As in [8], the underlying assumption is that the probability of any outcome of a random edge is the same regardless of the actual non-null outcomes of the endpoints.

A *function-described graph* or FDG is formally defined as a tuple $F = (\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon, R_\omega, R_\varepsilon, A_\omega, A_\varepsilon, C_\omega, C_\varepsilon)$, where $(\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon)$ are as in a random graph, so that there is a set of random vertices $\{\alpha_i \mid \alpha_i = \gamma_\omega(\omega_i),\ \omega_i \in \Sigma_\omega,\ 1 \leq i \leq n\}$ with values in $\Delta_\omega = \Delta_v \cup \{\phi\}$ and a set of random arcs $\{\beta_j \mid \beta_j = \gamma_\varepsilon(\varepsilon_{kl}),\ \varepsilon_{kl} \in \Sigma_\varepsilon,\ 1 \leq j \leq m\}$ with values in $\Delta_\varepsilon = \Delta_e \cup \{\phi\}$; R_ω, R_ε are the (marginal) probability density functions for nodes and edges, respectively, i.e. $R_\omega = \{p_i(\mathbf{a}),\ 1 \leq i \leq n\}$ and $R_\varepsilon = \{p_j(\mathbf{b}),\ 1 \leq j \leq m\}$; A_ω, A_ε are the so-called *antagonistic node* and *antagonistic arc functions*, where $A_\omega : \Sigma_\omega \times \Sigma_\omega \rightarrow \{0,1\}$, $A_\omega(\omega_i, \omega_j) = 1 \Leftrightarrow Pr(\alpha_i \neq \phi \wedge \alpha_j \neq \phi) = 0$, $A_\varepsilon : \Sigma_\varepsilon \times \Sigma_\varepsilon \rightarrow \{0,1\}$, $A_\varepsilon(\varepsilon_{kl}, \varepsilon_{pq}) = 1 \Leftrightarrow \gamma_\varepsilon(\varepsilon_{kl}) = \beta_i \wedge \gamma_\varepsilon(\varepsilon_{pq}) = \beta_j \wedge Pr(\beta_i \neq \phi \wedge \beta_j \neq \phi) = 0$; as complementary information, C_ω, C_ε are the *co-occurrent node* and *co-occurrent arc functions*, where $C_\omega : \Sigma_\omega \times \Sigma_\omega \rightarrow \{0,1\}$, $C_\omega(\omega_i, \omega_j) = 1 \Leftrightarrow Pr(\alpha_i \neq \phi \wedge \alpha_j = \phi) = Pr(\alpha_i = \phi \wedge \alpha_j \neq \phi) = 0$, $C_\varepsilon : \Sigma_\varepsilon \times \Sigma_\varepsilon \rightarrow \{0,1\}$, $C_\varepsilon(\varepsilon_{kl}, \varepsilon_{pq}) = 1 \Leftrightarrow \gamma_\varepsilon(\varepsilon_{kl}) = \beta_i \wedge \gamma_\varepsilon(\varepsilon_{pq}) = \beta_j \wedge Pr(\beta_i \neq \phi \wedge \beta_j = \phi) = Pr(\beta_i = \phi \wedge \beta_j \neq \phi) = 0$. In addition, it is assumed that for every arc $\varepsilon_{kl} \in \Sigma_\varepsilon$ (connecting vertices ω_k and ω_l) and its corresponding random variable $\beta_j = \gamma_\varepsilon(\varepsilon_{kl})$, it follows that $Pr(\beta_j \neq \phi \mid \alpha_k = \phi) = Pr(\beta_j \neq \phi \mid \alpha_l = \phi) = 0$, where α_k and α_l are the random variables associated with vertices ω_k and ω_l.

Note that the functions $A_\omega, A_\varepsilon, C_\omega, C_\varepsilon$, which can be represented as symmetric boolean matrices, incorporate structural dependencies between vertices and arcs that can be used as constraints in the processes of ARG generation and recognition. If two nodes of an FDG are antagonistic ($A_\omega(\omega_i, \omega_j) = 1$) then they cannot be instantiated to a non-null vertex at the same time in an outcome graph of the FDG, whereas if they are co-occurrent ($C_\omega(\omega_i, \omega_j) = 1$) then whenever one of them is instantiated to a non-null vertex, the other one must be instantiated too. The same applies to antagonistic and co-occurrent arcs in an FDG. It is likewise assumed that all the constraints in $A_\omega, A_\varepsilon, C_\omega, C_\varepsilon$ are mutually consistent; for instance, it two arcs are antagonistic then the vertices connected by them cannot be co-occurrent.

The probability density functions of random vertices in R_ω can be used, either directly or indirectly, to provide measures of the compatibility between an attributed node of an ARG and a node of the FDG, which can be employed, in turn, in the process of labelling an ARG with respect to the FDG [4]. A similar argument applies to the probability density functions of random arcs in R_ε. In practice, the probability density functions $\{p_i(\mathbf{a}),\ 1 \le i \le n\}$ and $\{p_j(\mathbf{b}),\ 1 \le j \le m\}$ can be represented by discrete approximations (histograms). Furthermore, if the number of vertex attributes t or the number of arc attributes s is greater than 1, then it might be difficult to represent the joint probability of the different attribute values, and an additional simplification might be made by assuming a statistical independence among the different attributes. In such a case, $p_i(\mathbf{a}) = p_i(a_1, \ldots, a_t) = \prod_{k=1}^{t} p_{ik}(a_k)$, where p_{ik} is a marginal probability density function for the k-th attribute of random vertex α_i, and $p_j(\mathbf{b}) = p_j(b_1, \ldots, b_s) = \prod_{k=1}^{s} p_{jk}(b_k)$, where p_{jk} is a probability density function for the k-th attribute of random arc β_j (given non-null endpoints). The mutual independence of attributes was also assumed in [8].

FDGs, as random graphs, are able to describe an ensemble of ARGs. We will see in the following sections how to synthesize an FDG from a set of ARGs that supposedly belong to the same class of structural patterns and can thus be regarded as positive examples of the outcome graphs of an unknown target FDG describing the class. The FDG resulting from the synthesis process must cover the sample ARGs and perform some "plausible" generalization of the examples. As in [3], the synthesis from a set of graphs with commonly labelled vertices and arcs will be discussed firstly and later the synthesis from unlabelled ARGs.

4 Synthesis of FDGs from graphs with a common labelling

4.1 Synthesis of FDGs from labelled ARGs

Let $D = \{G_1, G_2, \ldots, G_z\}$ be a set of ARGs defined over some common homogeneous domains for attributed vertices and arcs, $\Delta_v = \{(a_1, \ldots, a_t) \mid a_i \in D_{vi},\ 1 \le i \le t,\ t \ge 1\}$ and $\Delta_e = \{(b_1, \ldots, b_s) \mid b_i \in D_{ei},\ 1 \le i \le s,\ s \ge 1\}$, respectively. Let $G_i = (\Sigma_{vi}, \Sigma_{ei}, \Delta_v, \Delta_e, \gamma_{vi}, \gamma_{ei})$, for $1 \le i \le z$. Assume that there are given labelling schemes $\Psi_i = (\Psi_{vi} : \Sigma_{vi} \to L_v,\ \Psi_{ei} : \Sigma_{ei} \to L_e),\ i = 1, \ldots, z$, where Ψ_{vi} is an injective mapping from the underlying structural vertex set of G_i to a common set of vertex labels $L_v = \{1, \ldots, n\}$ and Ψ_{ei} similarly labels arcs with labels from $L_e = \{1, \ldots, n(n-1)\}$. The labelling schemes Ψ_i can be extended to bijective mappings $\Psi'_i = (\Psi'_{vi} : \Sigma'_{vi} \to L_v,\ \Psi'_{ei} : \Sigma'_{ei} \to L_e),\ i = 1, \ldots, z$, if each ARG G_i is previously extended to an isomorphic complete graph G'_i of order n. The arc labellings are also assumed to be consistent across all graphs in D, i.e. the arc from the vertex labelled i to the vertex labelled j $(i < j)$ has the same label k in all graphs, e.g. $k = (j - i) + \sum_{f=1}^{i-1} (n - f)$.

An FDG $F = (\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon, R_\omega, R_\varepsilon, A_\omega, A_\varepsilon, C_\omega, C_\varepsilon)$ can be synthesized from D and Ψ' in a straightforward manner. F includes a complete underlying graph structure with a set of n vertices $\Sigma_\omega = \{\omega_1, \ldots, \omega_n\}$ and a set of $n(n-1)$ arcs $\Sigma_\varepsilon = \{\varepsilon_{12}, \ldots, \varepsilon_{(n-1)n}\}$. Each vertex $\omega_i \in \Sigma_\omega$ is associated with a random variable $\alpha_i = \gamma_\omega(\omega_i)$ with values in $\Delta_\omega = \Delta_v \cup \{\phi\}$ and each arc $\varepsilon_{kl} \in \Sigma_\varepsilon$ is associated with a random variable $\beta_j = \gamma_\varepsilon(\varepsilon_{kl})$ with values in $\Delta_\varepsilon = \Delta_e \cup \{\phi\}$. In addition, let $\varphi = (\varphi_\omega : \Sigma_\omega \rightarrow L_v,\ \varphi_\varepsilon : \Sigma_\varepsilon \rightarrow L_e)$ be a labelling scheme on F defined simply by $\varphi_\omega(\omega_i) = i$ and $\varphi_\varepsilon(\varepsilon_{kl}) = (l-k) + \sum_{f=1}^{k-1}(n-f)$, where $k < l$. From labellings Ψ' and φ, a set of bijective mappings $\{\mu_g = (\mu_{vg}, \mu_{eg}) : (\Sigma'_{vg}, \Sigma'_{eg}) \rightarrow (\Sigma_\omega, \Sigma_\varepsilon),\ 1 \leq g \leq z\}$ can be determined such that $\Psi'_{vg} = \varphi_\omega \circ \mu_{vg}$ and $\Psi'_{eg} = \varphi_\varepsilon \circ \mu_{eg}$, for $g = 1, \ldots, z$. Finally, let $v_i^g = \mu_{vg}^{-1}(\omega_i)$ and $e_{j_1 j_2}^g = \mu_{eg}^{-1}(\varepsilon_{j_1 j_2})$ be respectively the node labelled i and the edge labelled j in ARG G'_g.

The probability density functions $R_\omega = \{p_i(\mathbf{a}),\ i = 1, \ldots, n\}$ of individual random vertices and $R_\varepsilon = \{p_j(\mathbf{b}),\ j = 1, \ldots, n(n-1)\}$ of individual random arcs (given non-null endpoints) can be estimated separately, in the maximum likelihood sense, using frequencies of attributes and null values in D. Thus,

$$p_i(\mathbf{a}) = Pr(\alpha_i = \mathbf{a}) = \frac{\#g :\ 1 \leq g \leq z :\ \gamma_{vg}(v_i^g) = \mathbf{a}}{z}, \tag{3}$$

for all possible values $\mathbf{a}$ of α_i including ϕ, and

$$\begin{aligned} p_j(\mathbf{b}) &= Pr(\beta_j = \mathbf{b} \mid \alpha_{j1} \neq \phi, \alpha_{j2} \neq \phi) \\ &= \frac{\#g :\ 1 \leq g \leq z :\ \gamma_{vg}(v_{j1}^g) \neq \phi \wedge \gamma_{vg}(v_{j2}^g) \neq \phi \wedge \gamma_{eg}(e_{j_1 j_2}^g) = \mathbf{b}}{\#g :\ 1 \leq g \leq z :\ \gamma_{vg}(v_{j1}^g) \neq \phi \wedge \gamma_{vg}(v_{j2}^g) \neq \phi} \end{aligned} \tag{4}$$

for all possible values $\mathbf{b}$ of β_j including ϕ.

The antagonistic node function A_ω and the antagonistic arc function A_ε are given by

$$A_\omega(\omega_i, \omega_j) = \begin{cases} 1 \text{ if } \forall g :\ 1 \leq g \leq z :\ \neg\,(\gamma_{vg}(v_i^g) \neq \phi \wedge \gamma_{vg}(v_j^g) \neq \phi) \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

$$A_\varepsilon(\varepsilon_{i_1 i_2}, \varepsilon_{j_1 j_2}) = \begin{cases} 1 \text{ if } \forall g :\ 1 \leq g \leq z :\ \neg\,(\gamma_{eg}(e_{i_1 i_2}^g) \neq \phi \wedge \gamma_{eg}(e_{j_1 j_2}^g) \neq \phi) \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

whereas the co-occurrent node function C_ω and the co-occurrent arc function C_ε are given by

$$C_\omega(\omega_i, \omega_j) = \begin{cases} 1 \text{ if } \forall g :\ 1 \leq g \leq z :\ \neg\,(\gamma_{vg}(v_i^g) \neq \phi \wedge \gamma_{vg}(v_j^g) = \phi \vee \\ \qquad\qquad\qquad\qquad \gamma_{vg}(v_i^g) = \phi \wedge \gamma_{vg}(v_j^g) \neq \phi) \\ 0 \text{ otherwise} \end{cases} \tag{7}$$

$$C_\varepsilon(\varepsilon_{i_1 i_2}, \varepsilon_{j_1 j_2}) = \begin{cases} 1 \text{ if } \forall g :\ 1 \leq g \leq z :\ \neg\,(\gamma_{eg}(e_{i_1 i_2}^g) \neq \phi \wedge \gamma_{eg}(e_{j_1 j_2}^g) = \phi \vee \\ \qquad\qquad\qquad\qquad \gamma_{eg}(e_{i_1 i_2}^g) = \phi \wedge \gamma_{eg}(e_{j_1 j_2}^g) \neq \phi) \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

4.2 Synthesis of FDGs from labelled FDGs

Let $D = \{F_1, F_2, \ldots, F_h\}$ be a set of FDGs independently synthesized from disjoint subsets of a class of ARGs with common homogeneous domains for attributed vertices and arcs. Let $F_k = (\Sigma_{\omega k}, \Sigma_{\varepsilon k}, \Delta_\omega, \Delta_\varepsilon, \gamma_{\omega k}, \gamma_{\varepsilon k}, R_{\omega k}, R_{\varepsilon k}, A_{\omega k}, A_{\varepsilon k}, C_{\omega k}, C_{\varepsilon k})$, for $1 \leq k \leq h$. For each FDG F_k, the number of ARGs from which it was formed, z_k, is stored together with the value of the denominator in Eq.(4), let us call it u_j^k, for each random arc β_j^k of F_k.

Assume that there are given labelling schemes $\Psi_k = (\Psi_{\omega k} : \Sigma_{\omega k} \rightarrow L_\omega,\ \Psi_{\varepsilon k} : \Sigma_{\varepsilon k} \rightarrow L_\varepsilon),\ k = 1, \ldots, h$, mapping the vertices and arcs of the FDGs F_k into common label sets $L_\omega = \{1, \ldots, n\}$ and $L_\varepsilon = \{1, \ldots, n(n-1)\}$, such that all $\Psi_{\omega k}$ and $\Psi_{\varepsilon k}$ are injective and all arc labellings are consistent throughout the set D. If the order of some FDG F_k is less than n, then F_k can be extended to an isomorphic complete FDG F'_k of order n by adding null vertices and arcs: if ω_i is an added null vertex, $p_i(\phi) = 1$, $A_\omega(\omega_i, \omega_j) = 1$ for all vertices ω_j, $C_\omega(\omega_i, \omega_j) = 1$ if ω_j is a null vertex and $C_\omega(\omega_i, \omega_j) = 0$ otherwise; similar definitions are applied in the case of an added null arc. Therefore, the labelling schemes Ψ_k can be extended to bijective mappings $\Psi'_k = (\Psi'_{\omega k} : \Sigma'_{\omega k} \rightarrow L_\omega,\ \Psi'_{\varepsilon k} : \Sigma'_{\varepsilon k} \rightarrow L_\varepsilon),\ k = 1, \ldots, h$, whenever each FDG F_k is previously extended to an isomorphic complete FDG F'_k of order n.

An FDG $F = (\Sigma_\omega, \Sigma_\varepsilon, \Delta_\omega, \Delta_\varepsilon, \gamma_\omega, \gamma_\varepsilon, R_\omega, R_\varepsilon, A_\omega, A_\varepsilon, C_\omega, C_\varepsilon)$ can be synthesized now from the FDGs F'_k and the common labelling Ψ' as follows. F includes a complete underlying graph structure with a set of n vertices $\Sigma_\omega = \{\omega_1, \ldots, \omega_n\}$ with corresponding random variables $\{\alpha_i,\ 1 \leq i \leq n\}$, and a set of $n(n-1)$ arcs $\Sigma_\varepsilon = \{\varepsilon_{12}, \ldots, \varepsilon_{(n-1)n}\}$ with corresponding random variables $\{\beta_j,\ 1 \leq j \leq n(n-1)\}$. As before, let $\varphi = (\varphi_\omega : \Sigma_\omega \rightarrow L_\omega,\ \varphi_\varepsilon : \Sigma_\varepsilon \rightarrow L_\varepsilon)$ be a labelling scheme on F given by $\varphi_\omega(\omega_i) = i$ and $\varphi_\varepsilon(\varepsilon_{kl}) = (l-k) + \sum_{f=1}^{k-1}(n-f)$, $k < l$; and let $\{\mu_k = (\mu_{\omega k}, \mu_{\varepsilon k}) : (\Sigma'_{\omega k}, \Sigma'_{\varepsilon k}) \rightarrow (\Sigma_\omega, \Sigma_\varepsilon),\ 1 \leq k \leq h\}$ be a set of bijective mappings such that $\Psi'_{\omega k} = \varphi_\omega \circ \mu_{\omega k}$ and $\Psi'_{\varepsilon k} = \varphi_\varepsilon \circ \mu_{\varepsilon k}$, for $k = 1, \ldots, h$. Finally, let $\omega_i^k = \mu_{\omega k}^{-1}(\omega_i)$ and $\varepsilon_{j_1 j_2}^k = \mu_{\varepsilon k}^{-1}(\varepsilon_{j_1 j_2})$ be respectively the node labelled i and the edge labelled j in FDG F'_k.

The probability density functions $R_\omega = \{p_i(\mathbf{a}),\ i = 1, \ldots, n\}$ of individual random vertices and $R_\varepsilon = \{p_j(\mathbf{b}),\ j = 1, \ldots, n(n-1)\}$ of individual random arcs can be estimated separately, again in the maximum likelihood sense, using the corresponding probabilities in $R_{\omega k}, R_{\varepsilon k}$, together with the values z_k and u_j^k, $k = 1, \ldots, h$. Let

$$q_k = \frac{z_k}{\sum_{g=1}^{h} z_g}, \quad 1 \leq k \leq h, \text{ and} \tag{9}$$

$$r_j^k = \frac{u_j^k}{\sum_{g=1}^{h} u_j^g}, \quad 1 \leq k \leq h, \ \ 1 \leq j \leq n(n-1). \tag{10}$$

Then, for all possible values $\mathbf{a}$ of random vertex α_i including ϕ, we have that

$$p_i(\mathbf{a}) = Pr(\alpha_i = \mathbf{a}) = \sum_{k=1}^{h} q_k \ p_i^k(\mathbf{a}), \tag{11}$$

and, for all possible values $\mathbf{b}$ of random arc β_j including ϕ,

$$p_j(\mathbf{b}) = Pr(\beta_j = \mathbf{b} \mid \alpha_{j1} \neq \phi, \alpha_{j2} \neq \phi) = \sum_{k=1}^{h} r_j^k \; p_j^k(\mathbf{b}). \tag{12}$$

The antagonistic node and arc functions A_ω, A_ε, as well as the co-occurrent node and arc functions C_ω, C_ε, are all readily calculated, since they are given by the logical *and* of the corresponding functions in the FDGs F_k:

$$A_\omega(\omega_i, \omega_j) = \bigwedge_{k=1}^{h} A_{\omega k}(\omega_i^k, \omega_j^k), \tag{13}$$

$$A_\varepsilon(\varepsilon_{i_1 i_2}, \varepsilon_{j_1 j_2}) = \bigwedge_{k=1}^{h} A_{\varepsilon k}(\varepsilon_{i_1 i_2}^k, \varepsilon_{j_1 j_2}^k), \tag{14}$$

$$C_\omega(\omega_i, \omega_j) = \bigwedge_{k=1}^{h} C_{\omega k}(\omega_i^k, \omega_j^k), \tag{15}$$

$$C_\varepsilon(\varepsilon_{i_1 i_2}, \varepsilon_{j_1 j_2}) = \bigwedge_{k=1}^{h} C_{\varepsilon k}(\varepsilon_{i_1 i_2}^k, \varepsilon_{j_1 j_2}^k). \tag{16}$$

5 Synthesis of FDGs from unlabelled ARGs

Algorithm 1 shows the incremental method that is proposed to synthesize an FDG from a sequence of ARGs without an a-priori common labelling. The algorithm uses the procedure described in Section 4.1 to transform an ARG G into an equivalent FDG H whose only outcome graph is G, as well as the procedure described in Section 4.2 to synthesize an FDG from two FDGs for which a common labelling is known. The synthesis method relies on (and is parametrized by) a matching algorithm $\mathcal{M}(G, F)$ that is supposed to return an optimal labelling between an ARG G and an FDG F according to some given optimization criterion. Several such algorithms $\mathcal{M}$ may be used. For instance, a method $\mathcal{M}_1$ is defined by minimizing the increment of entropy resulting from the merge of F and (an FDG equivalent to) G [3, 8]; note that an entropy measure, reflecting the variability of outcome graphs, can be computed for any FDG based on the probability density functions in R_ω and R_ε similarly to the case of random graphs [7]. An alternative method $\mathcal{M}_2$ is defined by maximizing a matching quality using a relaxation labelling scheme with a support function that involves both local attributes and contextual information [4].

Algorithm 1: *Incremental FDG synthesis from unlabelled ARGs*
Inputs: A sequence of ARGs $G_1, \ldots, G_m$ ($m \geq 1$) belonging to the same class of patterns.
An ARG-to-FDG matching algorithm $\mathcal{M}(G, F)$ that finds an "optimal" labelling according to some optimization criterion.

Output: An FDG F_m that generates the ARGs $G_1, \ldots, G_m$ and other "similar" ARGs as outcome graphs.
begin
$i := 1$ { i counts the number of ARGs in the sequence }
read_ARG(G_1) { read the first ARG in the sequence}
F_1 := FDG_synthesis_from_labelled_ARGs(G_1) { build the first FDG from G_1 using the synthesis procedure described in Section 4.1 with $z = 1$ }
while not end of sequence of ARGs **do**
  $i := i + 1$; read_ARG(G_i) { read the next ARG }
  let $\mu_i : G_i \rightarrow F_{i-1}$ be the optimal labelling found by $\mathcal{M}(G_i, F_{i-1})$
  if μ_i is not bijective **then**
    let $\mu_i' : G_i' \rightarrow F_{i-1}'$ be a bijective mapping that extends μ_i by extending G_i to G_i' and F_{i-1} to F_{i-1}' with null vertices and arcs appropriately
  else
    $\mu_i' := \mu_i$; $G_i' := G_i$; $F_{i-1}' := F_{i-1}$
  end_if
  H_i := FDG_synthesis_from_labelled_ARGs(G_i') { build an FDG from G_i' using the synthesis procedure described in Section 4.1 with $z = 1$ }
  let $\varphi_i : G_i' \rightarrow H_i$ be the bijective mapping constructed in the previous synthesis
  let $\psi_i : H_i \rightarrow F_{i-1}'$ be the bijective mapping determined by the composition $\mu_i' = \psi_i \circ \varphi_i$
  F_i := FDG_synthesis_from_labelled_FDGs(H_i, F_{i-1}', ψ_i) { using the synthesis procedure described in Section 4.2 with $h = 2$ and a common labelling based on ψ_i }
end_while
end_algorithm

A drawback of the above incremental approach is that, given a set of unlabelled ARGs $\{G_1, \ldots, G_m\}$, different FDGs could be synthesized from them depending on the order of presentation. To infer a unique FDG, a hierarchical synthesis process could be followed by merging successively pairs of FDGs with minimal distance, as in [3], whenever a distance measure between FDGs could be computed. Such a distance measure may be given by the minimal increment of entropy yielded by the synthesis of two FDGs (similarly to random graphs [3]) or derived from the matching quality of an optimal labelling found through relaxation [4].

6 Conclusions

FDGs are a type of compact representation of a set of ARGs that borrow from random graphs the capability of probabilistic modelling of structural and attribute information, while improving the capacity of first-order random graphs to record structural relationships that consistently appear through the data. The synthesis of an FDG from a set of commonly labelled graphs, either ARGs or

FDGs, has been described precisely. Then, a general incremental algorithm to synthesize an FDG from a sequence of unlabelled ARGs has been proposed (assuming that the input graphs constitute a sample of a single class of patterns). Specific synthesis methods are obtained when an ARG-to-FDG matching algorithm is selected to find optimal labellings. An efficient matching method based on relaxation labelling that uses a new support function is reported elsewhere [4], which may be applied in the synthesis process. Both non-incremental synthesis and clustering of ARGs using FDGs may be approached by a hierarchical process that requires a distance measure between FDGs, but these problems have not been discussed enough here because they need further study.

References

1. F. Serratosa and A. Sanfeliu: Function-described graphs. In *Pattern Recognition and Image Analysis*, Preprints of the 7th National Symposium on Pattern Recognition and Image Analysis, Centre de Visió per Computador, Univ. Autònoma de Barcelona, Vol.I (1997) 37-42
2. F. Serratosa and A. Sanfeliu: Function-described graphs applied to 3D object recognition. In *Proc. ICIAP'97, 9th Int. Conf. Image Analysis and Processing*, Firenze, Italy, Vol.I (1997) 701-708
3. A.K.C. Wong, J. Constant and M.L. You: Random graphs. In *Syntactic and Structural Pattern Recognition: Theory and Applications*, H.Bunke and A.Sanfeliu (eds.), World Scientific Publishing Co. Pte. Ltd., Singapore (1990) 197-234
4. F. Serratosa, A. Sanfeliu and R. Alquézar: Distance and matching between function-described graphs. *Tech. Rep. IRI-DT-9803*, Inst. de Robòtica i Informàtica Industrial, Univ. Politècnica de Catalunya - CSIC, Barcelona, Spain (1998)
5. W.H. Tsai and K.S. Fu: Error-correcting isomorphism of attributed relational graphs for pattern analysis. *IEEE Trans. Systems, Man and Cybernetics* **9** (1979) 757-768
6. A. Sanfeliu and K.S. Fu: A distance measure between attributed relational graphs for pattern recognition. *IEEE Trans. Systems, Man and Cybernetics* **13** (1983) 353-362
7. A.K.C. Wong and M.L. You: Entropy and distance of random graphs with application to structural pattern recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence* **7** (1985) 599-609
8. D.S. Seong, H.S. Kim and K.H. Park: Incremental clustering of attributed graphs. *IEEE Trans. Systems, Man and Cybernetics* **23** (1993) 1399-1411

This article was processed using the LaTeX macro package with LLNCS style