RAIRO. INFORMATIQUE THÉORIQUE

D. JANSSENS
G. ROZENBERG

Hypergraph systems and their extensions

RAIRO. Informatique théorique, tome 17, nº 2 (1983), p. 163-196
<http://www.numdam.org/item?id=ITA_1983__17_2_163_0>

R.A.I.R.O. Informatique théorique/Theoretical Informatics
(vol. 17, n° 2, 1983, p. 163 à 196)

HYPERGRAPH SYSTEMS AND THEIR EXTENSIONS (*)

by D. JANSSENS [1] and G. ROZENBERG [2]

Communicated by J. BERSTEL

Abstract. — *The notion of a graph is a natural generalization of the notion of a string. Using* one *graph (the transition graph of a finite automaton) one can define a language (the language of a given automaton). This well-known idea is generalized as follows: the notion of a hypergraph is a natural generalization of the notion of a graph. Using* one *hypergraph (equipped with an additional graph structure) one can define a graph language (a set of graphs). Several variants of graph grammars based on this idea are introduced in this paper together with formalism needed to investigate them and illustrating examples. The generating power of various graph-language-generating systems is compared.*

Résumé. — *La notion de graphe est une généralisation naturelle de la notion de mot. En utilisant un graphe (le graphe de transition d'un automate fini) on définit un langage (le langage accepté par l'automate). Cette idée est généralisée comme suit : la notion de hypergraphe est une généralisation naturelle de la notion de graphe. Avec un hypergraphe (muni d'une structure de graphe supplémentaire) on peut définir un langage (un ensemble) de graphes. Dans cet article on introduit diverses variantes des grammaires de graphes basées sur cette idée; on donne le formalisme nécessaire pour les étudier, et on les illustre par des exemples. La puissance générative de divers systèmes engendrant des langages de graphes est comparée.*

INTRODUCTION

As documented e. g. in [4] and [1] (in particular in [2] and [3]) the theory of graph grammars is a well-motivated research area. However, this theory is much poorer than that of the classical "string" grammars. This is due not only to the fact that the subject is intrinsically more difficult (a graph is a more complicated structure than a string) but also to the fact that the number of people working in this area is considerably smaller than the number of people working on string grammars.

In the present state of the theory, new approaches to defining graph languages are still needed (as well as in depth research of old approaches). In particular, we do not have yet the class of graph grammars (and languages) which would correspond to finite automata (and regular languages). What we mean by this

(*) Received December 1981, revised April 1982.
(1) Department of Mathematics, University of Antwerp, U.I.A., 2610 Wilrijk, Belgium.
(2) Institute of Applied Mathematics and Computer Science, University of Leiden, Leiden, The

correspondence is that (1) one would like to have a very "natural" device to generate nontrivial languages (as a finite automaton is), and (2) one would like to have a class of graph languages that would be as essential for the theory of graph languages as the regular languages are for string languages.

In this paper we present an attempt that could provide a solution of (1) above. Our approach is methodologically quite analogous to that of defining languages by transition graphs (of finite automata). The notion of a graph is a natural generalization of the notion of a string and in finite automata theory one uses *one* (transition) graph to define a set of strings (its language). The notion of a hypergraph generalizes the notion of a graph. In our approach we will use *one* hypergraph (equipped with an additional graph structure) to define a set of graphs (its language).

The aim of this paper is to introduce some basic notions and formalisms concerning our approach, to illustrate it by examples and to compare several classes of graph-generating systems that we introduce. A basic notion concerning hypergraphs is that of the intersection of its "blocks" (edges); the generation of graphs in our system is nothing els but gluing given "elementary" graphs (blocks equipped with graph structure) to each other in a way controlled by the hypergraph structure. In this way the operation of gluing of graphs becomes very central for our paper. This operation was used already before in graph grammars (*see* e. g. [2] and [3] in [1]). However it seems to us that the basic idea of the "hypergraph systems", considered in this paper, is new.

I. PRELIMINARIES

In this section we recall some basic terminology and notation to be used in this paper.

(1) Let X be a set. Then $\mathscr{P}(X)$ denotes the set of all subsets of X and Id_X denotes the identity relation on X.

(2) For a function g from A into B, $\mathrm{Im}(g)$ denotes the range of g.

(3) Let A, B, C be sets and let f, g be functions from A into B and from B into C respectively. Then by $g \circ f$ we denote the composition of f and g (first f and then g).

(4) An *unlabelled graph*, in the sequel simply called "*graph*", is a pair $H=(V, E)$ where V is a finite nonempty set, called the *set of nodes* of H, and E is a set of multisets of two elements from V; E is called the *set of edges* of H. If H is a graph, then by V_H and E_H we will denote the set of nodes and the set of edges of H respectively.

(5) If H and M are graphs, then H is a *subgraph* of M if $V_H \subseteq V_M$ and $E_H \subseteq E_M$. H is a *full subgraph* of M if H is a subgraph of M and:

$$E_H = \{\{x, y\} \mid x, y \in V_H \text{ and } \{x, y\} \in E_M\}.$$

(6) Let $X = (V_X, E_X)$ be a graph and let Y be a subset of V_X. Then X_Y denotes the full subgraph of X with node set Y.

(7) Graphs X and Y are called *disjoint* if $V_X \cap V_Y = \emptyset$.

(8) The *degree* of a graph X is the maximal number of edges, incident to one node of X.

(9) Let H and M be graphs. A function h from V_H into V_M is called a (*graph-*) *homomorphism* from H into M if $\{\{h(x), h(y)\} \mid \{x, y\} \in E_H\} \subseteq E_M$. h is an *isomorphism* from H onto M if h is bijective homomorphism from V_H onto V_M and h^{-1} is a bijective homomorphism from V_M onto V_H. If there exists an isomorphism from H into M then we say that H is *isomorphic to M*.

(10) A graph X is called *discrete* if $E_X = \emptyset$. If X is a discrete graph and Y is a graph, then every function from V_X into V_Y is a graph homomorphism from X into Y.

(11) A *hypergraph* is a system $H = (V, E, f)$ where V is a finite nonempty set, called the set of *nodes* of H, E is a finite set, called the set of *edges* of H and f is an injective function from E into $\mathscr{P}(V)$ such that $\bigcup_{e \in E} f(e) = V$; f is called the *edge function* of H. If H is a hypergraph, then the set of nodes of H, the set of edges of H and the edge function of H will be denoted by V_H, E_H and f_H respectively.

(12) Let $H = (V, E, f)$ be a hypergraph. By *int* H we denote the set $\{X \mid X \neq \emptyset$ and there exist distinct edges $e, \bar{e}$ in E such that $f(e) \cap f(\bar{e}) = X\}$ (*int* H is the *set of intersections* of H).

(13) Let G be a graph-generating system. Them $L(G)$ denotes the language generated by the system G. Two graph-generating systems G and $\bar{G}$ are called *equivalent* if $L(G) = L(\bar{G})$.

(14) In the following, if X denotes a class of graph-generating systems (e. g. H systems, FDH systems, etc.), then by $\mathscr{L}(X)$ we will denote the set of languages L for which there exists a X system G such that $L = L(G)$.

Gluing graphs is the very basic operation used in our paper. Formally it is defined as follows.

DEFINITION 1.1 : Let A, B and H be graphs and let I be a discrete graph. Let f and g be injective homomorphisms from I into A and into B respectively. Then we say that H is the *gluing of A and B along I by f and g* if H is isomorphic to the graph (V, E) constructed as follows.

Let $\bar{A}$ and $\bar{B}$ be graphs, isomorphic to A and B such that $V_{\bar{A}} \cap V_{\bar{B}} = \emptyset$ and let h_A and h_B be the corresponding isomorphisms from A into $\bar{A}$ and from B into $\bar{B}$ respectively. Let $\bar{f} = h_A \circ f$ and $\bar{g} = h_B \circ g$. Then define:

$$V = V_{\bar{A}} \cup (V_{\bar{B}} \setminus \mathrm{Im}(\bar{g})),$$

$$\begin{aligned} E = E_{\bar{A}} &\cup \{\{x, y\} \mid \{x, y\} \in E_{\bar{B}} \text{ and } x, y \notin \mathrm{Im}(\bar{g})\} \\ &\cup \{\{\bar{f} \circ \bar{g}^{-1}(x), y\} \mid \{x, y\} \in E_{\bar{B}} \text{ and } x \in \mathrm{Im}(\bar{g}), y \notin \mathrm{Im}(\bar{g})\} \\ &\cup \{\{\bar{f} \circ \bar{g}^{-1}(x), \bar{f} \circ \bar{g}^{-1}(y)\} \mid \{x, y\} \in E_{\bar{B}} \text{ and } x, y \in \mathrm{Im}(\bar{g})\}. \end{aligned}$$

Let h be an isomorphism from the graph (V, E) into H, let $\hat{f} = h \circ h_A$, and let $\hat{g}$ be defined by $\hat{g}(x) = h \circ h_B$ for $x \in V_B \setminus \mathrm{Im}(g)$ and $\hat{g}(x) = \hat{f} \circ f \circ g^{-1}(x)$ for $x \in \mathrm{Im}(g)$. Then $\hat{f}$ and $\hat{g}$ are isomorphisms from A and B respectively into subgraphs of H. $\hat{f}$ and $\hat{g}$ will be called the *natural injections* of A into H and B into H respectively. □

The gluing operation is illustrated by example 1.1.

Example 1.1: Figure 1.1 shows two graphs A and B, together with a discrete graph I.

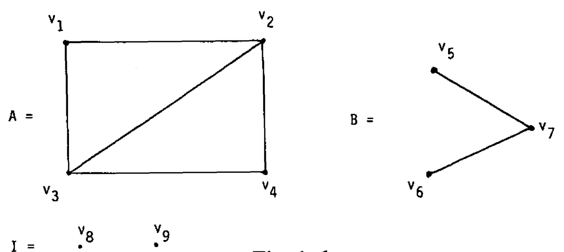


Fig. 1.1

Let f and g be injective homomorphisms from I into A and B respectively, defined by $f(v_8) = v_2$, $f(v_9) = v_4$, $g(v_8) = v_5$ and $g(v_9) = v_6$.

H is the gluing of A and B along I by f and g if H is isomorphic to the graph (V, E) depicted in figure 1.2.

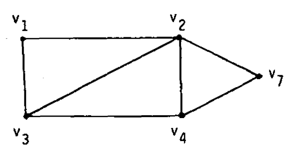


Figure 1.2: The graph (V, E)

In the rest of this paper we will frequently use the set of nodes of I instead of the graph I itself. The homomorphisms f and g of the above definition are then simply injective functions from a finite nonempty set into V_A and V_B.

REMARK 1.1: The gluing of two graphs along a third one is a classical operation in graph grammars (*see*, e. g., [2]). An alternative way of defining this operation is to use the pushout construction as discussed in [2]. We use a somewhat different notion of homomorphism. However one can easily see that the set of graphs we consider together with the set of graph homomorphisms we consider form a cathegory. Then the pushouts in this cathegory (defined analogously to [2], *see* fig. 1.3) correspond to the construction of definition 1.1.

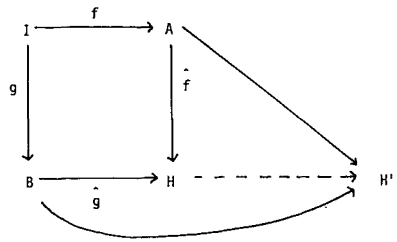


Fig. 1.3.

The following technical notion will be very useful in the sequel.

DEFINITION 1.2: If $\mathscr{H}$ is a set of graphs, then a *trace over* $\mathscr{H}$ is a triple (A, H, g) such that A is a graph, $H \in \mathscr{H}$ and g is an injective homomorphism from H into A. □

II. HYPERGRAPH SYSTEMS

In this section a grammatical device to define graph languages is introduced – it is very basic for this paper. It is based on hypergraphs. Given a hypergraph one first imposes on its node set an additional graph structure. Then one uses edges of such a hypergraph as elementary blocks (letters) to build graphs. The way that these elementary blocks are glued together is controlled through the structure of the given hypergraph (the way its edges intersect). Formally such a construct is defined as follows.

DEFINITION 2.1: A *hypergraph system* (abbreviated H *system*) is a system $G=(H, \Gamma, e_{\text{in}})$ where H is a hypergraph, Γ is a set of multisets of the form $\{x, y\}$ with $x, y \in V_H$ and e_{in} is an element of E_H; e_{in} is called the *initial edge* of G. □

Note that (V_H, Γ) is a graph. It is called the *underlying graph* of the H system and is denoted by *und* G.

For $e \in E_H$, $(und\ G)_e$ denotes the full subgraph $(und\ G)_{f_H(e)}$ of *und* G. □

DEFINITION 2.2: Let $G=(H, \Gamma, e_{in})$ be an H system. (A, e, g) is a *trace for G* if $(A, (und\ G)_e, g)$ is a trace over $\mathscr{H}=\{(und\ G)_2 \mid e \in E_H\}$. □

DEFINITION 2.3: Let $G=(H, \Gamma, e_{in})$ be an H system and let (A, e, g), $(B, \tilde{e}, \tilde{g})$ be traces for G. We say that (A, e, g) *directly derives* $(B, \tilde{e}, \tilde{g})$ *in* G, denoted $(A, e, g) \underset{G}{\vdash} (B, \tilde{e}, \tilde{g})$, if:

(1) $u=f_H(e) \cap f_H(\tilde{e}) \neq \emptyset$,

(2.1) B is the gluing of A and $(und\ G)_{\tilde{e}}$ along $(und\ G)_u$ by α and β, where α is the restriction of g to u and β equals Id_u, and:

(2.2) $\tilde{g}$ is the natural injection of $(und\ G)_{\tilde{e}}$ into B.

Then $\overset{*}{\underset{G}{\vdash}}$ *denotes the transitive and the reflexive closure of the relation* $\underset{G}{\vdash}$. *If* $(A, e, g) \overset{*}{\underset{G}{\vdash}} (B, \tilde{e}, \tilde{g})$ the we say that (A, e, g) *derives* $(B, \tilde{e}, \tilde{g})$ *in* G. □

DEFINITION 2.4: The *language* of the H system $G=(H, \Gamma, e_{in})$, denoted $L(G)$, is defined by:

$$L(G)=\{M \mid ((und\ G)_{e_{in}}, e_{in}, Id_{f_H(e_{in})}) \overset{*}{\underset{G}{\vdash}} (M, e, g) \text{ where } (M, e, g) \text{ is a trace for } G\}. \quad \square$$

REMARK 2.1: Let $G=(H, \Gamma, e_{in})$ be a H system. From the previous definitions it is obvious that elements $\{x, y\}$ of Γ for which there exists no $e \in E_H$ with both $x \in f_H(e)$ and $y \in f_H(e)$ have no influence on the language defined by G. Therefore, from now on we assume that Γ does not contain such edges. □

Example 2.1: Let $G=(H, \Gamma, e_{in})$ where:

$$H=(\{v_1, v_2, v_3, v_4, v_5, v_6\}, \{e_1, e_2, e_3\}, f)$$

with:

$$f(e_1)=\{v_1, v_2\}, \qquad f(e_2)=\{v_2, v_3, v_4\}$$

and:

$$f(e_3)=\{v_3, v_4, v_5, v_6\},$$

$$\Gamma=\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_2\}, \{v_4, v_5\}, \{v_5, v_6\}, \{v_6, v_3\}\} \quad \text{and} \quad e_{in}=e_1.$$

G is depicted in figure 2.1.

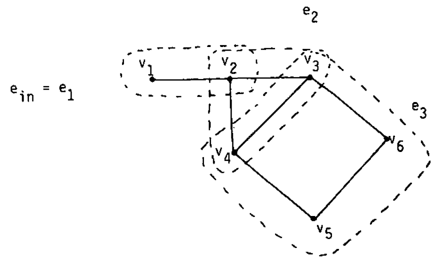


Figure 2.1.

A description of the derivation:

$$((und\ G)_{e_{\text{in}}}, e_{\text{in}}, \text{Id}_{f(e_{\text{in}})}) \underset{G}{\vdash} (M^{(1)}, e_2, g^{(1)}) \underset{G}{\vdash} (M^{(2)}, e_3, g^{(2)}) \underset{G}{\vdash} (M^{(3)}, e_2, g^{(3)})$$

is given in figure 2.2.

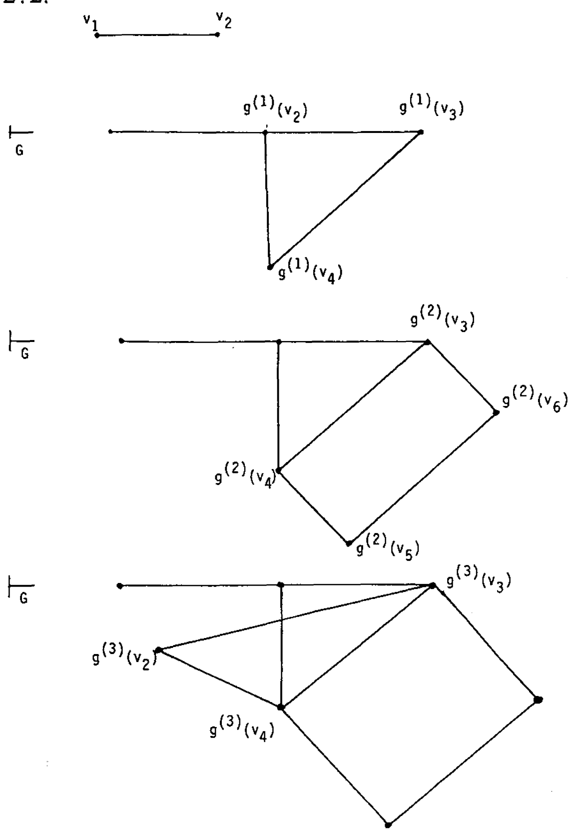


Figure 2.2.

REMARK 2.2: Observe that in performing a derivation step in a H system (and in all systems to be considered in this paper later on) the number of nodes, as well as the number of edges, is never decreased.

When deriving a graph in the language of a hypergraph system one uses the edges of the "underlying hypergraph" as building blocks. The way that those building blocks are glued together is determined by the way that edges in the hypergraph intersect. Hence it is natural to consider a system also based on a hypergraph, in which rather than to follow "consecutive" edges and glue them according to their intersections, one "follows" the intersections themselves.

DEFINITION 2.5 : An *intersection-based Hypergraph system* (abbreviated IH *system*) is a system $G=(H, \Gamma, u_{in})$ where H is a hypergraph, Γ is a set of multisets of the form $\{x, y\}$ with $x, y \in V_H$ and u_{in} is an element of *int* H; u_{in} is called the *initial intersection* of G. □

The graph (V_H, Γ) is denoted by *und* G.

DEFINITION 2.6: Let $G=(H, \Gamma, u_{in})$ be an IH system. A system (A, e, u, g) is an *extended trace for* G if (A, e, g) is a trace over $\{(\text{und } G)_e \mid e \in E_H\}$ and u is an element of *int* H such that there exists an $\bar{e}$ in E_H with $e \neq \bar{e}$ and $f_H(e) \cap f_H(\bar{e}) = u$. □

DEFINITION 2.7: Let $G=(H, \Gamma, u_{in})$ be an IH system and let (A, e, u, g) and $(B, \tilde{e}, \tilde{u}, \tilde{g})$ be extended traces for G. We say that (A, e, u, g) *directly derives* $(B, \tilde{e}, \tilde{u}, \tilde{g})$, denoted $(A, e, u, g) \underset{G}{\vdash} (B, \tilde{e}, \tilde{u}, \tilde{g})$, if:

(1) $f_H(e) \cap f_H(\tilde{e}) = u$,

(2.1) B is the gluing of A and $(\textit{und } G)_{\tilde{e}}$ along u by α and Id_u where α is the restriction of g to u, and:

(2.2) $\tilde{g}$ is the natural injection of $(\textit{und } G)_{\tilde{e}}$ in B.

The relation $\overset{*}{\underset{G}{\vdash}}$ is defined to be the transitive and the reflexive closure of $\underset{G}{\vdash}$. If $(A, e, u, g) \overset{*}{\underset{G}{\vdash}} (B, \tilde{e}, \tilde{u}, \tilde{g})$, then we say that (A, e, u, g) *derives* $(B, \tilde{e}, \tilde{u}, \tilde{g})$ *in* G. □

DEFINITION 2.8: Let $G=(H, \Gamma, u_{in})$ be an IH system. The *language of* G, denoted $L(G)$, is defined by:

$$L(G) = \{ M \mid ((\textit{und } G)_e, e, u_{in}, \mathrm{Id}_{f_H(e)}) \overset{*}{\underset{G}{\vdash}} (M, \tilde{e}, u, g)$$

where $((\textit{und } G)_e, e, u_{in}, \mathrm{Id}_{f_H(e)})$ and $(M, \tilde{e}, u, g)$ are extended traces for $G\}$. □

Example 2.2: Let $G=(H, \Gamma, u_{in})$ where:

$$H=(\{v_1, v_2, v_3, v_4, v_5\}, \{e_1, e_2, e_3\}, f)$$

with:

$$f(e_1)=\{v_1, v_2\},$$

$$f(e_2)=\{v_1, v_3, v_4\} \quad \text{and} \quad f(e_3)=\{v_2, v_4, v_5\},$$

$$\Gamma=\{\{v_1, v_2\}, \{v_1, v_3\}, \{v_3, v_4\}, \{v_4, v_1\},$$

$$\{v_2, v_4\}, \{v_2, v_5\}, \{v_4, v_5\}\} \quad \text{and} \quad u_{in}=\{v_1\}.$$

G is depicted in figure 2.3.

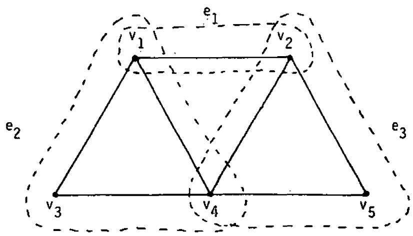


Figure 2.3.

A description of the derivation:

$$((und\ G)_{e_1}, e_1, u_{in}, \mathrm{Id}_{f(e_1)}) \underset{G}{\vdash} (M^{(1)}, e_2, \{v_4\}, g^{(1)})$$

$$\underset{G}{\vdash} (M^{(2)}, e_3, \{v_4\}, g^{(2)}) \underset{G}{\vdash} (M^{(3)}, e_2, \{v_1\}, g^{(3)})$$

is depicted in figure 2.4.

REMARK 2.3: Our proofs will be presented somewhat informally. Since the formalism of graph-rewriting systems we consider in this paper is rather involved (a situation common to practically all graph grammars considered in the literature), in this way (we hope) our proofs are more readable. We hope that our proofs are rigorous enough so that if necessary the reader can complete them to very formal (and tedious) proofs.

LEMMA 2.1: $\mathscr{L}(\mathrm{IH}) \setminus \mathscr{L}(\mathrm{H}) \neq \emptyset$.

Proof: Let G be an IH system, $G=(H, \Gamma, u_{in})$ where $H=(\{v_1, v_2, v_3\}, \{e_1, e_2\}, f)$ with $f(e_1)=\{v_1, v_2\}$ and $f(e_2)=\{v_2, v_3\}$, and where $\Gamma=\{\{v_1, v_2\}\}$ and $u_{in}=\{v_2\}$.

Then clearly $L(G)$ contains two graphs with only two nodes; •——• and • •, and no graphs with only one node. If $\bar{G}=(\bar{H}, \bar{\Gamma}, \bar{e}_{in})$ is an H system with

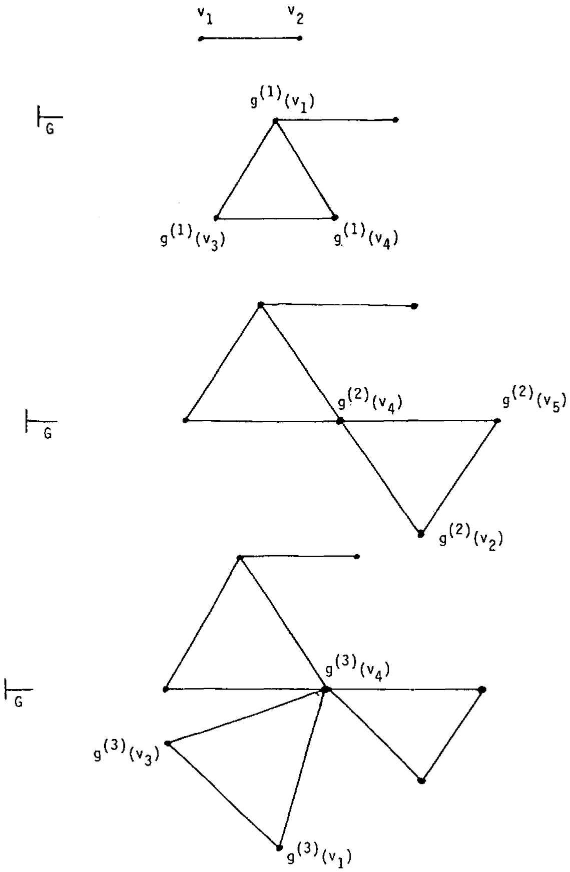


Figure 2.4.

$L(\overline{G})=L(G)$ then it is clear that $(und\ \overline{G})_{\bar{e}_{\text{in}}}$ must be of the form •   • This leads to a contradiction because the graph •—• cannot be obtained by gluing $(und\ \overline{G})_{\bar{e}}$ and $(und\ \overline{G})_{\bar{e}_{\text{in}}}$ for any $\bar{e}\in E_{\overline{H}}$, because $(und\ \overline{G})_{\bar{e}_{\text{in}}}$ is a full subgraph of $und\ \overline{G}$. □

LEMMA 2.2: $\mathscr{L}(\mathrm{H})\setminus\mathscr{L}(\mathrm{IH})\neq\emptyset$.

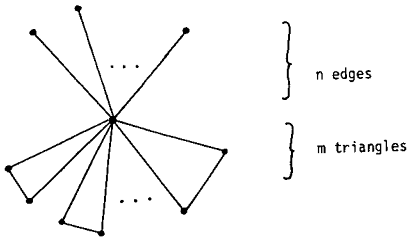
Proof: Let G be the H system $(H, \Gamma, e_{\text{in}})$ where:

$$H=(\{v_1, v_2, v_3, v_4\}, \{e_1, e_2\}, f_H)$$

with:

$$f_H(e_1)=\{v_1, v_2\} \qquad \text{and} \qquad f_H(e_2)=\{v_2, v_3, v_4\},$$

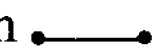
and where $\Gamma=\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_2\}\}$ and $e_{\text{in}}=e_1$. Then the language $L(G)$ is the set of graphs of the form $(*)$



with $n \geqq 1$ and either $m=n-1$ or $m=n$.

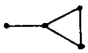
Now assume $\bar{G}=(\bar{H}, \bar{\Gamma}, \bar{u}_{\text{in}})$ is an IH system with $L(\bar{G})=L(G)$.

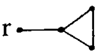
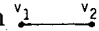
Since the graphs of the form are in $L(G)$, $\bar{u}_{\text{in}}$ contains either one node or two nodes.

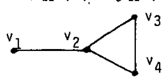
First let us assume that $\bar{u}_{\text{in}}$ contains only one node. Then since $L(G)$ contains more than one graph, there exist edges e, $\bar{e}$ in $E_{\bar{H}}$ such that $f_{\bar{H}}(e) \cap f_{\bar{H}}(\bar{e})=\bar{u}_{\text{in}}$, $(und\ \bar{G})_e$ is of the form  and $(und\ \bar{G})_{\bar{e}}$ belongs to $L(G)$. Now consider a derivation:

$$((und\ G)_e,\ e,\ u_{\text{in}},\ \text{Id}_{f_{\bar{H}}(e)}) \underset{\bar{G}}{\vdash} (M_1,\ \bar{e},\ u_{\text{in}},\ g_1) \underset{G}{\vdash} (M_2,\ e,\ u_{\text{in}},\ g_2)$$

with M_1, M_2, g_1, g_2 as described in definition 2.7 then it is easily seen that M_2 is of the form $(*)$ with $n \geqq m+2$; a contradiction.

On the other hand, assume that $\bar{u}_{\text{in}}$ contains two nodes. Then consider a derivation in $\bar{G}$ of a graph M of the form , which clearly belongs to $L(G)$.

Since all graphs in $L(G)$ that are not of the form or  have at least five nodes, and since for every pair of edges $e, \bar{e}$ in $E_{\bar{H}}, f_{\bar{H}}(e) \cap f_{\bar{H}}(\bar{e})=\bar{u}_{\text{in}}$ implies that $(und\ \bar{G})_e$ and $(und\ \bar{G})_{\bar{e}}$ belong to $L(\bar{G})$ and hence to $L(G)$, it follows that there exist such a pair $e, \bar{e}$ with $f_{\bar{H}}(e) \cap f_{\bar{H}}(\bar{e})=\bar{u}_{\text{in}}$, $(und\ \bar{G})_e$ is of the form 

and $(und\ \bar{G})_{\bar{e}}$ is of the form 

$(\bar{u}_{\text{in}}=\{v_1, v_2\})$.

Now it is easily seen that the graph M_2 obtained by:

$$((und\ G)_{\bar{e}},\ \bar{e},\ u_{\text{in}},\ \text{Id}_{f_{\bar{H}}(\bar{e})}) \underset{\bar{G}}{\vdash} (M_1,\ e,\ u_{\text{in}},\ g_1) \underset{\bar{G}}{\vdash} (M_2,\ \bar{e},\ u_{\text{in}},\ g_2),$$

with M_1, M_2, g_1, g_2 as specified in definition 2.7, is not in $L(G)$; a contradiction. $\square$

THEOREM 2.1: *$\mathscr{L}$(H) and $\mathscr{L}$(IH) are incomparable but not disjoint.*

Proof: That $\mathscr{L}$(H) and $\mathscr{L}$(IH) are incomparable follows from lemma 2.1 and 2.2. To see that they are not disjoint consider the H system $G=(H, \Gamma, e_{\text{in}})$ where:

$$H=(\{v_1, v_2\}, \{e_1, e_2\}, f_H)$$

with:

$$f_H(e_1)=\{v_1\}, \qquad f_H(e_2)=\{v_1, v_2\}$$

and where $\Gamma=\emptyset$ and $e_{\text{in}}=e_1$. Let G be the IH system $(\overline{H}_1, \overline{\Gamma}, \overline{u}_{\text{in}})$ where $\overline{H}=H$, $\overline{\Gamma}=\Gamma$ and $u_{\text{in}}=\{v_1\}$. Then it is easily seen that $L(G)=L(\overline{G})$. [More precisely, $L(G)$ is the set of all discrete graphs]. □

Although hypergraph systems and intersection-based hypergraph systems were presented as grammatical (thus generative) devices there is a quite close analogy between those systems and finite automata defining string languages. Given the transition graph of a finite automaton one may view this quite naturally as a hypergraph system where all the edges are of cardinality two. Following edges in the transition graph of a finite automaton corresponds to following edges in the multigraph system and so it corresponds to an H system. On the other hand, following states (nodes) in the transition graph corresponds to following intersections in an IH system.

However, this analogy is not complete because in (the transition graph of) a finite automaton there are two additional components controlling the way it defines a language. Firstly, transitions are directed, and so if after a transition A a transition B follows it does not necessarily mean that A can follow B; in other words, transitions do not have to be "symmetric". Secondly, certain "places" (nodes) are distinguished as terminal places and a derivation following the transition graph is considered successful only if its last step corresponds to a terminal place in the graph. We will now consider these two additional "control features" within the framework of H systems and IH systems. In this way one can view H systems and IH systems as examples of the *exhaustive* approach to graph language definition: one takes into the language of a given system everything the system generates (each "intermediate" graph also belongs to the language). On the other hand the systems we will consider next may be viewed as an example of a "selective" approach to graph language definition: from the set of all graphs that a system generates one takes into the language of the system only those graphs that satisfy a certain "filtering condition".

DEFINITION 2.9: *A directed hypergraph system with final edges*, abbreviated GFH system ([1]), is a system $G(H, \Gamma, e_{\text{in}}, E_{\text{fin}}, C)$ where *base* $G=(H, \Gamma, e_{\text{in}})$ is an H system, E_{fin} is a subset of E_H and C is a subset of $E_H \times E_H$ such that $(e, \overline{e}) \in C$ implies that $f_H(e) \cap f_H(\overline{e}) \neq \emptyset$. □

The graph (V_H, Γ) is denoted by *und* G.

DEFINITION 2.10: Let $G=(H, \Gamma, e_{in}, E_{fin}, C)$ be a GFH system.

(*a*) (A, e, g) is a *trace for* G if it is a trace for *base* G.

(*b*) Let (A, e, g) and $(B, \tilde{e}, \tilde{g})$ be traces for G. Then (A, e, g) *directly derives* $(B, \tilde{e}, \tilde{g})$ *in* G, denoted $(A, e, g) \underset{G}{\vdash} (B, \tilde{e}, \tilde{g})$, if $(e, \tilde{e}) \in C$ and $(A, e, g) \underset{\text{base } G}{\vdash} (B, \tilde{e}, \tilde{g})$

The relation $\overset{*}{\underset{G}{\vdash}}$ is defined to be the transitive and the reflexive closure of $\underset{G}{\vdash}$.

(*c*) the *language* of G, denoted by $L(G)$, is defined by:

$$L(G)=\{ M \mid ((und\, G)_{e_{in}}, e_{in}, \mathrm{Id}_{f_H(e_{in})}) \overset{*}{\underset{G}{\vdash}} (M, e, g) \text{ where } (M, e, g) \text{ is a trace for } G \text{ and } e \in E_{fin} \}. \quad \square$$

DEFINITION 2.11: A *directed intersection-based hypergraph system with final intersections* (abbreviated GFIH *system*) is a system $G=(H, \Gamma, u_{in}, I_{fin}, C)$ where *base* $G=(H, \Gamma, u_{in})$ is an IH system, I_{fin} is a subset of *int* H and C is a subset of *int* $H \times$ *int* H such that $(u, \bar{u}) \in C$ implies that there exist e_1, e_2, e_3 in E_H with $f_H(e_1) \cap f_H(e_2)=u$ and $f_H(e_2) \cap f_H(e_3)=\bar{u}$. $\square$

The graph (V_H, Γ) is denoted as *und* G.

DEFINITION 2.12: Let $G=(H, \Gamma, u_{in}, I_{fin}, C)$ be a GFIH system.

(*a*) (A, e, u, g) is an *extended trace for* G if it is an extended trace for *base* G.

(*b*) Let (A, e, u, g) and $(B, \tilde{e}, \tilde{u}, \tilde{g})$ be extended traces for G. Then (A, e, u, g) *directly derives* $(B, \tilde{e}, \tilde{u}, \tilde{g})$ *in* G, denoted $(A, e, u, g) \underset{G}{\vdash} (B, \tilde{e}, \tilde{u}, \tilde{g})$, if $(u, \tilde{u}) \in C$ and $(A, e, u, g) \underset{\text{base } G}{\vdash} (B, \tilde{e}, \tilde{u}, \tilde{g})$. The relation $\overset{*}{\underset{G}{\vdash}}$ is the transitive and the reflexive closure of $\underset{G}{\vdash}$.

(*c*) The *language* of G, denoted by $L(G)$, is defined by:

$$L(G)= \{ M \mid ((und\, G)_{e_{in}}, e, u_{in}, \mathrm{Id}_{f_H(e_{in})}) \overset{*}{\underset{G}{\vdash}} (M, \tilde{e}, \tilde{u}, g) \text{ where } ((und\, G)_e, e, u_{in}, \mathrm{Id}_{f_H(e)}) \text{ and } (M, \tilde{e}, \tilde{u}, \tilde{g}) \text{ are extended traces for } G \text{ and } \tilde{u} \in I_{fin} \}. \quad \square$$

If in a GFH system $G=(H, \Gamma, e_{in}, E_{fin}, C)$, $E_{fin}=E_H$, then we omit E_{fin} from the specification of G. In this case G will be called a GH *system*. On the other hand if $G=(H, \Gamma, e_{in}, E_{fin}, C)$ and $C=\{(e, \tilde{e}) \mid e, \tilde{e} \in E_H \text{ and } f_H(e) \cap f_H(\tilde{e}) \neq \emptyset \}$ then we omit C from the specification of G and we say that G is a FH *system*. Analogously, we define GIH *systems* and FIH *systems*. A GIH *system* is a GFIH system $G=(H, \Gamma, u_{in}, I_{fin}, C)$ where $I_{fin}=int(H)$; in this case, I_{fin} will be omitted from the specification of G. If on the other hand

[1] G abbreviates "gericht" which is "directed" in Dutch.

$C=\{(u,\tilde{u})\mid u,\tilde{u}\in int\,H$ and there exist e_1, e_2, e_3 in E_H with $f_H(e_1)\cap f_H(e_2)=u$ and $f_H(e_2)\cap f_H(e_3)=\tilde{u}\}$ then we call G a FIH *system* and C is omitted from the specification of G.

Example 2.3: Let $G=(H, \Gamma, u_{in}, I_{fin}, C)$ where:

$$H=(\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9\}, \{e_1, e_2, e_3\}, f)$$

with:

$$f(e_1)=\{v_1, v_2, v_3, v_4, v_5, v_6\}, \qquad f(e_2)=\{v_2, v_7, v_4, v_8, v_6, v_9\},$$

$$f(e_3)=\{v_1, v_7, v_3, v_8, v_5, v_9\},$$

$$\Gamma=\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}, \{v_5, v_6\},$$
$$\{v_6, v_1\}, \{v_2, v_7\}, \{v_7, v_4\}, \{v_4, v_8\}, \{v_8, v_6\},$$
$$\{v_6, v_9\}, \{v_9, v_2\}, \{v_1, v_7\}, \{v_7, v_3\}, \{v_3, v_8\},$$
$$\{v_8, v_5\}, \{v_5, v_9\}, \{v_9, v_1\}\},$$

where:

$$C=\{(u_1, u_2), (u_2, u_3), (u_3, u_1)\}$$

and:

$$u_1=f(e_1)\cap f(e_2), \qquad u_2=f(e_2)\cap f(e_3)$$

$$u_3=f(e_3)\cap f(e_1),$$

$$u_{in}=u_1 \text{ and } I_{fin}=\{u_2\}.$$

The system G is depicted in figure 2.5:

$$u_{in}=\{v_2, v_4, v_6\}, \qquad I_{fin}=\{\{v_7, v_8, v_9\}\}$$

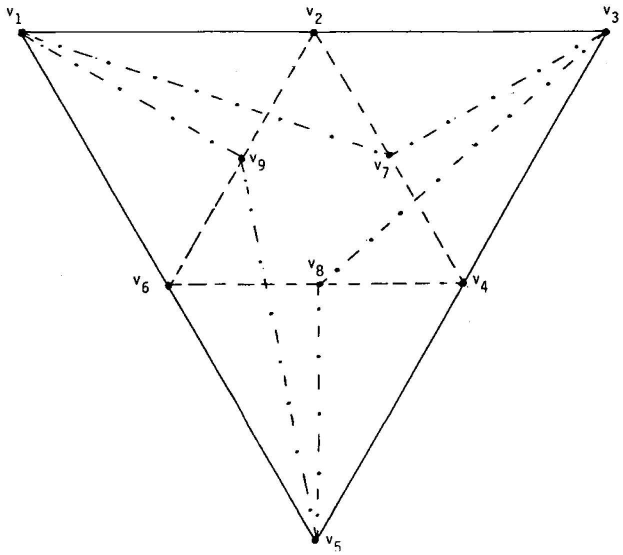


Figure 2.5.

Figure 2.6 depicts the "derivation sequence":

$$((und\,G)_{e_1}, e_1, u_{in}, \mathrm{Id}_{f(e_1)} \underset{G}{\vdash} (M^{(1)}, e_2, u_2, g^{(1)}) \underset{G}{\vdash} (M^{(2)}, e_3, u_3, g^{(2)}).$$

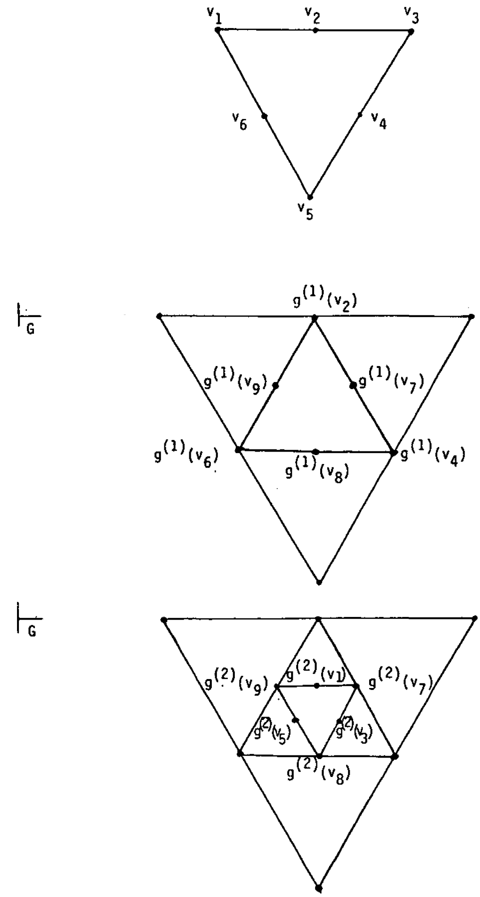


Figure 2.6.

$L(G)$ is the set of graphs of the form depicted in figure 2.7.

We conclude this section with the following observation. Although the graph-language generating systems discussed in this section bear a certain similarity to finite automata defining string languages there are certain important differences between our systems and finite string-automata. Since our systems define graphs rather than strings, they are considerably more difficult to analyze. In particular,

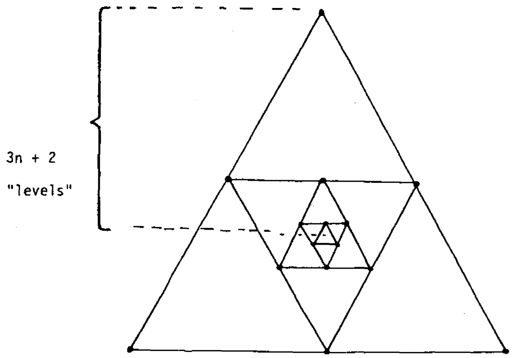


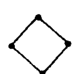
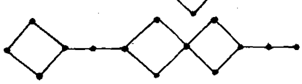
Figure 2.7. — General form of the graphs in $L(G)$. □

certain questions concerning the effectiveness of defining graph-languages by our systems turn out to be undecidable, while the corresponding questions for finite string-automata are "easily" decidable. Here is an example of such a situation.

THEOREM 2.2: *For two arbitrary given* GFH *systems* G *and* $\overline{G}$, *it is undecidable whether or not* $L(G) \cap L(\overline{G})$ *is empty.*

Proof: We show that a decision procedure for this question yields a decision procedure for the Post Correspondence Problem.

Let $A = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$ and $B = \langle \beta_1, \beta_2, \ldots, \beta_n \rangle$ be two lists of words from $\{0, 1\}^+$. Since a formal description of the construction is complicated, we give only an intuitive idea of it.

Firstly, for every word α_i in A (and β_i in B) we construct the graph $\alpha^{(i)}$ ($\beta^{(i)}$ respectively) by coding every occurrence of 1 in the word by a graph
and every 0 by a graph  . e. g., to the word 01001 corresponds the graph 

Assume that the $V_{\alpha^{(i)}}$ and $V_{\beta^{(i)}}$ are pairwise disjoint.

Now for each $\alpha^{(i)}$ let $l(\alpha^{(i)})$ and $r(\alpha^{(i)})$ denote the "leftmost" and the "rightmost" node of $\alpha^{(i)}$ respectively. Let $l(\beta^{(i)})$ and $r(\beta^{(i)})$ be defined analogously. By $\xi^{(i)}$ we denote the graph $(V_{\alpha^{(i)}} \cup V_{\beta^{(i)}}, E_{\alpha^{(i)}} \cup E_{\beta^{(i)}})$ (note that $V_{\alpha^{(i)}}$ and $V_{\beta^{(i)}}$ are disjoint). Now let v_1, v_2, v_3, v_4 be distinct nodes and construct the graph M by identifying, for $1 \leqq i \leqq n$, the nodes $l(\alpha^{(i)})$ with v_1, the nodes $l(\beta^{(i)})$

with v_2, $r(\alpha^{(i)})$ with v_3 and $r(\beta^{(i)})$ with v_4. Thus M is of the form depicted in figure 2.8.

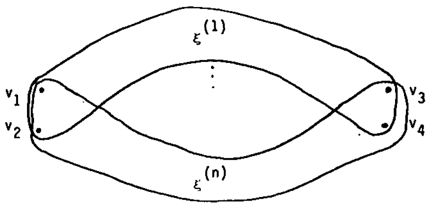
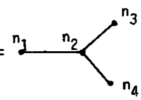


Figure 2.8. – The graph M.

Let I be the discrete graph $(\{v_1, v_2, v_3, v_4\}, \emptyset)$, let g be the injective function from $\{v_1, v_2, v_3, v_4\}$ defined by:

$$g(v_1)=v_3, \qquad g(v_2)=v_4, \; g(v_3)=v_1, \qquad g(v_4)=v_2$$

and let $\overline{M}$ be the gluing of M and M along I by Id_{V_I} and g. Hence for each integer i with $0 \leqq i \leqq n$, $\overline{M}$ contains two copies of $\xi^{(i)}$: a copy $\xi_1^{(i)}$ in which the nodes corresponding to $l(\alpha^{(i)})$ and $l(\beta^{(i)})$ are identified with v_1 and v_2, and in which the nodes corresponding to $r(\alpha^{(i)})$ and $r(\beta^{(i)})$ are identified with v_3 and v_4, and another copy, $\xi_2^{(i)}$ in the "reverse direction", that is, the nodes corresponding to $l(\alpha^{(i)})$, $l(\beta^{(i)})$, $r(\alpha^{(i)})$ and $r(\beta^{(i)})$ are identified with v_3, v_4, v_1 and v_2 respectively.

Finally, construct the graph $\overline{\overline{M}}$ by gluing the graph $K=$ 

and $\overline{M}$ along I by α and β where I is the discrete graph $(\{n_3, n_4\}, \emptyset)$, $\alpha=\mathrm{Id}_{\{n_3, n_4\}}$ and β is defined by $\beta(n_3)=v_1$, $\beta(n_4)=v_2$.

We are now ready to construct the GFH system $G=(H, \Gamma, e_{\mathrm{in}}, E_{\mathrm{fin}}\, C)$ where H is the hypergraph $(V_M, \{e_0, e_1^{(1)}, e_1^{(2)}, \ldots, e_1^{(n)}, e_2^{(1)}, e_2^{(2)}, \ldots, e_2^{(n)}\}, f)$ with:

$$f(e_0)=v_k, \qquad f(e_1^{(1)})=V_{\xi_1^{(1)}}, f(e_1^{(2)}) = V_{\xi_1^{(2)}}, \ldots, \qquad f(e_1^{(n)})=V_{\xi_1^{(n)}},$$

$$f(e_2^{(1)}) = V_{\xi_2^{(1)}}, f(e_2^{(2)})=V_{\xi_2^{(2)}}, \ldots, f(e_2^{(n)})=V_{\xi_2^{(n)}},$$

$$\Gamma=E_M, \qquad e_{\mathrm{in}}=e_0, \qquad E_{\mathrm{fin}}=E_H \setminus \{e_0\}$$

and:

$$C=(\{e_0\} \times E_1) \cup (E_1 \times E_2) \cup (E_2 \times E_1)$$

where $E_1=\{e_1^{(1)}, e_1^{(2)}, \ldots, e_1^{(n)}\}$ and $E_2=\{e_2^{(1)}, e_2^{(2)}, \ldots, e_2^{(n)}\}$.

The language $L(G)$ consists of the graphs of the form of figure 2.9.

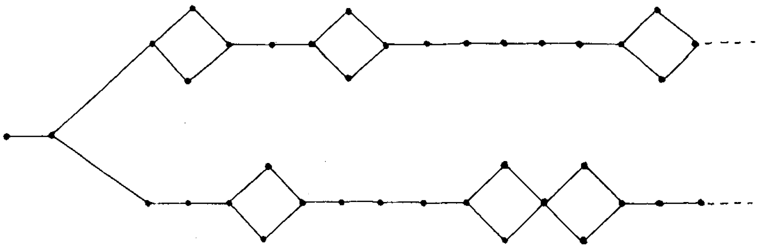


Figure 2.9.

With the property that there exists a sequence of indices $i_1, i_2, \ldots, i_k$ with $0 \leqq i_j \leqq n$ and such that the "upper half" of the graph corresponds to $\alpha_{i_1} \alpha_{i_2} \ldots \alpha_{i_k}$ and the "lower half" corresponds to $\beta_{i_1} \beta_{i_2} \ldots \beta_{i_k}$.

On the other hand let W be the graph depicted in figure 2.10.

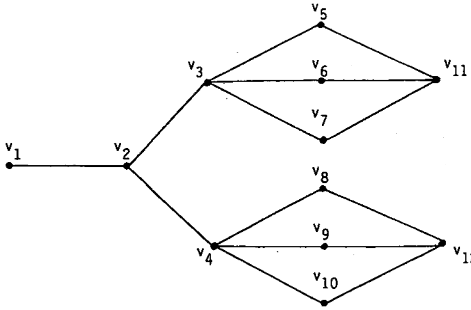


Figure 2.10. — The graph *W*.

Let $\bar{G}$ be the GFH system $\bar{G} = (\bar{H}, \bar{\Gamma}, \bar{e}_{in}, \bar{E}_{fin}, \bar{C})$ where:

$$\bar{H} = (V_W, \{e_1, e_2, e_3, e_4, e_5\}, f)$$

with f defined by:

$$f(e_1) = \{v_1, v_2, v_3, v_4\}, \qquad f(e_2) = \{v_3, v_4\},$$

$$f(e_3) = \{v_3, v_5, v_7, v_{11}, v_4, v_8, v_{10}, v_{12}\},$$

$$f(e_4) = \{v_3, v_6, v_{11}, v_4, v_9, v_{12}\}, \qquad f(e_5) = \{v_{11}, v_{12}\},$$

$$\bar{\Gamma} = E_W,$$

$$\bar{e}_{in} = e_1,$$

$$\bar{E}_{fin} = E_{\bar{H}} \setminus \{e_1\}$$

and:

$$\overline{C}=\{(e_1, e_3), (e_1, e_4), (e_3, e_5), (e_4, e_5), (e_5, e_3), (e_5, e_4), (e_3, e_2), (e_4, e_2), (e_2, e_3), (e_2, e_4)\}.$$

$L(\overline{G})$ is the set of graphs of the form depicted in figure 2.9 such that the word, corresponding to the "upper half" of the graph equals the word, corresponding to the "lower half" of it.

We conclude that $L(G) \cap L(\overline{G}) \neq \emptyset$ if and only if the instance (A, B) of the Post Correspondence problem has a solution. □

Observe that the proof technique can be modified to yield an analogous theorem for GFIH systems.

III. OVERLAPPING GRAPH SYSTEMS

In all the systems considered in the last section one notices the following phenomenon: even though two intersecting edges of a hypergraph (including their graph structure) may differ considerably, they always are identical within their intersection area. Since in a derivation step only intersecting edges may be used, this particular feature implies the following restriction: if X is an intermediate graph obtained in a derivation of a graph Y and X contains two nodes x_1, x_2 with no edge connecting them, then also in Y (nodes corresponding to) x_1 and x_2 will have no edge connecting them. For this reason it seems natural to consider systems in which the basic building blocks will be graphs. Some of these graphs may have common nodes, however the structure of edges on the nodes common to two different graphs may be quite different. Such systems are considered in this section.

DEFINITION 3.1: An *overlapping graph system* (abbreviated O system) is a pair $G=(\mathscr{H}, H_{in})$ where $\mathscr{H}$ is a finite nonempty set of graphs and $H_{in} \in \mathscr{H}$; H_{in} is called the *initial graph* of G. □

DEFINITION 3.2: Let $G=(\mathscr{H}, H_{in})$ be an O system. (A, H, g) is a *trace for* G if and only if it is a trace over $\mathscr{H}$. □

DEFINITION 3.3: Let $G=(\mathscr{H}, H_{in})$ be an O system and let (A, H, g), $(B, \tilde{H}, \tilde{g})$ be traces for G. We say that (A, H, g) *directly derives* $(B, \tilde{H}, \tilde{g})$ *in* G, denoted $(A, H, g) \underset{G}{\vdash} (B, \tilde{H}, \tilde{g})$, if:

(1) $u=V_H \cap V_{\tilde{H}} \neq \emptyset$,

(2.1) B is the gluing of A and $\tilde{H}$ along $\tilde{H}_u$ by α and β where α is the restriction of g to u and β equals Id_u, and:

(2.2) $\tilde{g}$ is the natural injection of $\tilde{H}$ into B.

By $\overset{*}{\underset{G}{\vdash}}$ we denote the transitive and the reflexive closure of the relation $\underset{G}{\vdash}$. If $(A, H, g) \overset{*}{\underset{G}{\vdash}} (B, \tilde{H}, \tilde{g})$ then we say that (A, H, g) *derives* $(B, \tilde{H}, \tilde{g})$ *in* G. □

DEFINITION 3.4: The *language* of the O system $G=(\mathscr{H}, H_{\text{in}})$, denoted $L(G)$, is defined by:

$$L(G)=\{ M \mid (H_{\text{in}}, H_{\text{in}}, \text{Id}_{V_{H_{\text{in}}}}) \overset{*}{\underset{G}{\vdash}} (M, H, g)$$

$$\text{where } (M, H, g) \text{ is a trace for } G\}. \quad \square$$

Remark 3.1: If $G=(\mathscr{H}, H_{\text{in}})$ is an O system with the property that for each H, $\overline{H}$ in $\mathscr{H}$ and for each $x, y \in V_H \cap V_{\overline{H}}$ either $\{x, y\} \in E_H \cap E_{\overline{H}}$ or $\{x, y\} \notin E_H$ and $\{x, y\} \notin E_{\overline{H}}$, then G is equivalent in a natural way to the H system G constructed as follows.

Let K be the hypergraph with $V_K=\bigcup_{H \in \mathscr{H}} V_H$, $E_K=\mathscr{H}$ and for $H \in \mathscr{H}$, $f_K(H)=V_H$. Let $\Gamma=\bigcup_{H \in \mathscr{H}} E_H$. Then $\bar{G}=(K, \Gamma, H_{\text{in}})$.

On the other hand, it is easily seen that every H system $G=(H, \Gamma, e_{\text{in}})$ gives rise to an equivalent O system $(\mathscr{H}, H_{\text{in}})$ where $\mathscr{H}=\{(und\, G)_e \mid e \in E_H\}$ and $H_{\text{in}}=(und\, G)_{e_{\text{in}}}$. □

As in the case of H systems we will now define a counterpart of O systems based on intersections rather than on edges.

DEFINITION 3.5: Let $\mathscr{H}$ be a set of graphs. Then the *set of intersections of* $\mathscr{H}$, denoted by *int* $\mathscr{H}$, is the set $\{X \mid$ there exist distinct H, $\overline{H}$ in $\mathscr{H}$ with $X=V_H \cap V_{\overline{H}} \neq \varnothing\}$. □

DEFINITION 3.6: An *intersection based overlapping graph system* (abbreviated IO *system*) is a system $(\mathscr{H}, u_{\text{in}})$ where $\mathscr{H}$ is a finite nonempty set of graphs and u_{in} is an element of *int* $\mathscr{H}$. □

DEFINITION 3.7: Let $G=(\mathscr{H}, u_{\text{in}})$ be an IO system. A system (A, H, u, g) is an *extended trace for* G if (A, H, g) is a trace over $\mathscr{H}$ and u is an element of *int* $\mathscr{H}$ such that there exists an $\overline{H}$ in $\mathscr{H}$ with $u=V_H \cap V_{\overline{H}}$. □

DEFINITION 3.8: Let $G=(\mathscr{H}, u_{in})$ be an IO system and let (A, H, u, g) and $(B, \tilde{H}, \tilde{u}, \tilde{g})$ be extended traces for G. We say that (A, H, u, g) *directly derives* $(B, \tilde{H}, \tilde{u}, \tilde{g})$ *in* G, denoted $(A, H, u, g) \underset{G}{\vdash} (B, \tilde{H}, \tilde{u}, \tilde{g})$, if:

$$(1)\ V_H \cap V_{\tilde{H}} = u,$$

(2.1) B is the gluing of A and $\tilde{H}$ along u by α and Id_u where α is the restriction of g to u, and:

(2.2) $\tilde{g}$ is the natural injection of $\tilde{H}$ in B.

The relation $\overset{*}{\underset{G}{\vdash}}$ is defined to be the transitive and the reflexive closure of $\underset{G}{\vdash}$. If $(A, H, u, g) \overset{*}{\underset{G}{\vdash}} (B, \tilde{H}, \tilde{u}, \tilde{g})$ then we say that (A, U, u, g) *derives* $(B, \tilde{H}, \tilde{u}, \tilde{g})$ *in* G. □

DEFINITION 3.9: Let $G=(\mathscr{H}, u_{in})$ be an IO system. The *language of* G, denoted $L(G)$, is defined by:

$$L(G)=\{ M \mid (H, H, u_{in}, \mathrm{Id}_{u_{in}}) \overset{*}{\underset{G}{\vdash}} (M, \tilde{H}, \tilde{u}, \tilde{g}) \text{ where } (H, H, u_{in}, \mathrm{Id}_{u_{in}}) \text{ and } (M, \tilde{H}, \tilde{u}, \tilde{g}) \text{ are extended traces for } G \}. \quad \square$$

Analogously to definitions 2.9, 2.10, 2.11 and 2.12 one can introduce extra control features into the framework of O systems and IO systems; these control features correspond to the directed transitions and the final places of finite automata. The so obtained systems will be called *directed overlapping graph systems with final graphs* (abbreviated GFO *systems*) and *directed intersection-based overlapping graph systems with final intersections* (abbreviated GFIO *systems*) respectively.

This gives rise to GO systems, FO systems, GIO systems and FIO systems, analogously to GH, GH, GIH and FIH systems.

Example 3.1: Let $G=(\mathscr{H}, H_{in}, \mathscr{H}_{fin}, C)$, where $\mathscr{H}=\{H_1, H_2, H_3, H_4\}$ with:

$$H_1=(\{v_1, v_2, v_3, v_4\}, \{\{v_1, v_2\}, \{v_3, v_4\}\}),$$
$$H_2=(\{v_1, v_3, v_5\}, \{\{v_1, v_3\}, \{v_3, v_5\}, \{v_5, v_1\}\}),$$
$$H_3=(\{v_2, v_4, v_6, v_7\}, \{\{v_2, v_4\}, \{v_4, v_6\}, \{v_6, v_7\}, \{v_7, v_2\}\}),$$
$$H_4=(\{v_3, v_5, v_6, v_7\}, \{\{v_3, v_5\}, \{v_5, v_6\}, \{v_6, v_7\}, \{v_7, v_3\}\}),$$
$$H_{in}=H_1,$$
$$\mathscr{H}_{fin}=\{H_4\}$$

and:

$$C=\{(H_1, H_2), (H_1, H_3), (H_2, H_4), (H_3, H_4)\}.$$

In the above we assume that $v_i \neq v_j$ whenever $i \neq j$. In this way the intersection structure is automatically given. *We will use this convention throughout this paper.*

The language $L(G)$ consists of the graphs of one of the forms depicted in figure 3.1.

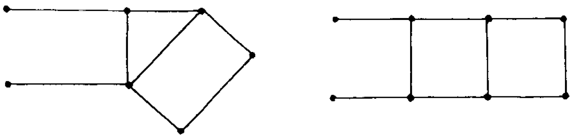


Figure 3.1

REMARK 3.2: As in remark 3.1 it is obvious that every XH system can be considered as a special form of a XO system where X stands for G, F, GF, GI, FI or GFI.

In the rest of this section we compare the graph-language generating power of the systems considered so far.

We start by investigating systems based on edges (rather than on intersections).

LEMMA 3.1 : $\mathscr{L}(\mathrm{GO}) \setminus \mathscr{L}(\mathrm{FO}) \neq \emptyset$.

Proof: Let G be the GO system $G=(\mathscr{H}, H_{in}, C)$ with $\mathscr{H}=\{H_1, H_2, H_3\}$ where:

$$H_1=(\{v_1, v_2\}, \{\{v_1, v_2\}\}),$$

$$H_2=(\{v_2, v_3\}, \{\{v_2, v_3\}\})$$

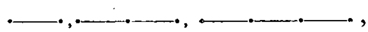
and:

$$H_3=(\{v_3, v_1\}, \{\{v_3, v_1\}\}),$$

$$H_{in}=H_1$$

and:

$$C=\{(H_1, H_2), (H_2, H_3), (H_3, H_1)\}.$$

Clearly, $L(G)=\{$  $, \ldots\}$.

Now assume $\overline{G}=(\mathscr{H}, \mathscr{H}_{in}, \mathscr{H}_{fin})$ is a FO system with $L(\overline{G})=L(G)$. Since $L(G)$ contains graphs of an arbitraryly large size, there exist H, $\overline{H}$ in $\mathscr{H}$ with

$V_{\bar{H}} \setminus V_H \neq \emptyset$, $V_H \cap V_{\bar{H}} \neq \emptyset$, and there exist traces $(H_{in}, H_{in}, \mathrm{Id}_{V_{H_{in}}})$, (M_1, H, g) and $(M_2, \tilde{H}, \tilde{g})$ for $\bar{G}$ such that $(H_{in}, H_{in}, \mathrm{Id}_{V_{H_{in}}}) \overset{*}{\underset{\bar{G}}{\vdash}} (M_1, H, g) \overset{*}{\underset{\bar{G}}{\vdash}} (M_2, \tilde{H}, \tilde{g})$ and $\tilde{H} \in \mathscr{H}_{fin}$.

This implies however that $L(\bar{G})$ contains either disconnected graphs or graphs of an arbitrarily large degree: indeed, after deriving M_1 one can choose to glue $\bar{H}$; followed by H. Repeating these two steps an arbitrary number n of times, either the degree of the resulting graph M is increased each time $\bar{H}$ is used, or the number of connected components of M is increased. Since we have $(M_1, H, g) \overset{*}{\underset{\bar{G}}{\vdash}} (M_2, \tilde{H}, \tilde{g})$ and $\tilde{H} \in \mathscr{H}_{fin}$ we know that there exists traces (M, H, h) and $(\tilde{M}, \tilde{H}, \tilde{h})$ with $(M, H, h) \overset{*}{\underset{\bar{G}}{\vdash}} (\tilde{M}, \tilde{H}, \tilde{h})$. Since $\tilde{H} \in \mathscr{H}_{fin}$ we see that $\tilde{M} \in L(G)$. However, if $n \geqq 1$ then it is easily seen that $\tilde{M} \notin L(G)$, a contradiction. □

LEMMA 3.2: $\mathscr{L}(\mathrm{GH}) \setminus \mathscr{L}(\mathrm{FH}) \neq \emptyset$.

Proof: This is an easy consequence of remark 3.2: the GO system G from the proof of lemma 3.1 can be considered as a GH system, and thus $\mathscr{L}(G) \in \mathscr{L}(\mathrm{GH}) \setminus \mathscr{L}(\mathrm{FO})$. Since $\mathscr{L}(\mathrm{FH}) \subseteq \mathscr{L}(\mathrm{FO})$ we have $L(G) \in \mathscr{L}(\mathrm{GH}) \setminus \mathscr{L}(\mathrm{FH})$. □

LEMMA 3.3: $\mathscr{L}(\mathrm{FO}) \setminus \mathscr{L}(\mathrm{GO}) \neq \emptyset$.

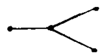
Proof: Consider the FO system $G = (\mathscr{H}, H_{in}, \mathscr{H}_{fin})$ with $\mathscr{H} = \{H_1, H_2, H_3\}$ where:

$$H_1 = (\{v_1\}, \emptyset\}),$$

$$H_2 = (\{v_1, v_2, v_3, v_4, v_5\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}\}),$$

$$H_3 = (\{v_1, v_6, v_7, v_8\}, \{\{v_1, v_6\}, \{v_1, v_7\}, \{v_1, v_8\}\}),$$

$$H_{in} = H_1 \quad \text{and} \quad \mathscr{H}_{fin} = \{H_2, H_3\}.$$

Assume that $\bar{G} = (\bar{\mathscr{H}}, \bar{H}_{in}, C)$ is a GO system with $L(\bar{G}) = L(G)$. Since the graph with the smallest number of nodes in $L(G)$ is of the form:  $\bar{H}_{in}$ must also be of this form.

However, $L(\bar{G})$ contains a graph of the form , which clearly cannot be obtained from $\bar{H}_{in}$ in $\bar{G}$. □

LEMMA 3.4: $\mathscr{L}(\mathrm{FH}) \setminus \mathscr{L}(\mathrm{GH}) \neq \emptyset$.

Proof: This is again an easy consequence of remark 3.2: The FO system from the proof of lemma 3.3 can be considered as a FH system.

Thus $L(G) \in \mathscr{L}(\mathrm{FH}) \setminus \mathscr{L}(\mathrm{GO})$. Since $\mathscr{L}(\mathrm{GH}) \subseteq \mathscr{L}(\mathrm{GO})$ we have $L(G) \in \mathscr{L}(\mathrm{FH}) \setminus \mathscr{L}(\mathrm{GH})$. □

THEOREM 3.1: *$\mathscr{L}(\mathrm{GO})$ and $\mathscr{L}(\mathrm{FO})$ are incomparable but not disjoint.*

Proof: This result follows easily from lemma 3.1, lemma 3.3 and from the fact that every O system $(\mathscr{H}, H_{\mathrm{in}})$ can be considered as being a GO system $(\mathscr{H}, H_{\mathrm{in}}, C)$ with $C = \{(H\ \tilde{H}) \mid H, \tilde{H} \in \mathscr{H}$ and $V_H \cap V_{\tilde{H}} \neq \emptyset\}$ as well as a FO system $(\mathscr{H}, H_{\mathrm{in}}, \mathscr{H}_{\mathrm{fin}})$ with $\mathscr{H}_{\mathrm{fin}} = \mathscr{H}$. □

THEOREM 3.2. $\mathscr{L}(\mathrm{GH})$ and $\mathscr{L}(\mathrm{FH})$ are incomparable but not disjoint.

Proof: This result follows easily from lemma 3.2, lemma 3.4 and from the fact that every H system can be considered as being a GH system as well as a FH system. □

THEOREM 3.3: $\mathscr{L}(\mathrm{H}) \subsetneqq \mathscr{L}(\mathrm{O})$.

Proof: The inclusion $\mathscr{L}(\mathrm{H}) \subseteq \mathscr{L}(\mathrm{O})$ follows from remark 3.1. To prove the strict inclusion, consider the O system $G = (\mathscr{H}, H_{\mathrm{in}})$ where $\mathscr{H} = \{H_1, H_2\}$ with:

$$H_1 = (\{v_1, v_2\}, \emptyset),$$

$$H_2 = (\{v_1, v_2\}, \{\{v_1, v_2\}\})$$

and:

$$H_{\mathrm{in}} = H_1.$$

Then clearly $L(G)$ contains only the graphs of the forms • • and •——•.

Now assume that $\bar{G} = (H, \Gamma, e_{\mathrm{in}})$ is a H system with $L(\bar{G}) = L(G)$. Then clearly $(und\,\bar{G})_{e_{\mathrm{in}}}$ must be of the form • •. Since $(und\,\bar{G})_{e_{\mathrm{in}}}$ is a full subgraph of $und\,\bar{G}$, the graph •——• does not belong to $L(\bar{G})$; a contradiction. □

THEOREM 3.4: $\mathscr{L}(\mathrm{GH}) \subsetneqq (\mathrm{GO})$.

Proof: The O system G of the proof of theorem 3.3 can be considered as a GO system and the same reasoning shows that $L(G) \notin \mathscr{L}(\mathrm{GH})$. □

THEOREM 3.4: *The diagram of figure 3.2 holds:*

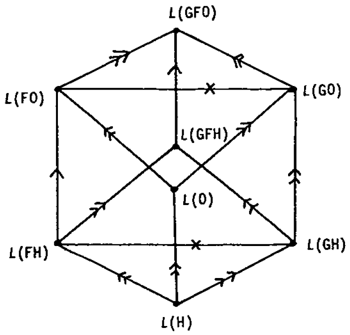


figure 3.2

where we denote $A \rightarrowtail B$ if $A \subseteq B$, $A \twoheadrightarrow B$ if $A \subsetneqq B$ and $A \multimap B$ if A and B are incomparable but not disjoint. □

Next we consider systems based on intersections.

Lemma 3.5: $\mathscr{L}(\mathrm{GIO})\setminus\mathscr{L}(\mathrm{FIO})\neq\emptyset$.

Proof: Let G be the GIO system $G=(\mathscr{H}, u_{in}, C)$ with $\mathscr{H}=\{H_1, H_2, H_3\}$, where:

$$H_1=(\{v_1, v_2\}, \{\{v_1, v_2\}\}),$$
$$H_2=(\{v_2, v_3\}, \{\{v_2, v_3\}\})$$
$$H_3=(\{v_3, v_1\}, \{\{v_3, v_1\}\}),$$
$$u_{in}=\{v_1\}$$

and:

$$C=\{(\{v_1\}, \{v_2\}), (\{v_2\}, \{v_3\}), (\{v_3\}, \{v_1\})\}.$$

Clearly, $L(G)$ is of the form:

$L(G)=\{$  $, \ldots\}$

Now assume that $\overline{G}=(\overline{\mathscr{H}}, \overline{u}_{in}, I_{fin})$ is a FIO system with $L(\overline{G})=L(G)$. Then there exist $H, \overline{H}\in\mathscr{H}$ with $V_H \cap V_{\overline{H}}=\overline{u}_{in}$, $V_{\overline{H}}\setminus V_H\neq\emptyset$ and $(\overline{H}, \overline{H}, \overline{u}_{in}, \mathrm{Id}_{V_{\overline{H}}}) \overset{*}{\underset{\overline{G}}{\vdash}} (\tilde{M}, \tilde{H}, \tilde{u}, \tilde{g})$ where $(\tilde{M}, \tilde{H}, \tilde{u}, \tilde{g})$ is an extended trace for $\overline{G}$ and where $\tilde{u}\in I_{fin}$.

Now consider the following sequence of derivation steps:

$$(\overline{H}, \overline{H}, \overline{u}_{in}, \mathrm{Id}_{V_{\overline{H}}}) \underset{\overline{G}}{\vdash} (M_2, \overline{H}, \overline{u}_{in}, g_2) \underset{\overline{G}}{\vdash} (M_3, H, \overline{u}_{in}, g_3) \underset{\overline{G}}{\vdash} (M_4, \overline{H}, \overline{u}_{in}, g_4)$$

with $M_1, M_2, M_3, M_4, g_1, g_2, g_3, g_4$ specified as in definition 3.8. From $(\overline{H}, \overline{H}, \overline{u}_{in}, \mathrm{Id}_{V_{\overline{H}}}) \overset{*}{\underset{\overline{G}}{\vdash}} (\tilde{M}, \tilde{H}, \tilde{u}, \tilde{g})$ and $\tilde{u}\in I_{fin}$ it follows that we have $(M_4, \overline{H}, \overline{u}_{in}, g_4) \underset{\overline{G}}{\vdash} (M, \tilde{H}, \tilde{u}, g)$ for some g and for some M in $L(\overline{G})$. However, since $V_{\overline{H}}\setminus V_H\neq\emptyset$, either there is an edge $\{a, b\}$ in $E_{\overline{H}}$ with $a\in V_H$ and $b\notin V_H$ or there is no such edge. In the first case, M_4 and M are of degree at least three, in the latter case M_4 and M are disconnected. In both cases, M is not in $L(G)$; a contradiction. □

Lemma 3.6: $\mathscr{L}(\mathrm{GIH})\setminus\mathscr{L}(\mathrm{FIH})\neq\emptyset$.

Proof: This is an easy consequence of remark 3.2: the GIO system G from the proof of lemma 3.5 can be considered as a GIH system. Since $\mathscr{L}(\mathrm{FIH}) \subseteq \mathscr{L}(\mathrm{FIO})$ the result follows. □

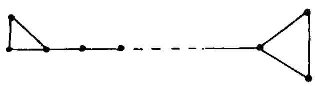
LEMMA 3.7: $\mathscr{L}(\mathrm{FIO}) \setminus \mathscr{L}(\mathrm{GIO}) \neq \emptyset$.

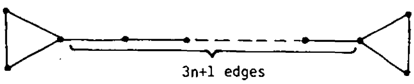
Proof: Consider the FIO system $g=(\mathscr{H}, u_{\mathrm{in}}, I_{\mathrm{fin}})$ with:

$$\mathscr{H}=\{H_1, H_2, H_3, H_4, H_5, H_6, H_7\},$$

where:

$$H_1=(\{v_1\}, \emptyset),$$
$$H_2=(\{v_1, v_2, v_3\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}),$$
$$H_3=(\{v_3, v_4\}, \{\{v_3, v_4\}\}),$$
$$H_4=(\{v_4, v_5\}, \{\{v_4, v_5\}\}),$$
$$H_5=(\{v_5, v_3\}, \{\{v_5, v_3\}\}),$$
$$H_6=(\{v_4, v_6, v_7\}, \{\{v_4, v_6\}, \{v_6, v_7\}, \{v_7, v_4\}\}),$$
$$H_7=(\{v_7\}, \emptyset),$$
$$u_{\mathrm{in}}=\{\{v_1\}\} \quad \text{and} \quad I_{\mathrm{fin}}=\{\{v_7\}\}.$$

Assume that $\overline{G}=(\overline{\mathscr{H}}, \overline{u}_{\mathrm{in}}, C)$ is a GIO system with $L(\overline{G})=L(G)$. Since for every pair $H, \overline{H}$ in $\overline{\mathscr{H}}$ with $V_H \cap V_{\overline{H}}=\overline{u}_{\mathrm{in}}$ we have $H, \overline{H} \in L(\overline{G})$ and since every graph of $L(G)$ contains a subgraph of the form  we know that every graph H in $\overline{\mathscr{H}}$ for which there exists a $\overline{H}$ in $\overline{\mathscr{H}}$ with $V_H \cap V_{\overline{H}}=\overline{u}_{\mathrm{in}}$ has a subgraph of this form. However, $L(G)$ contains the graphs of the form



where $n \geqq 0$. If M is such a graph with $3n+1 > \max_{H \in \overline{\mathscr{H}}} \# V_{\overline{H}}$ then it is easily seen that M cannot be derived in $\overline{G}$; a contradiction. □

LEMMA 3.8: $\mathscr{L}(\mathrm{FIH}) \setminus \mathscr{L}(\mathrm{GIH}) \neq \emptyset$.

Proof: The proof of this lemma follows easily from remark 3.2. The FIO system of the proof of lemma 3.7 can be considered as a FIH system. Since $\mathscr{L}(\mathrm{GIH}) \subseteq \mathscr{L}(\mathrm{GIO})$, the result follows. □

THEOREM 3.5: $\mathscr{L}(\mathrm{FIO})$ *and* $\mathscr{L}(\mathrm{GIO})$ *are incomparable but not disjoint.*

Proof: The result follows from lemma 3.5 and lemma 3.7, and from the fact that every IO system can be considered to be a FIO system as well as a GIO system. □

THEOREM 3.6: *$\mathscr{L}$(FIH) and $\mathscr{L}$(GIH) are incomparable but not disjoint.*

Proof: This result follows from lemma 3.6 and 3.7 and from the fact that every IH system can be considered to be a FIH system as well as a GIH system. □

THEOREM 3.7: $\mathscr{L}(\text{IH}) \subsetneqq \mathscr{L}(\text{IO})$.

Proof: The inclusion follows from remark 3.2. To prove the strict inclusion consider the IO system $G=(\mathscr{H}, u_{\text{in}})$ where $\mathscr{H}=\{H_1, H_2\}$ with:

$$H_1=(\{v_1, v_2\}, \emptyset),$$
$$H_2=(\{v_1, v_2\}, \{\{v_1, v_2\}\})$$

and:

$$u_{\text{in}}=\{v_1, v_2\}.$$

Then clearly $L(G)$ is the set of all graphs of the forms • • and •——•. By an argument, very similar to that of the proof of theorem 3.3 it follows that $L(G) \notin \mathscr{L}(\text{IH})$. □

THEOREM 3.8: $\mathscr{L}(\text{GIH}) \subsetneqq \mathscr{L}(\text{GIO})$.

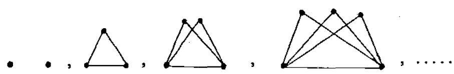
Proof: The inclusion follows from remark 3.2. To prove the strict inclusion consider the GIO system $G=(\mathscr{H}, u_{\text{in}}, C)$ where $\mathscr{H}=\{\text{H}_1, \text{H}_2\}$ with:

$$H_1=(\{v_1, v_2\} \emptyset),$$
$$H_2=(\{v_1, v_2, v_3\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}),$$
$$u_{\text{in}}=\{v_1, v_2\}$$

and:

$$C=\{(u_{\text{in}}, u_{\text{in}})\}.$$

Then $L(G)$ is the set of all graphs of the form:



Now let $\overline{G}=(\overline{H}, \overline{\Gamma}, \overline{u}_{\text{in}}, \overline{C})$ be a GIH system with $L(\overline{G})=L(G)$. Since $L(\overline{G})$ contains the discrete graphs with two nodes, either u_{in} has only one node or it has two nodes. In the latter case there exists an edge e in $E_{\overline{H}}$ with $F_{\overline{H}}(e)=\overline{u}_{\text{in}}$. This means that $(\text{und}\ \overline{G})_e$ is of the form • •, and that no graph of the form  can be derived in $\overline{G}$: a contradiction. In the case that u_{in} contains only one node, there exist edges e_1, e_2, e_3 in $E_{\overline{H}}$ such that $(und\,\overline{G})_{e_1}$ is of the form $\overset{n_1}{\bullet}\quad\overset{n_2}{\bullet}$ with $\{n_2\} = \overline{u}_{in}$:

$$f_{\overline{H}}(e_1) \cap f_{\overline{H}}(e_2) = \overline{u}_{in} \qquad \text{and} \qquad (\overline{u}_{in}, f_{\overline{H}}(e_2) \cap f_{\overline{H}}(e_3)) \in \overline{C}.$$

If $f_{\overline{H}}(e_2) \setminus f_{\overline{H}}(e_1) = \emptyset$ then $f_{\overline{H}}(e_2) = \overline{u}_{in}$, $f_{\overline{H}}(e_2) \cap f_{\overline{H}}(e_3) = \overline{u}_{in}$, $(\overline{u}_{in}, \overline{u}_{in}) \in \overline{C}$ and it is easily seen that the discrete graph with 3 nodes can be derived in $\overline{G}$. [Gluing first $(und\,\overline{G})_{e_2}$ and $(und\,\overline{G})_{e_1}$ and then gluing $(und\,\overline{G})_{e_1}$ and the resulting graph.]

On the other hand, if $f_{\overline{H}}(e_2) \setminus f_{\overline{H}}(e_1) \neq \emptyset$ then by gluing $(und\,\overline{G})_{e_2}$ and $(und\,\overline{G})_{e_1}$ one clearly derives a graph that is not in $L(G)$. Hence all possible cases lead to a contradiction. □

THEOREM 3.9: *The diagram of figure* 3.3 *holds:*

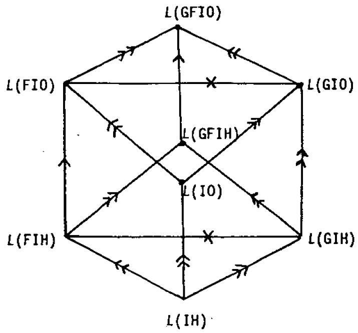


Figure 3.3.

where we denote A→— B if A ⊆ B, A →→— B if A ⊊ B and A—×— B if A and B are incomparable but not disjoint. □

IV. FINITE GRAPH-AUTOMATA

Although there is an analogy between intersections in systems we consider and states in finite automata, this analogy cannot be pushed too far. When one considers stades (nodes) in the transition graph of a finite automaton as intersections (of edges) then these are very simple intersections: they consist of one node only. In our systems we may have intersections of arbitrary cardinality between arbitrary many edges (graphs). This implies that in general if an intersection involves m edges (graphs) then the pairwise intersections of these edges are not independent. To remove this obstacle we will equip our systems *explicitly* with states — a state being now an abstract entity remembering a specific information about the derivation performed so far. As usual, we will consider systems with a finite number of states only. Such systems are defined formally as follows.

DEFINITION 4.1: A *finite graph automaton*, abbreviated FGA, is a system $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$ where Σ is a set of graphs such that no two elements of Σ are isomorphic. Σ is called *the alphabet of* $\mathscr{A}$;

Q is a finite nonempty set, called the *set of states;*

q_0 is an element of Q, called the *initial state;*

F is a subset of Q, called the set of *final states;*

ρ is a function from Q into the set of all discrete graphs; and δ is a function from $Q \times \Sigma$ such that for each (q, H) in $Q \times \Sigma$, $\delta(q, H)$ is a finite set of elements of the form $(\bar{q}, \gamma_{in}, \gamma_{out})$ where $\bar{q} \in Q$ and γ_{in}, γ_{out} are injective homomorphisms from $\rho(q)$ into H and from $\rho(\bar{q})$ into H respectively. δ is called the *transition function.* □

Observe that in the above we do not require δ to be a total function.

DEFINITION 4.2: Let $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$ be a FGA. A triple (A, q, g) is a *trace for* $\mathscr{A}$ if $(A, \rho(q), g)$ is a trace over $\{\rho(q) \mid q \in Q\}$. □

DEFINITION 4.3: Let $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$ be a FGA and let (A, q, g) and $(B, \tilde{q}, \tilde{g})$ be traces for $\mathscr{A}$. (A, q, g) *directly derives* $(B, \tilde{q}, \tilde{g})$, denoted by $(A, q, g) \underset{\mathscr{A}}{\vdash} (B, \tilde{q}, \tilde{g})$, if there exists an H in Σ such that:

(1) there exists a triple $(\tilde{q}, \gamma_{in}, \gamma_{out})$ in $\delta(q, H)$,

(2.1) B is the gluing of A and H along $\rho(q)$ by g and γ_{in}, and:

(2.2) $\tilde{g}$ equals $h \circ \gamma_{out}$ where h is the natural injection of H into B.

By $\underset{\mathscr{A}}{\vdash}$ we denote the transitive and the reflexive closure of $\underset{\mathscr{A}}{\vdash}$.

If we have $(A, q, g) \underset{\mathscr{A}}{\vdash} (B, \tilde{q}, \tilde{g})$ then we say that (A, q, g) *derives* $(B, \tilde{q}, \tilde{g})$ *in* $\mathscr{A}$. □

DEFINITION 4.4: Let $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$ be a FGA. The language of $\mathscr{A}$, denoted $L(\mathscr{A})$ is defined by:

$$L(\mathscr{A})=\{M \mid (\rho(q_0), q_0, \mathrm{Id}_{\rho(q_0)}) \underset{\mathscr{A}}{\vdash} (M, q, g) \text{ where } q \in F \text{ and } (M, q, g) \text{ is a trace for } \mathscr{A}\}. \quad \square$$

Example 4.1: Let $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$ where:

$$\Sigma=\{H_1, H_2\},$$
$$H_1=(\{v_1, v_2, v_3\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}),$$
$$H_2=(\{v_1, v_2, v_3, v_4\}, \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_1\}\}),$$
$$Q=\{q_0, q_1, q_f\},$$
$$F=\{q_f\},$$

ρ is defined by:

$$\rho(q_0)=(\{n_1\}, \emptyset),\ \rho(q_1)=(\{n_1, n_2, n_3\}, \emptyset), \qquad \rho(q_f)=(\{n_1\}, \emptyset)$$

and δ is defined by:

$$\delta(q_0, H_1)=\{(q_1, \{(n_1, v_1)\}, \{(n_1, v_1), (n_2, v_2), (n_3, v_3)\})\},$$
$$\delta(q_1, H_1)=\{(q_f, \{(n_1, v_2), (n_2, v_1), (n_3, v_3)\}, \{(n_1, v_2)\})\},$$
$$\delta(q_1, H_2)=\{(q_1, \{(n_1, v_4), (n_2, v_1), (n_3, v_3)\}, \{(n_1, v_4), (n_2, v_2), (n_3, v_3)\})\}$$

and δ is undefined on the rest of $Q\times\Sigma$.

The sequence of derivation steps:

$$(\rho(q_0), q_0, \mathrm{Id}_{\rho(q_0)}) \underset{\mathscr{A}}{\vdash} (M^{(1)}, q_1, g^{(1)}) \underset{\mathscr{A}}{\vdash} (M^{(2)}, q_1, g^{(2)}) \underset{\mathscr{A}}{\vdash} (M^{(3)}, q_f, g^{(3)})$$

is depicted in figure 4.1.

We conclude this section by demonstrating that finite graph-automata generalize both GFO systems and GFIO systems.

THEOREM 4.1: *For every* GFO *system G there exists an equivalent* FGA $\mathscr{A}$.

Proof: Let $G=(\mathscr{H}, H_{\mathrm{in}}, \mathscr{H}_{\mathrm{fin}}, C)$ be an arbitrary GFO system and let $\mathscr{H}=\{H_1, H_2, \ldots, H_n\}$ with $H_{\mathrm{in}}=H_1$. Let Σ be a set of representatives of the isomorphism classes of $\mathscr{H}$. Hence Σ contains no two elements that are isomorphic to each other. For each H_i in $\mathscr{H}$ let $\hat{H}_i$ denote the representant in Σ of the isomorphism class of H_i and let h_i denote the corresponding isomorphism from H_i into $\hat{H}_i$. Let $Q=\{q_{ij} \mid V_{H_i} \cup V_{H_j} \neq \emptyset\} \cup \{q_0\}$. Let $\rho(q_0)=(\{v\}, \emptyset)$ and for each $q_{ij}\in Q\setminus\{q_0\}$ let $\rho(q_{ij})$ be the graph $(V_{H_i}\cap V_{H_j}, \emptyset)$.

δ is defined as follows:

(1) For each j such that $(H_1, H_j)\in C$, let $(q_{1j}, \gamma_{\mathrm{in}}, \gamma_{\mathrm{out}})\in\delta(q_0, \hat{H}_1)$ where γ_{in} is an arbitrary injection of $\rho(q_0)$ in $\hat{H}_1$ and γ_{out} is the restriction of h_1 to $V_{H_1}\cap V_{H_j}$.

(2) For each i, j, k such that $(H_i, H_j)\in C$ and $V_{H_i}\cap V_{H_k}\neq\emptyset$, let $(q_{jk}, \gamma_{\mathrm{in}},$

$\gamma_{\text{out}}) \in \delta(q_{ij}, H_j)$ where γ_{in} and γ_{out} are the restriction of h_j to $V_{H_i} \cap V_{H_j}$ and to $V_{H_j} \cap V_{H_k}$ respectively.

Finally let $F = \{ q_{jk} \mid q_{jk} \in Q \text{ and } H_j \in \mathscr{H}_{\text{fin}} \}$ and let $\mathscr{A} = (\Sigma, Q, q_0, F, \rho, \delta)$.

We show that $L(G) = L(\mathscr{A})$.

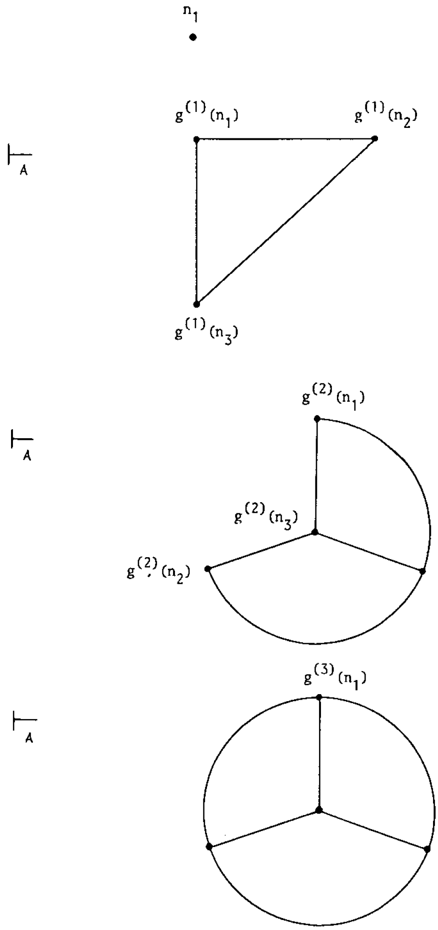


Figure 4.1

Let (A, H_i, g_1) and (B, H_j, g_2) be traces for G and assume that $(A, H_i g_1) \vdash_G (B, H_j, g_2)$. Let $\bar{g}_1$ be the restriction of g_1 to $V_{H_i} \cap V_{H_j}$ and let $\bar{B}$ be the gluing of A and $\hat{H}_j$ along $V_{H_i} \cap V_{H_j}$ by $\bar{g}_1$ and the restriction of h_j to $V_{H_i} \cap V_{H_j}$. (Hence there exists an isomorphism μ from B into $\bar{B}$.) Let k be such that $q_{jk} \in Q$. Then it follows from the construction of δ that we have $(A, q_{ij}, \bar{g}_1) \vdash_{\mathscr{A}} (\bar{B}, q_{jk}, \bar{g}_2)$ where $\bar{g}_2$ is the restriction to $\rho(q_{jk})$ of $\mu \circ g_2 \circ h_j^{-1}$. On the other hand, if $(A, q_{ij}, \bar{q}_1)$ and $(\bar{B}, q_{jk}, \bar{g}_2)$ are traces for $\mathscr{A}$ such that $(A, q_{ij}, \bar{g}_1) \underset{\mathscr{A}}{\vdash} (\bar{B}, q_{jk}, \bar{g}_2)$ then it is easily seen that there exist traces (A, H_i, g_1) and (B, H_j, g_2) for G such that $(A, H_i, g_1) \underset{G}{\vdash} (B, H_j, g_2)$, $\bar{g}_1$ is the restriction to $V_{H_i} \cap V_{H_j}$ of g_1 and there exists an isomorphism μ from B into $\bar{B}$ such that $\bar{g}_2$ is the restriction to $\rho(q_{jk})$ of $\mu \circ g_2 \circ h_j^{-1}$.

We conclude that for each graph M we have that there exists an integer $k \leqq n$ and a trace (M, q_{jk}, g) for $\mathscr{A}$ such that $(\rho(q_0), q_0, \mathrm{Id}_{(q_0)}) \underset{\mathscr{A}}{\vdash} (M, q_{jk}, g)$ in $n+1$ steps if and only if there exists a trace (M, H_j, g) for G with $(H_{in}, H_{in}, \mathrm{Id}_{V_{H_{in}}}) \overset{*}{\underset{G}{\vdash}} (M, H_j, g)$ in n steps. The result now easily follows from the definition of F. □

THEOREM 4.2: *For every* GFIO *system G there exists an equivalent* FGA $\mathscr{A}$.

Proof: Let $G=(\mathscr{H}, u_{in}, I_{fin}, C)$ be an arbitrary GFIO system and let $\mathscr{H}=\{H_1, H_2, \ldots, H_n\}$. Let Σ be a set of representatives of the isomorphism classes of $\mathscr{H}$. For each H_i in $\mathscr{H}$ let $\hat{H}_i$ denote the representant of the class of H_i and let h_i denote the corresponding isomorphism from H_i into $\hat{H}_i$. Let:

$$Q=\{q_{ij} \mid V_{H_i} \cup V_{H_j} \neq \emptyset\} \cup \{q_0\}, \qquad F=\{q_{ij} \mid V_{H_i} \cap V_{H_j} \in I_{fin}\}$$

and let ρ be defined by:

$$\begin{cases} \rho(q_0)=(u_{in}, \emptyset) \\ \rho(q_{ij})=(V_{H_i} \cap V_{H_j}, \emptyset) \end{cases}$$

δ is defined as follows:

(1) For each i, j, k such that $V_{H_i} \cap V_{H_j} = u_{in}$ and $(u_{in}, V_{H_j} \cap V_{H_k}) \in C$, let $(q_{jk}, \gamma_{in}, \gamma_{out}) \in \delta(q_0, H_j)$ where γ_{in} and γ_{out} are the restrictions of h_j to u_{in} and $V_{H_j} \cap V_{H_k}$ respectively.

(2) For each i, j, k such that $(V_{H_i} \cap V_{H_j}, V_{H_j} \cap V_{H_k}) \in C$, let $(q_{jk}, \gamma_{in}, \gamma_{out}) \in \delta(q_{ij}, \hat{H}_j)$ where γ_{in} and γ_{out} are the restrictions of h_j to $V_{H_i} \cap V_{H_j}$ and $V_{H_j} \cap V_{H_k}$ respectively.

Let $\mathscr{A}=(\Sigma, Q, q_0, F, \rho, \delta)$.

We show that $L(\mathscr{A})=L(G)$.

Let (A, H_i, r_1, g_1) and (B, H_j, r_2, g_2) be extended traces for G and assume that $(A, H_i, r_1, g_1) \underset{G}{\vdash} (B, H_j, r_2, g_2)$. Let $\bar{g}_1$ be the restriction of g_1 to r_1 and let $\bar{B}$ be the gluing of A and $\hat{H}_j$ along $V_{H_i} \cap V_{H_j}$ by $\bar{g}_1$ and the restriction of H_j to r_1. (Hence there exists an isomorphism μ from B into $\bar{B}$.) Let k be such that $q_{jk} \in Q$. Then it follows from the construction of δ that we have $(A, q_{ij}, \bar{g}_1) \underset{\mathscr{A}}{\vdash} (\bar{B}, q_{jk}, \bar{g}_2)$ where $\bar{g}_2$ is the restriction to $\rho(q_{jk})$ of $\mu \circ g_2 \circ h_j^{-1}$.

On the other hand, if $(A, q_{ij}, \bar{g}_1)$ and $(\bar{B}, q_{jk}, \bar{g}_2)$ are traces in $\mathscr{A}$ such that $(A, q_{ij}, \bar{g}_1) \underset{\mathscr{A}}{\vdash} (\bar{B}, q_{jk}, \bar{g}_2)$ then it is easily seen that there exist extended traces (A, H_i, r_1, g_1) and (B, H_j, r_2, g_2) for G such that $(A, H_i, r_1, g_1) \underset{G}{\vdash} (B, H_j, r_2, g_2)$. $\bar{g}_1$ is the restriction to r_1 of g_1 and there exists an isomorphism μ from B into $\bar{B}$ such that $\bar{g}_2$ is the restriction to $\rho(q_{jk})$ of $\mu \circ g_2 \circ h_j^{-1}$.

We conclude that for each graph M we have that there exists a trace (M, q_{jk}, g) for $\mathscr{A}$ such that $(\rho(q_0), q_0, \mathrm{Id}_{\rho(q_0)}) \underset{\mathscr{A}}{\vdash} (M, q_{jk}, g)$ in $n+1$ steps if and only if there exist a graph $H \in \mathscr{H}$, traces $(H, u_{in}, \mathrm{Id}_{u_{in}})$ and (M, u, g) with $u = V_{H_j} \cap V_{H_k}$ and $(H, u_{in}, \mathrm{Id}_{u_{in}}) \underset{\mathscr{A}}{\vdash} (M, u, g)$ in n steps. The result now easily follows from the definition of F. □

V. DISCUSSION

Starting from the observation that the notion of a hypergraph generalizes the notion of a graph, we have shown that if one equips a hypergraph with an "ordinary" graph structure, then this hypergraph naturally defines a family of graphs (a graph language). We have presented here a number of systems defining graph languages. The major objective of this paper was to introduce a formalism adequate to discuss these systems, to illustrate them by examples and to compare the classes of languages they generate.

As far as the comparison of the generative power of the systems is concerned, the basic missing results are the following.

(i) For the edge-based approach we do not know whether the inclusion $\mathscr{L}(\mathrm{GFH}) \subseteq \mathscr{L}(\mathrm{GFO})$ is strict.

(ii) For the intersection-based approach we do not know whether the inclusion $\mathscr{L}(\mathrm{GFIH}) \subseteq \mathrm{L}(\mathrm{GFIO})$ is strict.

(iii) We do not know whether the inclusions $\mathscr{L}(\mathrm{GFO}) \subseteq \mathscr{L}(\mathrm{FGA})$ and $\mathscr{L}(\mathrm{GFIO}) \subseteq \mathscr{L}(\mathrm{FGA})$ are strict.

In our opinion four major relationships to be considered are the following relationships between the edge-based approach and the intersection-based approach.

(i) The relationship between $\mathscr{H}$(H) and $\mathscr{L}$(IH).
(ii) The relationship between $\mathscr{L}$(O) and $\mathscr{L}$(IO).
(iii) The relationship between $\mathscr{L}$(GFH) and $\mathscr{L}$(GFIH).
(iv) The relationship between $\mathscr{L}$(GFO) and $\mathscr{L}$(GFIO).

Theorem 2.1 settles (i). We are not able to settle (iii) and (iv) and (ii) is settled by the following result.

THEOREM 5.1: $\mathscr{L}$(O) and $\mathscr{L}$(IO) are incomparable but not disjoint.

Proof: To show that $\mathscr{L}(O)\setminus\mathscr{L}(IO)\neq\emptyset$ the argument of lemma 2.2 can be used. To see that $\mathscr{L}(IO)\setminus\mathscr{L}(O)\neq\emptyset$ consider the IO system:

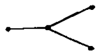
$$G=(\mathscr{H}, u_{in}) \qquad \text{where} \quad \mathscr{H}=\{H_1, H_2\}$$

with:

$$H_1=(\{v_1, v_2, v_3, v_4\}, \{\{v_1, v_4\}, \{v_2, v_4\}, \{v_3, v_4\}\}),$$
$$H_2=(\{v_4, v_5, v_6, v_7\}, \{\{v_4, v_5\}, \{v_5, v_6\}, \{v_6, v_7\}\})$$

and:

$$u_{in}=\{v_4\}.$$

Then clearly $L(G)$ contains the graphs of the forms  (a)

and (b), and the graphs in $L(G)$ that are not of this form have at least 7 nodes. It follows that if $\overline{G}=(\overline{\mathscr{H}}, \overline{H}_{in})$ is a O system with $L(G)=L(\overline{G})$ then $\overline{H}_{in}$ is of one the forms (a) or (b). Both cases lead to a contradiction since a graph of the form (a) cannot be derived from a graph of the form (b) and vice versa. That $\mathscr{L}$(O) and $\mathscr{L}$(IO) are not disjoint follows from theorem 2.1 and remark 3.2. □

REFERENCES

1. V. CLAUS, H. EHRIG, G. ROZENBERG, *Graph Grammars and their Application to Computer Science and Biology*, Springer-Verlag, Lecture notes in Computer Science, Vol. 73, Berlin-Heidelberg-New York, 1979.
2. H. EHRIG, *Introduction to the Algebraic Theory of Graph Grammars*, pp. 1-69, in [1].
3. M. NAGL, *A Tutorial and Bibliographical Survey on Graph Grammars*, pp, 70-126, in [1].
4. M. NAGL, *Graph-Grammatiken: Theorie, Anwendungen, Implementierung*, Vieweg-Verlag, Wiesbaden, 1979.