Memorial
University of Newfoundland

Technical Report #9105

The Gallai-Edmonds Algebra of Graphs

by

Miklós Bartha

Department of Computer Science
Memorial University of Newfoundland
St. John's, NF, Canada, A1C 5S7

and

Éva Gombás

Department of Computer Science, A. József University
Szeged, 6720, Hungary

May 1991

The Gallai-Edmonds algebra of graphs

Miklós Bartha *
Department of Computer Science, Memorial University of Newfoundland
St. John's, Newfoundland, Canada A1C 5S7

Éva Gombás
Department of Computer Science, A. József University
Szeged, 6720, Hungary

Abstract: Graphs with labeled extrenal vertices, called open graphs, are introduced to model the switching behavior of molecules. Open graphs are given the structure of an algebra sorted by the set of all finite subsets of the alphabet used for labeling the external vertices. The constants of this algebra are the star graphs and the only operation is composition, which practically merges two graphs along their external vertices wearing the same label. It is shown that every open graph can be built up from the constant graphs using composition. Composition is introduced as a different operation over the collection of open graphs having a perfect internal matching. The resulting structure is called the Gallai-Edmonds algebra and is specified as a homomorphic image of the algebra of open graphs.

Introduction

For more than a decade chemists have been trying to develop a molecular computer in which most of the switching is done by molecular devices. The phenomenon "molecular switching" has been suggested as an extrapolation of the dominant trend in semiconductor technology to reduce the size of electronic switching elements in computers down to a physically realizable minimum.

*On leave from Department of Computer Science, A. József University, Szeged, 6720, Hungary

It has been known for a long time that certain properties of a chemical compound can be studied effectively through the topological model of the molecule. This model is simply the underlying undirected graph of the molecule in which atoms are represented by vertices and chemical bonds by edges. The purpose of this paper is to show that the switching behavior of a molecule, too, is a property that can be successfully pursued by studying this simple model.

Many organic compounds, e.g. most hydrocarbons, are such that in their molecules one can recognize an alternating pattern of single and double bonds. Patterns of this kind are known as conjugated systems in organic chemistry, and the graphs of molecules possessing conjugated systems are called Hückel graphs. For simplicity, let us restrict our discussion to hydrocarbons, i.e. let us assume that our molecules consist of carbon and hydrogen atoms only. Then the "perfect" molecules, in which every atom is a link in an appropriate conjugated system, are those having the property that every carbon atom has a unique neighbor to which it is connected by a double bond. Strictly speaking, hydrogen atoms are never part of a conjugated system, because it is implicitly understood by the chemical description of such systems that the process of switching all the bonds within the system to the opposite results in another comformational state of the molecule. Hydrogen atoms, in contrast, are connected to their unique neighbor by a constant single bond. Bearing this in mind, we shall ignore hydrogen atoms in Hückel graphs. Then, in terms of graph theory, the Hückel graph of a perfect molecule is a graph having a perfect matching.

A relevant property of Hückel graphs is that, after suppressing the irrelevant hydrogen atoms in them, they will generally not contain external vertices, that is, vertices with degree one. This is a disadvantage, since the molecules will not have an "interface" by which they could be observed or manipulated from the outside world. There is no reason to stick back the hydrogen atoms as external vertices, because these atoms would only represent "dead ends" in the molecule. However, if we do not have a reasonable interface, then the phenomena that can be studied on Hückel graphs are confined to those of resonance theory as described in [13]. A suitable interface can be provided for a molecule by breaking the homogeneous hydrocarbon structure at its border and inserting some "alien" atoms or molecules that can serve as electron donors or acceptors for the internal part of the molecule. For some examples, the reader is referred to [6].

The abstract model of a hydrocarbon molecule supplied with an interface is a *soliton graph*. As introduced in [7], a soliton graph is a finite undirected graph G without loops and multiple edges such that the degree of every vertex in G is at most 3. A *state* of a soliton graph G is a matching M in G which covers all the internal vertices of G. We wish to emphasize that M need not be a perfect matching, since the external vertices of G need not be covered by M. Any of the external vertices, however, might be covered by an edge in M which covers an internal vertex in the first instance. Such a matching M is called a perfect internal matching.

There is one subtlety regarding the representation of the interface by external vertices in soliton graphs which is not captured by the above description. This is the labeling of the external vertices. In fact, we cannot consider two soliton graphs to be isomorphic if they are just isomorphic in the usual graph theoretic sense. We have to provide a unique label for each external vertex and require that an isomorphism between two soliton graphs preserves the labeling.

The labeling of the external vertices provides another source of benefits, too. Given two disjoint graphs with their external vertices being labeled, we can define the composite of the two graphs by gluing them together at those external vertices that have the same label. We shall see in Sections 3 and 4 that the operation of composition, together with the star graphs as constants, allows an algebraic treatment for general graphs and also for graphs having a perfect internal matching. A similar algebraic method has been introduced in [4] to study graph rewritings of finite directed hypergraphs. Our graph algebra is also closely related to flowchart scheme algebras, see e.g. [3, 5, 9].

The switching behavior of molecules is modeled by soliton automata. The underlying object of a soliton automaton is a perfect soliton graph, that is, a soliton graph G which has at least one state (i.e. perfect internal matching). The states of the automaton are those of the graph G, and the input symbols are pairs of external vertices of G. A transition of the automaton in a current state is induced by propagating a particle (electron, soliton) from the first vertex of the input to the second vertex along an alternating walk. The new state is obtained by switching all the bonds to the opposite throughout the walk in a dynamic way, so that each bond will change immediately after traversing the corresponding edge.

For more details, see [7].

This paper is organized as follows. In Section 2 we provide a short summary of the notation and terminology used, and restate the Gallai-Edmonds Structure Theorem for maximum internal matchings in general graphs. Section 3 describes the many-sorted algebra $\mathcal{G}$ of graphs with labeled external vertices. It also contains a discussion about how to choose the sorting set of this algebra in an optimal way. The main result of the paper is the construction of the Gallai-Edmonds algebra $\mathcal{S}$ of graphs having a perfect internal matching. The algebra $\mathcal{S}$ is characterized as a homomorphic image of $\mathcal{G}$. These results are contained in Section 4. Finally, in Section 5, we return to our motivating problem and discuss the relationship between the syntax and semantics of soliton automata in terms of equational axioms.

2. Preliminaries

In this section we review some of the basic notions concerning graphs. Our notation and terminology will be compatible with that of [13] except that the words "point" and "line" will be replaced by "vertex" and "edge", respectively.

By a graph we mean a finite undirected graph in the most general sense, i.e. with multiple edges and loops allowed. The empty graph, which has no vertices and no edges, will also be allowed. For a graph G, $V(G)$ and $E(G)$ will denote the set of vertices and the set of edges of G, respectively. An edge $e \in E(G)$ connects two vertices $v_1, v_2 \in V(G)$, which are said to be *adjacent* in G. The vertices v_1 and v_2 are called the *endpoints* of e, and we say that e is *incident with* v_1 and v_2. If $v_1 = v_2$, then e is called a *loop* around v_1.

The *degree* of a vertex v in graph G is the number of occurrences of v as an endpoint of some edge in $E(G)$. According to this definition, every loop around v contributes two occurrences to the count. The vertex v is called *external* if its degree is one, *internal* if its degree is greater than one and *isolated* otherwise. An edge $e \in E(G)$ is said to be an *external edge* if one of its endpoints is an external vertex. *Internal edges* are those that are not external. The sets of external and internal vertices of G will be denoted by $\mathrm{Ext}(G)$ and $\mathrm{Int}(G)$, respectively.

A *matching* M of graph G is a subset of $E(G)$ such that no vertex of G occurs more than once as an endpoint of some edge in M. Again, it is understood that loops are not allowed to participate in M. The endpoints of the edges contained in M are said to be covered by M. A *maximum matching* is one that covers a maximum number of vertices. A matching is *perfect* if it covers all of $V(G)$. The terms *maximum internal matching* and *perfect internal matching* are defined analogously, with the understanding that only the vertices of $\text{Int}(G)$ are required to be covered by these matchings. The number of vertices (internal vertices) covered by a maximum matching (maximum internal matching) of G will be denoted by $2\nu(G)$ (respectively, $\eta(G)$). By the *deficiency* (*internal deficiency*) of G we mean the number $\|V(G)\| - 2\nu(G)$ (respectively, $\|\text{Int}(G)\| - \eta(G)$), where for a set X, $\|X\|$ denotes the cardinality of X. An edge $e \in E(G)$ is *forbidden* if e is not contained in any maximum internal matching of G. Note that this terminology has originally been introduced for plain maximum matchings in [13], but we shall use it in a different context here.

If G is a graph and X is any subset of $V(G)$, then $\Gamma(X)$ denotes the set of all vertices in $V(G)$ which are adjacent to at least one vertex in X. The subgraph of G induced by X, denoted $G[X]$, is the restriction of G to X. For short, the graph $G[V(G) - X]$ will be denoted by $G - X$.

A graph G is *factor-critical* if $G - v$ has a perfect matching for any $v \in V(G)$. Note that G itself does not have a perfect matching if it is factor-critical. Moreover, G has no external vertices. Indeed, if $v \in \text{Ext}(G)$ and v' is the vertex adjacent to v, then in $G - v'$ the vertex v becomes isolated and thus unmatchable. A maximum matching of a factor-critical graph is called a *near-perfect* matching. Clearly, the deficiency of every factor-critical graph is 1.

Let G be a bipartite graph with bipartition (V_1, V_2). The *surplus* of G viewed from V_1 is the number

$$\min\{(\|\Gamma(X)\| - \|X\|) \mid X \subseteq V_1, X \neq \emptyset\}.$$

One of the most useful results in matching theory is the so called Gallai-Edmonds Structure Theorem found independently by Gallai [10, 11] and Edmonds [8]. The reader is referred to Section 3.2 in [13] for a detailed discussion of this theorem, which characterizes the structure of maximum matchings in graphs. A counterpart of the Gallai-Edmonds

Structure Theorem for maximum internal matchings, Theorem 2.1 below, has been stated and proved in [1].

For any graph G, let $D(G)$ denote the set of all internal vertices in G that are not covered by at least one maximum internal matching. Furthermore, let $A(G)$ be the set of vertices (internal or external) in $V(G) - D(G)$ adjacent to at least one vertex in $D(G)$. Finally, let $C(G) = V(G) - A(G) - D(G)$. Then we have

Theorem 2.1 *The following five statements hold for the decomposition $D(G)$, $A(G)$, $C(G)$:*

(i) *the components of the subgraph induced by $D(G)$ are factor-critical,*

(ii) the subgraph induced by $C(G)$ has a perfect internal matching,

(iii) the bipartite graph obtained from G by deleting the vertices of $C(G)$ and the edges spanned by $A(G)$ and by contracting each component of $D(G)$ to a single vertex has positive surplus (as viewed from $A(G)$),

(iv) if M is any maximum internal matching of G, it contains a near-perfect matching of each component of $D(G)$, a perfect internal matching of $C(G)$ and matches all vertices of $A(G)$ with vertices in distinct components of $D(G)$,

(v) $\eta(G) = \|\mathrm{Int}(G)\| - c(D(G)) + \|A(G)\|$, where $c(D(G))$ denotes the number of components of the graph spanned by $D(G)$.

As the first result of this paper we generalize Theorem 2.1 for graphs containing loops.

Proposition 2.2 *Theorem 2.1 holds also for graphs containing loops.*

Proof. First suppose that G contains only one loop e around some vertex $v \in V(G)$. Replace e with a triangle by inserting two new vertices v_1 and v_2 on e. Let G' denote the resulting graph.

Let us locate the vertices v, v_1 and v_2 in the decomposition $D(G')$, $A(G')$, $C(G')$ of $V(G')$ by Theorem 2.1. If one of v, v_1 and v_2 is in $D(G')$, then all of them must be in $D(G')$, otherwise condition (iii) would be violated or a factor-critical graph would have an external vertex. Consequently, v, v_1 and v_2 are in the same component of the subgraph of G' induced by $D(G')$. Moreover, the deletion of v_1 and v_2 from e preserves the factor-critical property of this component.

If any of v, v_1 and v_2 is in $A(G')$, then it must be v, so that v_1 and v_2 are in $C(G')$. In this case, too, the deletion of v_1 and v_2 from e is harmless, showing that e is a loop around $v \in A(G)$.

If all of v, v_1 and v_2 are in $C(G')$, then the edges connecting v with v_1 and v_2 in G' are clearly forbidden. In this way we can associate a perfect internal matching of $C(G)$ with each perfect internal matching of $C(G')$ by deleting the vertices v_1 and v_2 from e.

If G has several loops, then repeat the above argument for all the loops to see that the desired decomposition of G is the restriction of the decomposition $D(G')$, $A(G')$, $C(G')$ to $V(G)$. □

3. The algebra of graphs

In this section we show how graphs can be given a many-sorted algebraic structure based on a distinctive treatment of their external vertices.

Recall from the Introduction that the external vertices of a graph G are intended to provide an interface for the internal part of G. To establish this interface in a proper way, it is necessary to assign a unique label to each external vertex of G. Since we are dealing with finite graphs, any countable infinite set is sufficiently large to choose the labels from it. We shall use the set N_+ of all positive integers for this purpose. A graph together with a labeling of its external vertices is called an *open graph*. Thus, an open graph is a pair (G, ρ), where G is a graph, called the *base graph* of G, and ρ is an injection of $\mathrm{Ext}(G)$ into N_+. If the image of the mapping ρ is $A \subseteq N_+$, then we say that (G, ρ) is *of sort* A.

Two open graphs (G_1, ρ_1) and (G_2, ρ_2) of the same sort are *isomorphic* if there exists a bijection $\iota : V(G_1) \to V(G_2)$ such that ι is an isomorphism between G_1 and G_2 in the usual sense and ι preserves the labeling, i.e. $\rho_1(v) = \rho_2(\iota(v))$ for every $v \in \mathrm{Ext}(G_1)$. For example, the two open graphs of Fig. 1 are not isomorphic, although their base graphs are such.

When drawing open graphs we shall, as in Fig. 1, omit the bullets representing the external vertices in order to emphasize the non-terminal nature of these vertices. Also, we shall identify each external vertex with its unique label if no danger of confusion arises. With this assumption we can as well identify an open graph with its base graph. As a

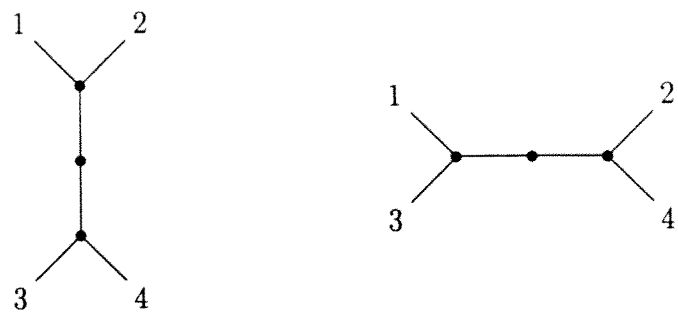

Figure 1: Two non-isomorphic open graphs

practical notational shorthand, rather than writing that (G, ρ) is an open graph of sort A, we shall just write that $G : A$ is a graph, bearing in mind the labeling ρ.

The algebra $\mathcal{G}$ of open graphs is sorted by the set U of all finite subsets of N_+. For any sort $A \in U$, the underlying set $\mathcal{G}_A$ of $\mathcal{G}$ is the set of all isomorphism classes of open graphs of sort A. To simplify the terminology, we shall be dealing with isomorphism classes of graphs by choosing appropriate representatives of them.

There is only one general binary operation defined in the algebra $\mathcal{G}$. This operation is called *composition* and is denoted by "$\cdot$". Composition is a mapping

$$\cdot : \mathcal{G}_A \times \mathcal{G}_B \to \mathcal{G}_{A \ominus B} \text{ for all } A, B \in U,$$

where $A \ominus B$ denotes the symmetric difference of sets A and B. Intuitively, composing two open graphs means pasting them together along those "colliding" pairs of external vertices that are labeled by the same number. In the extreme case when this collision process becomes circular, composition will result in an isolated vertex by definition.

Formally, let $G : A$ and $H : B$ be two graphs. Without loss of generality we may assume that $V(G)$ and $V(H)$ are such that they have their external vertices $A \cap B$ in common, but otherwise they do not intersect each other. The composite $L : A \ominus B$ of G and H is constructed through the following four steps.

(1) Let $L_1 = G \cup H$, that is, $V(L_1) = V(G) \cup V(H)$ and $E(L_1) = E(G) \cup E(H)$.

(2) For any two vertices $v_1, v_2 \in V(G) \cup V(H) - (A \cap B)$ which are connected in L_1 by

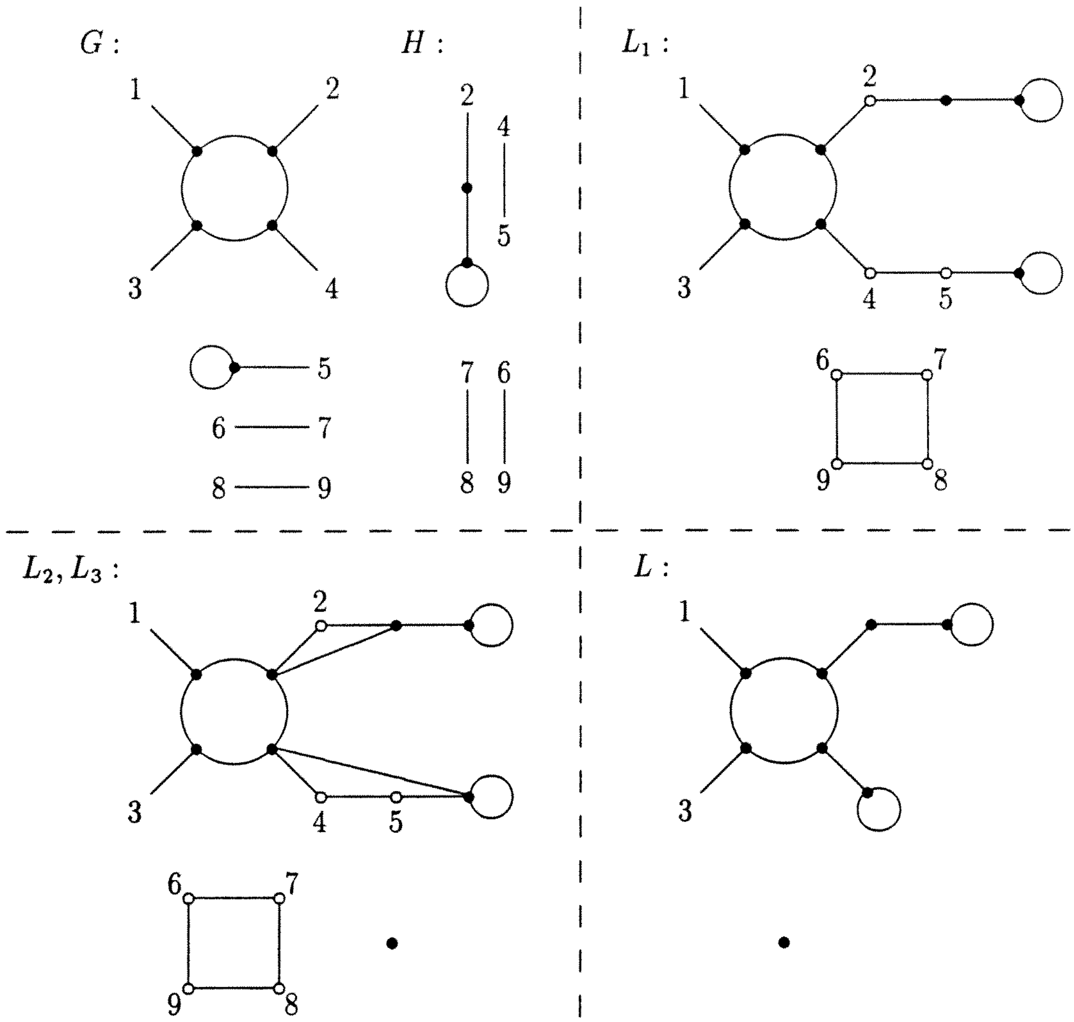

Figure 2: The four steps of composition

a path consisting only of external edges with endpoints in $A \cap B$, add a new edge connecting v_1 and v_2. Let L_2 be the resulting graph.

(3) For each cycle in L_2 containing only vertices in $A \cap B$, add a new isolated vertex to L_2. Let L_3 be the resulting graph.

Note that a cycle considered in step (3) has no multiple edges and its length is always even.

(4) $L = L_3 - (A \cap B)$.

An example showing the four steps of composition is provided in Fig. 2.

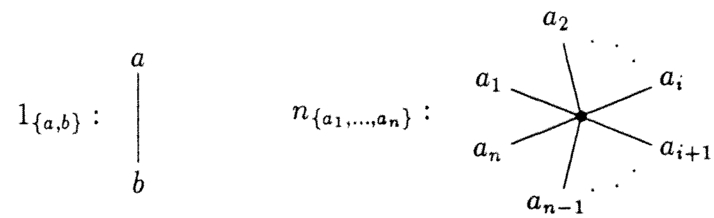

Figure 3: The constants of the algebra $\mathcal{G}$

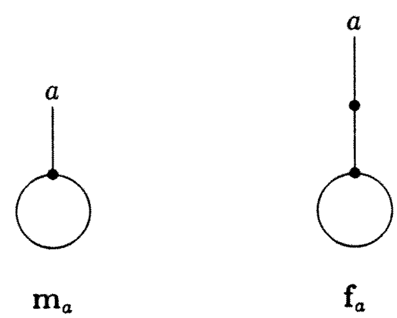

Figure 4: Two loops with different handles

The algebra $\mathcal{G}$ has the following infinite number of constants:

$0 : \emptyset$ stands for the empty graph;

$1_A : A$ for each two-element set $A \in U$;

$n_A : A$ for each positive integer $n \geq 2$ and set $A \in U$ such that $\|A\| = n$.

The interpretation of the constants is shown in Fig. 3. Note that for every $n \in N_+$, the base graph of n_A is the n-star $K_{1,n}$, cf. [13], hence all the labelings of the external vertices of this graph by the elements of A results in the same open graph.

The graphs

$$\mathbf{m}_a = 3_{\{a,b,c\}} \cdot 1_{\{b,c\}} \quad \text{and} \quad \mathbf{f}_a = 2_{\{a,b\}} \cdot \mathbf{m}_b$$

shown in Fig. 4 will be of special interest in the next section. The "handle" of $\mathbf{m}_a$ is the unique perfect internal matching of that graph as a singleton set, hence the external edge of $\mathbf{f}_a$ is forbidden.

Composition is sometimes too complex to deal with. We therefore introduce a pair

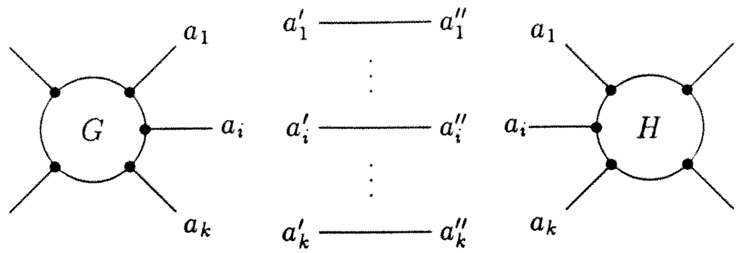

Figure 5: Deriving composition from sum and merge

of simpler operations as an equivalent substitute for composition in the algebra $\mathcal{G}$. The first operation is *sum*, which is the restriction of composition to operands having disjoint sorts. Sum is denoted by "+". The other operation is called *merge*. Merge is rather a collection of operations associating with each two-element set $B \in U$ and sort $A \supseteq B$ a unary operation

$$\updownarrow_B : \mathcal{G}_A \to \mathcal{G}_{A-B}.$$

If $G : A$ is a graph and $B = \{a, b\}$, then $\updownarrow_B G$ is obtained from G by merging the external edges incident with a and b in it. Again, in extreme cases like $G = 1_B$, $\updownarrow_B G$ introduces a single isolated vertex by definition. Instead of giving a formal definition of the operation merge, we invite the reader to check that

$$\updownarrow_B G = G \cdot 1_B. \tag{1}$$

Thus, merge is a derived operation in $\mathcal{G}$.

Let $G : A$ and $H : B$ be two graphs. If $A \cap B = \{a_1, \ldots, a_k\}$, then choose $D_1 = \{a'_1, \ldots, a'_k\}$ and $D_2 = \{a''_1, \ldots, a''_k\}$ from U in such a way that $D_1 \cap D_2 = \emptyset$ and $D_i \cap A = D_i \cap B = \emptyset$ for $i = 1, 2$. It is easy to see that

$$G \cdot H = {}_{i=1}^{k} \updownarrow_{\{a''_i, a_i\}} \left({}_{i=1}^{k} \updownarrow_{\{a_i, a'_i\}} \left(G + \sum_{i=1}^{k} 1_{\{a'_i, a''_i\}}\right) + H\right), \tag{2}$$

see also Fig. 5.

Equations (1) and (2) above show that in $\mathcal{G}$, composition is equivalent to the couple of derived operations sum and merge. We can also learn from (2) that the relabeling of an open graph of sort A according to a bijection $\chi : A \longrightarrow B$, too, is a derived operation in $\mathcal{G}$.

In the algebra $\mathcal{G}$, relabelings and composition are related in a characteristic way. For a bijection $\chi : A \longrightarrow B$ between sets $A, B \in U$, let $\mathcal{G}_\chi$ denote the relabeling $\mathcal{G}_A \longrightarrow \mathcal{G}_B$ induced by χ. Consider two bijections $\alpha : A \longrightarrow A'$ and $\beta : B \longrightarrow B'$ for sets $A, A', B, B' \in U$. The bijections α and β are said to be *compatible* if, for all $a \in A$ and $b \in B$,

$$\alpha(a) = \beta(b) \iff a = b \in A \cap B.$$

In this case we say that the relabelings $\mathcal{G}_\alpha$ and $\mathcal{G}_\beta$ are *allowable* for composition. For compatible bijections α and β, we define $\alpha \ominus \beta : A \ominus B \longrightarrow A' \ominus B'$ by

$$(\alpha \ominus \beta)(x) = \begin{cases} \alpha(x), & \text{if } x \in A \\ \beta(x), & \text{if } x \in B. \end{cases}$$

If $G : A$ and $H : B$ are graphs and $\alpha : A \longrightarrow A'$, $\beta : B \longrightarrow B'$ are compatible, then it is clear that

$$\mathcal{G}_{\alpha \ominus \beta}(G \cdot H) = \mathcal{G}_\alpha(G) \cdot \mathcal{G}_\beta(H).$$

This observation motivated the category theoretical definition of indexed algebras in [2].

Proposition 3.1 *The algebra $\mathcal{G}$ is generated by its constants.*

Proof. We have to prove that all open graphs can be built up from the constant graphs using composition or, equivalently, using the operations sum and merge. Every graph $G : A$ is the sum of its connected components, hence we can assume that G is connected. If G is empty, then $G = 0$. If G is not empty but it has no internal vertices, then G is either the constant 1_A or it is an isolated vertex, in which case $G = \updownarrow_{\{1,2\}} 1_{\{1,2\}}$.

Suppose now that G has at least one internal vertex. Consider the "skeleton" of G, which is a collection of star graphs, one n-star for each internal vertex in $V(G)$ of degree n. Obviously, the skeleton of G can be represented as the sum of appropriate constants in $\mathcal{G}$ and G can be reconstructed from its skeleton by adding the internal edges to it one-by-one using the operation merge. The details of this construction are straightforward and therefore omitted. □

By a graph expression we mean an expression built up from the constant and operation symbols of the algebra $\mathcal{G}$. Although merge is not a basic operation in $\mathcal{G}$, we shall nevertheless use $\updownarrow$ in graph expressions in accordance with equation (1).

Definition 3.2 A graph expression t is in *normal form* if

$$t =_{p\in\mathcal{P}}\updownarrow_p (c_1 + \ldots + c_n) + t',$$

where

— $n \geq 0$ and $c_i : A_i$ is a constant for every $i \in [n] = \{1, \ldots, n\}$. If $n = 0$, then $t = t'$ by definition.

— $\mathcal{P}$ is a set of pairwise disjoint two-element subsets of $\cup_{j\in J} A_j$, where J is the set of all subscripts j such that $c_j \neq 1_{A_j}$.

— $t' = \sum_{i=1}^{k} \updownarrow_{\{1,2\}} 1_{\{1,2\}}$ is the sum of k isolated points for some $k \geq 0$.

The normal form of graph expressions is akin to the normal form of flowchart scheme expressions defined in [9]. In fact, our intention was to introduce the algebra $\mathcal{G}$ as the symmetric counterpart of the algebra of flowchart schemes. For the description of the algebra of flowchart schemes, the reader is referred to [3, 5]. The operation sum has the same interpretation in the two algebras, while merge is the symmetric equivalent of the operation feedback in the algebra of schemes. General composition in $\mathcal{G}$ corresponds to a mixture of composition, sum and iteration (feedback) in scheme algebras. The only non-ideal components of the algebra $\mathcal{G}$ in the sense of [9] are the constants 1_A.

For a graph expression t, let $|t|$ denote the open graph which results from the evaluation of t in the algebra $\mathcal{G}$. The following theorem is an immediate consequence of the construction applied in the proof of Proposition 3.1.

Theorem 3.3 *For every open graph G there exists a graph expression t in normal form such that $G = |t|$. The normal form t is unique up to an appropriate relabeling of its constants.*

As opposed to the algebra of flowchart schemes, the sorting set U of the algebra $\mathcal{G}$ has been chosen too generously. To do the labeling more economically, we could fix the labels of a graph with n external vertices to be exactly the integers $1, 2, \ldots, n$. The sorting set of the family of open graphs labeled in this way is the set N of all nonnegative integers, each sort $n \in N$ corresponding to the set $[n] \in U$. As to the operations in the resulting contracted structure $\mathrm{Con}(\mathcal{G})$, composition will associate with each triple

(n, m, q) of nonnegative integers satisfying $q \leq \min(n, m)$ a binary operation

$$\cdot_{(n,m,q)} : \text{Con}(\mathcal{G})_n \times \text{Con}(\mathcal{G})_m \to \text{Con}(\mathcal{G})_{n+m-2q}.$$

If $G : n$ and $H : m$ are open graphs in Con($\mathcal{G}$), then $G \cdot_{(n,m,q)} H$ is obtained by first taking the disjoint union of G and H, then merging the edges adjacent to the last q external vertices of the two graphs, and finally, relabeling the remaining external vertices of H to $n - q + 1, \ldots, n - q + m - q$ in a monotone way. Composition, however, must be supported by all n-ary permutations for all $n \in N$ as unary operations in Con($\mathcal{G}$) in order to be able to adjust the desired labeling of the operands before composing them together.

Concerning constants in Con($\mathcal{G}$), there is a unique constant $1 : 2$ and there are constants $n : n$ for each $n \geq 2$, corresponding to the constants 1_A and n_A of $\mathcal{G}$, respectively.

Although the algebra Con($\mathcal{G}$) is more succinct than $\mathcal{G}$, it is less attractive because it lacks the original symmetry, namely the commutativity of composition, that $\mathcal{G}$ has by its definition. Moreover, Con($\mathcal{G}$) is rather inconvenient to work with, for in this algebra we have to keep "rotating" graphs in graph expressions (i.e. relabeling them by appropriate permutations) to make composition work. It is natural to ask: in what sense are the algebras $\mathcal{G}$ and Con($\mathcal{G}$) equivalent? This question has been answered in [2] in a category theoretical framework.

4. The Gallai-Edmonds algebra of graphs having a perfect internal matching

The Gallai-Edmonds Structure Theorem is especially well-suited to demonstrate the use our new algebraic method on it. In this section we shall see that the decomposition of the set of vertices of a graph G by Theorem 2.1 is in fact a decomposition of G itself as a graph according to the algebra $\mathcal{G}$.

A somewhat disturbing fact about Theorem 2.1 is that, although it concerns maximum internal matchings of graphs, the definition of factor-critical graphs is based on the concept of (plain) perfect matching. We want to be able to express the factor-critical property in our own framework, that is, in terms of perfect internal matchings. For, let $G : A$ be a graph and e be an edge in $E(G)$. By cutting G at the edge e we mean creating a new graph $cut(G, e) : A \cup B$ for some indefinite two-element set B disjoint from A, so that $\updownarrow_B (cut(G, e)) = G$. Thus, cut is a kind of inverse of the operation merge.

Let e_1 and e_2 be two distinct edges of a graph G. We say that e_1 and e_2 are *inherently incompatible* if neither of e_1 and e_2 is forbidden, but for every maximum internal matching M in G, $e_1 \in M$ iff $e_2 \notin M$. Observe that if e_1 and e_2 are inherently incompatible in G, then both e_1 and e_2 must have at least one endpoint in $\mathrm{Int}(G)$.

Proposition 4.1 *A graph G is factor-critical iff one of the following two conditions is satisfied:*

(a) *G is a single isolated vertex;*

(b) G has no isolated vertices and for every edge $e \in E(G)$, $cut(G,e)$ has a perfect internal matching. Moreover, the two new external edges created in $cut(G,e)$ are inherently incompatible.

Proof. *If.* Since an isolated vertex is a factor-critical graph, we can assume that condition (b) holds. Supposing that G has an external edge e, consider the external edges e_1 and e_2 created in $cut(G,e)$. One of e_1 and e_2 will have two external endpoints, contradicting the fact that these edges are inherently incompatible. Thus, G has no external vertices.

Let $v \in V(G)$ be arbitrary. By assumption, there exists an edge e incident with v in G. In $cut(G,e)$, let e_1 and e_2 be the two new external edges, so that e_1 be incident with v. Furthermore, let M be a perfect internal matching of $cut(G,e)$ such that $e_1 \in M$ and $e_2 \notin M$. Then $M - e_1$ is a perfect matching of $G - v$.

Only if. As noted in Section 2, G has no external vertices. Supposing that G is not a single isolated vertex, let $e \in E(G)$ be arbitrary. Let v_1 and v_2 be the two endpoints of e and denote by e_1 and e_2 the two external edges of $cut(G,e)$ incident with v_1 and v_2, respectively. Reversing the argument in the second paragraph of the *If* part of the proof shows that $cut(G,e)$ has a perfect internal matching. Assume indirectly that e_1 and e_2 are not inherently incompatible. Since G itself does not have a perfect (internal) matching, one of e_1 and e_2 must be forbidden. Say e_1 is such. Then $G - v_1$ cannot have a perfect matching. Indeed, if M were a perfect matching of $G - v_1$, then $M \cup e_1$ would be a perfect internal matching of $cut(G,e)$, contradicting the fact that e_1 is forbidden. □

Note that the graph consisting of a single isolated vertex is the only factor-critical graph which has a perfect internal matching. We say that a graph $G : A$ is *internally factor-critical* if $G - A$ is factor-critical.

The rephrasing of Theorem 2.1 in terms of the algebra $\mathcal{G}$ is the following.

Theorem 4.2 *If $G : A$ is a graph, then there exist graphs $\hat{C}(G) : A \cup P$, $\hat{A}(G) : P \cup Q$ and $\hat{D}(G) : Q$ for some sets $P, Q \in U$ such that $G = \hat{C}(G) \cdot \hat{A}(G) \cdot \hat{D}(G)$ and, furthermore:*

- *(a) $\hat{D}(G) = D_1 + \ldots + D_k$ for some internally factor-critical graphs $D_i : Q_i$ with $\cup_{i=1}^{k} Q_i = Q$;*
- *(b) $\hat{A}(G) = (n_1)_{R_1} + \ldots + (n_m)_{R_m}$ is the sum of constant graphs such that $n_j \geq 2$ for each $j \in [m]$ and $\cup_{j=1}^{m} R_j = P \cup Q$;*
- *(c) For every nonempty $B \subseteq [m]$, $\|\{l \in [k] \mid (\cup_{j \in B} R_j \cap Q_l) \neq \emptyset\}\| > \|B\|$;*
- *(d) $\hat{C}(G) \cdot \sum_{p \in P} \mathbf{f}_p$ has a perfect internal matching.*

The decomposition $\hat{C}(G) \cdot \hat{A}(G) \cdot \hat{D}(G)$ is unique up to an allowable relabeling of its components.

Proof. We derive the desired decomposition of G from the Gallai-Edmonds decomposition of $V(G)$ by Theorem 2.1. The graphs $\hat{D}(G)$, $\hat{A}(G)$ and $\hat{C}(G)$ are obtained essentially by cutting G at those internal edges that join the subgraphs induced by $D(G)$, $A(G)$ and $C(G)$ to each other. In addition, we also cut G at every edge connecting two (not necessarily distinct) vertices of $A(G)$. For each such edge, the created two external edges will then be remerged by a constant 1_C, $C \subseteq P$ when the composition $\hat{C}(G) \cdot \hat{A}(G)$ is performed. The constants 1_C must be placed as extra components in $\hat{C}(G)$.

Since the edges spanned by $A(G)$, as well as those joining $A(G)$ to $C(G)$, are forbidden in G, the composite $\hat{C}(G) \cdot \sum_{p \in P} \mathbf{f}_p$ will have a perfect internal matching. Statement (c) is equivalent to (iii) of Theorem 2.1. To prove the uniqueness of the decomposition, one must observe that if $G = \hat{C}(G) \cdot \hat{A}(G) \cdot \hat{D}(G)$ with $\hat{C}(G)$, $\hat{A}(G)$ and $\hat{D}(G)$ satisfying (a)–(d) in the assertion of the theorem, then the internal vertices of $\hat{D}(G)$ are the only ones that can be left uncovered by any maximum internal matching of G. Thus, the internal vertices of $\hat{D}(G)$ are exactly the vertices contained in $D(G)$. The rest of the proof is obvious. □

The graph $\hat{C}(G) \cdot \sum_{p \in P} \mathbf{f}_p$ occurring in (d) of Theorem 4.2 will play an important role in the sequel, therefore we introduce the notation $h(G)$ for this graph. Clearly, $h(G)$ does not depend on the choice of the labeling set P and its sort is the same as that of G. Note that $h(G) = G$ iff G has a perfect internal matching.

Now we turn to defining the U-sorted algebra $\mathcal{S}$, which we call the *Gallai-Edmonds algebra* of graphs. The underlying set of $\mathcal{S}$ corresponding to sort $A \in U$ is the set of graphs $G : A$ having a perfect internal matching. The algebra $\mathcal{S}$ has the same operations and constants as $\mathcal{G}$ with the following interpretation.

(i) For each constant symbol n_A, $(n_A)_{\mathcal{S}} = (n_A)_{\mathcal{G}}$.

Note that the number of perfect internal matchings of n_A is n if $n \geq 2$ and 2 if $n = 1$.

(ii) If $G : A$ and $H : B$ are graphs having a perfect internal matching, then

$$(G \cdot H)_{\mathcal{S}} = h((G \cdot H)_{\mathcal{G}}). \tag{3}$$

For simplicity, we shall omit the subscript $\mathcal{G}$ referring to interpretation by the algebra $\mathcal{G}$ in graph expressions. In other words, $\mathcal{G}$ is the "default" interpretation. It is easy to derive from (3) that for graphs G and H of appropriate sorts,

$$(G + H)_{\mathcal{S}} = h(G) + h(H) \tag{4}$$

and

$$(\updownarrow_B G)_{\mathcal{S}} = h(\updownarrow_B G). \tag{5}$$

Our goal is to prove that $h : \mathcal{G} \to \mathcal{S}$ is a homomorphism. To this end we need the following theorem.

Theorem 4.3 *If $G : A \cup B$ is a graph for some $A \in U$ and two-element set $B \in U$ disjoint from A, then*

$$h(\updownarrow_B G) = h(\updownarrow_B h(G)).$$

Proof. Assume that $h(G) = \hat{C}(G) \cdot \sum_{p \in P} \mathbf{f}_p$ as described in Theorem 4.2. For each $p \in P$, let $v_0(p)$ denote the vertex of $h(G)$ that supports the loop of $\mathbf{f}_p$. Furthermore, let $v_1(p) \neq v_0(p)$ be the vertex adjacent to $v_0(p)$ in $h(G)$, and let $v_A(p)$ and $v_C(p)$ be those vertices in $A(G)$ and $C(G)$, respectively, that are adjacent to the external vertex p in $\hat{A}(G)$ and $\hat{C}(G)$, see Fig. 6. In Fig. 6, dotted lines indicate those edges of $h(G)$ that are not present in the original graph G. The dashed edge e on the top of the figure comes

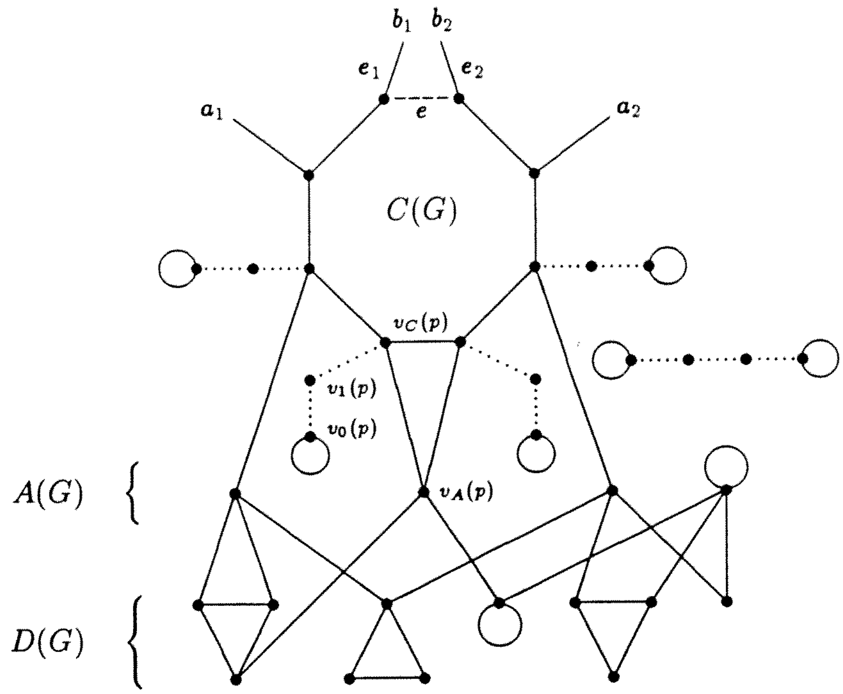

Figure 6: An illustration of Theorem 4.3

from merging the edges e_1 and e_2 incident with the two external vertices b_1 and b_2 in B. Let $G_0 = \updownarrow_B h(G)$. We have to prove that $h(\updownarrow_B G) = h(G_0)$. Two cases are possible.

Case 1. G_0 still has a perfect internal matching.

In this case it is clear that $D(\updownarrow_B G) = D(G)$, hence by Theorem 4.2,

$$\hat{C}(\updownarrow_B G) = \updownarrow_B \hat{C}(G).$$ [1]

Consequently,

$$\begin{aligned} h(\updownarrow_B G) &= \hat{C}(\updownarrow_B G) \cdot \sum_{p \in P} \mathbf{f}_p = (\updownarrow_B \hat{C}(G)) \cdot \sum_{p \in P} \mathbf{f}_p \\ &= \updownarrow_B (\hat{C}(G) \cdot \sum_{p \in P} \mathbf{f}_p) = \updownarrow_B h(G) \\ &= G_0 = h(G_0). \end{aligned}$$

[1]**Strictly speaking, the graphs on the two sides of this equation differ only in the labeling of those external vertices that do not belong to A.**

Note that $\eta(G) = \eta(\updownarrow_B G)$, i.e. the internal deficiency does not increase switching from G to $\updownarrow_B G$.

Case 2. G_0 does not have a perfect internal matching.

Now $\eta(\updownarrow_B G) = \eta(G) - 1$. For, let M be any maximum internal matching of G. Then one of the edges e_1, e_2 must be in M while the other must stay out, otherwise G_0 would have a perfect internal matching. Say $e_1 \in M$. Leaving out e_1 from M, we obtain a matching of $\updownarrow_B G$ covering $\eta(G) - 1$ internal vertices. On the other hand, it is impossible for any matching M of $\updownarrow_B G$ to cover $\eta(G)$ or more internal vertices because in this case the successive opening and reclosing of the "gate" e in $\updownarrow_B G$ would produce a perfect internal matching in G_0 by (iv) of Theorem 2.1. For the same reason, the internal deficiency of G_0 is 1.

Consider the Gallai-Edmonds decomposition of $V(G_0)$ concentrating on the vertices

$$\{v_0(p), v_1(p), v_A(p), v_C(p) \mid p \in P\}.$$

Our aim is to show that every maximum internal matching of $\updownarrow_B G$ is derivable from a suitable maximum internal matching of G_0 in one of the two ways described below. Let M_0 be a maximum internal matching of G_0.

Pattern 1. If M_0 covers all of $\{v_0(p) \mid p \in P\}$, then we can associate with M_0 a group of maximum internal matchings of $\updownarrow_B G$ by leaving out the edges $\{(v_0(p), v_1(p)) \mid p \in P\}$, adding a near-perfect matching of each component of $D(G)$ and by matching all vertices of $A(G)$ with vertices in distinct components of $D(G)$ as described in (iv) of Theorem 2.1. See again Fig. 6.

Pattern 2. If M_0 does not cover $v_0(p)$ for some $p \in P$, then p is unique and M_0 covers all the internal vertices of $\updownarrow_B G$ not contained in $A(G) \cup D(G)$. (Remember that the internal deficiency of G_0 is 1.) In this case, too, it is possible to extend M_0 to a group of maximum internal matchings of $\updownarrow_B G$ through the following steps: (1) leave out the edges $\{(v_0(q), v_1(q)) \mid q \in P - p\}$, (2) add a near-perfect matching of each component of $D(G)$, (3) match all vertices of $A(G) - v_A(p)$ with distinct components of $D(G)$ and (4) substitute the edge $(v_1(p), v_C(p))$ by the edge $(v_C(p), v_A(p))$. Step (3) is enabeled by (iii) of Theorem 2.1. Note that the edge $(v_1(p), v_C(p))$ is in M_0, since the only internal vertex of G_0 not covered by M_0 is $v_0(p)$.

We claim that all maximum internal matchings of $\updownarrow_B G$ can be obtained as an application of one of the above two extension patterns. For, let M_1 be a maximum internal matching of $\updownarrow_B G$. Clearly, every vertex of $A(G)$ must be covered by M_1, and there exists at most one exceptional vertex in $A(G)$ which is matched with a vertex not in $D(G)$. Indeed, the assumption that there are two such vertices would imply by (iii) of Theorem 2.1 that $\eta(\updownarrow_B G) \leq \eta(G) - 2$, which is a contradiction. Now, if there is no exceptional vertex in $A(G)$, then the first pattern applies, otherwise the second.

We have thus proved that an internal vertex v of $\updownarrow_B G$ can be left uncovered by a maximum internal matching of this graph iff $v \in D(G)$ or $v \in D(G_0)$. In other words, since $v_1(p) \notin D(G_0)$ for any $p \in P$,

$$D(\updownarrow_B G) = D(G) \cup D(G_0) - \{v_0(p) \mid p \in P\}.$$

It follows that

$$\begin{aligned} A(\updownarrow_B G) &= A(G) \cup A(G_0) - \{v_1(p) \mid p \in P\}, \\ C(\updownarrow_B G) &= C(G_0) - \{v_0(p), v_1(p) \mid p \in P\}. \end{aligned} \tag{6}$$

Let $G_0 = \hat{C}(G_0) \cdot \hat{A}(G_0) \cdot \hat{D}(G_0)$ be the decomposition of G_0 according to Theorem 4.2 such that $\hat{C}(G_0) : A \cup P_0$. If $P' = \{p \in P \mid v_0(p) \notin D(G_0)\}$, then $\hat{C}(G_0)$ can be written in the form $\tilde{C} \cdot \sum_{p \in P'} \mathbf{f}_p$ for some graph $\tilde{C} : A \cup P_0 \cup P'$. Equations (6) imply that, up to a relabeling of the external vertices $P_0 \cup P'$ in $\tilde{C}$, $\tilde{C} = \hat{C}(\updownarrow_B G)$. Thus,

$$\begin{aligned} h(G_0) &= \hat{C}(G_0) \cdot \sum_{p \in P_0} \mathbf{f}_p = \tilde{C} \cdot \sum_{p \in P_0 \cup P'} \mathbf{f}_p \\ &= \hat{C}(\updownarrow_B G) \cdot \sum_{p \in P_0 \cup P'} \mathbf{f}_p \\ &= h(\updownarrow_B G). \quad \square \end{aligned}$$

Corollary 4.4 *The mapping h is a homomorphism of $\mathcal{G}$ onto $\mathcal{S}$.*

Proof. It is sufficient to prove that h preserves the operations sum and merge. For sum, this statement is equivalent to equation (4) above. Concerning merge, let $G : A \cup B$ be a graph for some $A \in U$ and two-element set B disjoint from A. Then by Theorem 4.3 and by equation (5),

$$h(\updownarrow_B G) = h(\updownarrow_B h(G)) = (\updownarrow_B h(G))_{\mathcal{S}}. \quad \square$$

5. Conclusion

We have introduced a many-sorted algebraic structure $\mathcal{G}$ on graphs with labeled external vertices. Rephrasing the Gallai-Edmonds Structure Theorem in terms of the algebra $\mathcal{G}$, we have seen that the operation composition reflects the decomposition feature of that theorem in a natural way. As the main result of the paper we have shown that graphs having a perfect internal matching can also be equipped with the operations and constants of $\mathcal{G}$ so that the resulting algebra $\mathcal{S}$ becomes a homomorphic image of $\mathcal{G}$. The homomorphism $h : \mathcal{G} \to \mathcal{S}$ is again characteristic of the Gallai-Edmonds Theorem.

We believe that the Gallai-Edmonds algebra $\mathcal{S}$ is a basic tool in the study of soliton automata described in the Introduction. Not only does it specify perfect soliton graphs as syntactical objects representing soliton automata, but it also provides an operation to build up any perfect soliton graph from the constant star graphs. The right choice of the operation composition in the algebras $\mathcal{G}$ and $\mathcal{S}$ is crucial. Were we not able to establish h as a homomorphism, the algebras $\mathcal{G}$ and $\mathcal{S}$ would be of little interest from the point of view of perfect soliton graphs. Moreover, to really justify our approach we still have to prove that soliton automata, i.e. the semantics of soliton graphs, can also be given the structure of an algebra which is a homomorphic image of $\mathcal{S}$. This rather complex issue will be dealt with in a forthcoming paper.

Another interesting problem is the axiomatization of the algebras describing the syntax and semantics of soliton automata. Here we would only like to provide a short overview of this problem without going into formal proofs. The reader is referred to [12] for the algebraic terminology used.

The identities A1–A3 below axiomatize the algebra $\mathcal{G}$.

A1: $G \cdot H = H \cdot G$ for all graphs $G : A$, $H : B$;

A2: $G \cdot (H \cdot L) = (G \cdot H) \cdot L$ for all graphs $G : A$, $H : B$ and $L : C$ such that $A \cap B \cap C = \emptyset$;

A3: $n_{A \cup a} \cdot 1_{\{a,b\}} = n_{A \cup b}$ for every constant $n_{A \cup a}$ such that $a \neq b$ and $\{a, b\} \cap A = \emptyset$.

For the moment we do not know if $\mathcal{S}$ can be axiomatized in the same equational way, in other words, if the homomorphism h is fully invariant or not. Concerning semantics, however, there are a couple of identities that are easy to observe. These identities are the following.

B1: $2_{\{1,2\}} \cdot 2_{\{2,3\}} = 1_{\{1,3\}}$;

B2: $3_{\{1,2,3\}} \cdot \mathbf{f}_3 = 2_{\{1,2\}}$;

B3: $3_{\{1,2,3\}} \cdot \mathrm{m}_3 = \mathbf{f}_1 + \mathbf{f}_2$;

B4: $G_1 = G_2$, where G_1 and G_2 are the two graphs of sort $\{1, 2, 3, 4\}$ in Fig. 1.

Perhaps the most interesting identity is B4 above. The hidden meaning of B4 is that the graphs G_1 and G_2 are both equivalent to the 4-star $4_{\{1,2,3,4\}}$. For soliton automata, this fact interprets as follows. There exist isomorphisms ϕ_i, $i = 1, 2$ mapping the set of all perfect internal matchings (states) of $4_{\{1,2,3,4\}}$ onto the set of states of G_i. The isomorphisms ϕ_i, $i = 1, 2$ have the property that if there is an alternating path w connecting two external vertices in $4_{\{1,2,3,4\}}$ with respect to state M, then there is a unique alternating path $\phi_i(w)$ connecting the same external vertices in G_i with respect to $\phi_i(M)$. Moreover, the state transition induced by w in state M of $4_{\{1,2,3,4\}}$ is the same as that induced by $\phi_i(w)$ in state $\phi_i(M)$ of G_i.

Applying the above heuristic argument iteratively, we find that every n-star has a set of equivalent alternative graphs, each alternative consisting only of vertices of degree at most 3. Furthermore, by the help of axiom B1 it becomes possible to eliminate multiple and loop edges by subdividing them twice. Conversely, we can as well introduce such edges if they offer more convenience for syntactical considerations.

Regarding soliton automata, axioms B1–B4 mean that not only is the semantics driven by the syntax, but semantics, too, has an impact back on syntax. Really, in the light of the previous paragraph, if the identities B1–B4 are in effect, then we can generalize the syntax of soliton automata from soliton graphs to arbitrary graphs without having to worry about generalizing the semantics at the same time. With this assumption the syntax becomes much more flexible to handle.

References

[1] M. Bartha, E. Gombás, A structure theorem for maximum internal matchings in graphs, Research Report #9102, Department of Computer Science, Memorial University of Newfoundland, St. John's, Canada, 1991.

[2] M. Bartha, H. Jürgensen, Characterizing finite undirected multigraphs as indexed algebras, Research Report #252, Department of Computer Science, The University of Western Ontario, London, Canada, 1989.

[3] M. Bartha, A finite axiomatization of flowchart schemes, *Acta Cybernet.* **2** (1987), 203–217.

[4] M. Bauderon, B. Courcelle, Graph expressions and graph rewritings, *Math. Systems Theory* **20** (1987), 83–127.

[5] S. L. Bloom and Z. Ésik, Axiomatizing schemes and their behaviors, *J. Comput. System Sci.* **31** (1985), 375–393.

[6] F. L. Carter, Comformational switching at the molecular level, in *Molecular Electronic Devices* (F. L. Carter ed.), Marcel Dekker, Inc., New York, 1982, pp. 51–72.

[7] J. Dassow, H. Jürgensen, Soliton automata, *J. Comput. System Sci.* **40** (1990) 154–181.

[8] J. Edmonds, Paths, trees, and flowers, *Canad. J. Math.* **17** (1965), 449–467.

[9] C. C. Elgot, Monadic computations and iterative algebraic theories, *Proceedings, Logic Colloquium'73*, (H. E. Rose and J. C. Shepherdson, eds.), Studies in Logic and the Foundations of Mathematics, North Holland, Amsterdam, 1975, pp. 175–230.

[10] T. Gallai, Kritische Graphen II, *Magyar Tud. Akad. Mat. Kutató Int. Közl.* **8** (1963), 373–395.

[11] T. Gallai, Maximale Systeme unabhängiger Kanten, *Magyar Tud. Akad. Mat. Kutató Int. Közl.* **9** (1964), 401–413.

[12] G. Grätzer, *Universal Algebra*, Springer Verlag, Berlin, 1968, 1979.

[13] L. Lovász, M. D. Plummer, *Matching Theory*, North Holland, Amsterdam, 1986.