Discrete Applied Mathematics 259 (2019) 31–40

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Decompositions of graphs based on a new graph product

Antoon H. Boode, Hajo Broersma *

Faculty of EEMCS, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 22 November 2017
Received in revised form 3 December 2018
Accepted 14 December 2018
Available online 21 January 2019

Keywords:
Vertex-removing synchronised product
Product graph
Graph decomposition
Synchronising processes

ABSTRACT

Recently, we have introduced a new graph product, motivated by applications in the context of synchronising periodic real-time processes. This vertex-removing synchronised product (VRSP) is based on modifications of the well-known Cartesian product, and closely related to the synchronised product due to Wöhrle and Thomas. Here, we recall the definition of the VRSP and use it to define two different decompositions of graphs. Although our main results apply to directed labelled acyclic multigraphs, the VRSP can also be used to decompose any undirected graph of order at least 4 into two smaller graphs.

1. Introduction

Recently, we have introduced a new graph product, motivated by applications in the context of synchronising periodic real-time processes, in particular in the field of robotics. More on the background, definitions and applications can be found in two conference contributions [4,5] and a journal paper [6]. We repeat some of the background and definitions here for convenience, and for supplying the motivation for the research that led to the decomposition theorems that we state and prove in Sections 5 and 6.

In [5], we have modelled periodic real-time processes as directed acyclic labelled multigraphs. These graphs are closely related to state transition systems [1]. The vertices of such a graph represent the states of a periodic real-time process, while the labelled arcs represent actions, i.e., transitions from one state to another. The label (in fact, a label pair) on an arc represents the name or type of the action together with the worst case duration of its execution. We give the formal definitions of these graphs in Section 2.

Embedded control systems play a crucial role in many application areas. In particular, in the field of robotics, it is obvious that these systems (embedded in robots) are key to the functionality and operational behaviour of robots. The software of such control systems is usually designed using a general purpose computing system (not in the robot). These general purpose computers generally have more processing power and memory available than the embedded control system. The embedded control system is the target system on which the software will run eventually, after it has been designed and validated. The hardware of the target system is usually much more limited with respect to available memory and processing power. If the processes that have to run on the target system are periodic and real-time, they have deadlines to fulfil the timing requirements, and they require memory for storing the data and software.

Periodic real-time (robotic) applications can be designed using process algebras like, for example, a calculus of communicating systems [10], communicating sequential processes [8], and finite state processes [9].

During the design phase, the designer distributes the required behaviour over sometimes more than a hundred processes. These processes very often synchronise over actions, e.g., to assert whether a subset of the processes will be ready to start

* Corresponding author.
E-mail address: h.j.broersma@utwente.nl (H. Broersma).

https://doi.org/10.1016/j.dam.2018.12.014

executing at the same time. Due to this synchronisation such applications usually suffer from a considerable overhead related to so-called context switches.

In [5], the vertex-removing synchronised product (VRSP) has been introduced as a means to reduce the number of context switches. This VRSP is a modification of the well-known Cartesian product of graphs. It is based on the synchronised product due to Wöhrle and Thomas [11], which is used in model-checking synchronised products of transition systems.

The VRSP reduces the number of context switches and in many cases realises a performance gain for periodic real-time applications. This is achieved by (repetitively) combining two graphs representing two processes that synchronise over some action. The combined graph of two graphs then represents a process that will have only one context switch per synchronising action, whereas the two processes separately would each have one context switch per synchronising action [5].

Using the VRSP, the set of graphs representing a set of different processes can, under certain conditions, be transformed into a new set of graphs. This can be particularly useful if the original set of graphs represents a set of processes that cannot meet their deadline or do not fit into the available memory. The aim is that for such a new set of graphs, the processes that they represent meet their deadline and fit into the available memory. In the worst case, it can happen that there is no set of processes with respect to the original set of processes that will do so. In that case, the VRSP cannot result in a suitable solution when applied in any way to the graphs representing the original set of processes.

One way out, for which we introduce and develop the tools here, is to use the VRSP in order to enable new combinations of subprocesses of the original set of processes, without changing the functionality and behaviour of the total set of new (sub)processes. We accomplish this by decomposing a graph G (representing one of the processes) into two smaller graphs H_1 and H_2 such that the VRSP of H_1 and H_2 is isomorphic to G. It should be noted here, that the graphs H_1 and H_2 are not subgraphs of G, but that they are obtained from G by applying a contraction operation, to be specified later.

The rest of the paper is organised as follows. In the next sections, we first recall the formal graph definitions (in Section 2) as well as the definition of the VRSP (in Section 3), together with other relevant terminology and notation. We also extend the notions of graph isomorphism and contraction to labelled acyclic directed multigraphs (in Section 4). In Sections 5 and 6, we use the VRSP to state and prove two results enabling different decompositions of graphs. Although our main results apply to directed labelled acyclic multigraphs, we will show in Section 7 that the VRSP can also be used to decompose any undirected graph of order at least 4 into two smaller graphs.

2. Terminology and notation

We use the textbook of Bondy and Murty [2] for terminology and notation we do not specify here. Throughout, unless we specify explicitly that we consider other types of graphs, all graphs we consider are *labelled acyclic directed multigraphs*, i.e., they may have multiple arcs. Such graphs consist of a *vertex set* V (representing the states of a process), an *arc set* A (representing the actions, i.e., transitions from one state to another), a set of *labels* L (in our applications in fact a set of label pairs, each representing a type of action and the worst case duration of its execution), and two mappings. The first mapping $\mu : A \to V \times V$ is an incidence function that identifies the *tail* and *head* of each arc $a \in A$. In particular, $\mu(a) = (u, v)$ means that the arc a is directed from $u \in V$ to $v \in V$, where $tail(a) = u$ and $head(a) = v$. We also call u and v the *ends* of a. Sometimes we slightly abuse the notation by writing that there exists an arc with $\mu = (u, v)$ without specifying the arc itself (because in these cases the name of the arc is irrelevant). The second mapping $\lambda : A \to L$ assigns a label pair $\lambda(a) = (\ell(a), t(a))$ to each arc $a \in A$, where $\ell(a)$ is a string representing the (name of an) action and $t(a)$ is the *weight* of the arc a. This weight $t(a)$ is a real positive number representing the worst case execution time of the action represented by $\ell(a)$.

Let G denote a graph according to the above definition. An arc $a \in A(G)$ is called an *in-arc* of $v \in V(G)$ if $head(a) = v$, and an *out-arc* of v if $tail(a) = v$. The *in-degree* of v, denoted by $d^-(v)$, is the number of in-arcs of v in G; the *out-degree* of v, denoted by $d^+(v)$, is the number of out-arcs of v in G. The subset of $V(G)$ consisting of vertices v with $d^-(v) = 0$ is called the *source* of G, and is denoted by $S'(G)$. The subset of $V(G)$ consisting of vertices v with $d^+(v) = 0$ is called the *sink* of G, and is denoted by $S''(G)$.

For disjoint nonempty sets $X, Y \subseteq V(G)$, $[X, Y]$ denotes the set of arcs of G with one end in X and one end in Y. If the head of the arc $a \in [X, Y]$ is in Y, we call a a *forward arc* (of $[X, Y]$); otherwise, we call it a *backward arc*.

The acyclicity of G implies a natural ordering of the vertices into disjoint sets, as follows. We define $S^0(G)$ to denote the set of vertices with in-degree 0 in G (so $S^0(G) = S'(G)$), $S^1(G)$ the set of vertices with in-degree 0 in the graph obtained from G by deleting the vertices of $S^0(G)$ and all arcs with tails in $S^0(G)$, and so on, until the final set $S^t(G)$ contains the remaining vertices with in-degree 0 and out-degree 0 in the remaining graph. Note that these sets are well-defined since G is acyclic, and also note that $S^t(G) \neq S''(G)$, in general. If a vertex $v \in V(G)$ is in the set $S^j(G)$ in the above ordering, we say that v is *at level* j in G. This ordering implies that each arc $a \in A(G)$ can only have $tail(a) \in S^{j_1}(G)$ and $head(a) \in S^{j_2}(G)$ if $j_1 < j_2$.

A graph G is called *weakly connected* if all pairs of distinct vertices u and v of G are connected through a sequence of distinct vertices $u = v_0 v_1 \dots v_k = v$ and arcs $a_1 a_2 \dots a_k$ of G with $\mu(a_i) = (v_{i-1}, v_i)$ or (v_i, v_{i-1}) for $i = 1, 2, \dots, k$. We are mainly interested in weakly connected graphs, or in the weakly connected components of a graph G. If $X \subseteq V(G)$, then the *subgraph of G induced by X*, denoted as $G[X]$, is the graph on vertex set X containing all the arcs of G which have both their ends in X (together with L, μ and λ restricted to this subset of the arcs). If $X \subseteq V$ induces a weakly connected subgraph of G, but there is no set $Y \subseteq V$ such that $G[Y]$ is weakly connected and X is a proper subset of Y, then $G[X]$ is called a *weakly connected component* of G. In the sequel, throughout we omit the words weakly connected, so a component should always be understood as a weakly connected component. In contrast to the notation in the textbook of Bondy and Murty [2], we use $\omega(G)$ to denote the number of components of a graph G.

We denote the components of G by G_i, where i ranges from 1 to $\omega(G)$ of G. In that case, we use V_i, A_i and L_i as shorthand notation for $V(G_i)$, $A(G_i)$ and $L(G_i)$, respectively. The mappings μ and λ have natural counterparts restricted to the subsets $A_i \subset A(G)$ that we do not specify explicitly. We use $G = \sum_{i=1}^{\omega(G)} G_i$ to indicate that G is the disjoint union of its components, implicitly defining its components as G_1 up to $G_{\omega(G)}$. In particular, $G = G_1$ if and only if G is weakly connected itself.

An arc a with label pair $\lambda(a)$ in a graph G is a *synchronising arc* with respect to another graph H, if and only if there exists an arc $b \in A(H)$ with label pair $\lambda(b)$ such that $\lambda(a) = \lambda(b)$.

In the next section, we recall some of the definitions that appeared in a mathematically less rigorous form in [5]. There we introduced a (directed labelled multigraph) analogue of the Cartesian product of two graphs, and several other new products we derived from it, resulting in what we refer to as the vertex-removing synchronised product (VRSP). For background and results on many other different types of graph products, we refer the interested reader to the book of Hammack et al. [7].

3. Graph products

Instead of defining products for general pairs of graphs, for notational reasons we find it convenient to define those products for two components G_i and G_j of a disconnected graph G. We start by introducing the next analogue of the Cartesian product.

The *Cartesian product* $G_i \Box G_j$ of G_i and G_j is defined as the graph on vertex set $V_{i,j} = V_i \times V_j$, and arc set $A_{i,j}$ consisting of two types of labelled arcs. For each arc $a \in A_i$ with $\mu(a) = (v_i, w_i)$, an *arc of type i* is introduced between tail $(v_i, v_j) \in V_{i,j}$ and head $(w_i, w_j) \in V_{i,j}$ whenever $v_j = w_j$; such an arc receives the label $\lambda(a)$. This implicitly defines parts of the mappings μ and λ for $G_i \Box G_j$. Similarly, for each arc $a \in A_j$ with $\mu(a) = (v_j, w_j)$, an *arc of type j* is introduced between tail $(v_i, v_j) \in V_{i,j}$ and head $(w_i, w_j) \in V_{i,j}$ whenever $v_i = w_i$; such an arc receives the label $\lambda(a)$. This completes the definition of $A_{i,j}$ and the mappings μ and λ for $G_i \Box G_j$. So, arcs of type i and j correspond to arcs of G_i and G_j, respectively, and have the associated labels. For $k \geq 3$, the Cartesian product $G_1 \Box G_2 \Box \cdots \Box G_k$ is defined recursively as $((G_1 \Box G_2) \Box \cdots) \Box G_k$. This Cartesian product is commutative and associative, as can be verified easily and is a well-known fact for the undirected analogue.

Since we are particularly interested in synchronising arcs, we modify the Cartesian product $G_i \Box G_j$ according to the existence of synchronising arcs, i.e., pairs of arcs with the same label pair, with one arc in G_i and one arc in G_j.

The first step in this modification consists of ignoring (in fact deleting) the synchronising arcs while forming arcs in the product, but additionally combining pairs of synchronising arcs of G_i and G_j into one arc, yielding the *intermediate product* which we denote by $G_i \boxtimes G_j$.

To be more precise, $G_i \boxtimes G_j$ is obtained from $G_i \Box G_j$ by first ignoring all except for the so-called *asynchronous* arcs, i.e., by only maintaining all arcs $a \in A_{i,j}$ for which $\mu(a) = ((v_i, v_j), (w_i, w_j))$, whenever $v_j = w_j$ and $\lambda(a) \notin L_j$, as well as all arcs $a \in A_{i,j}$ for which $\mu(a) = ((v_i, v_j), (w_i, w_j))$, whenever $v_i = w_i$ and $\lambda(a) \notin L_i$. Additionally, we add arcs that replace synchronising pairs $a_i \in A_i$ and $a_j \in A_j$ with $\lambda(a_i) = \lambda(a_j)$. If $\mu(a_i) = (v_i, w_i)$ and $\mu(a_j) = (v_j, w_j)$, such a pair is replaced by an arc $a_{i,j}$ with $\mu(a_{i,j}) = ((v_i, v_j), (w_i, w_j))$ and $\lambda(a_{i,j}) = \lambda(a_i)$. We call such arcs of $G_i \boxtimes G_j$ *synchronous* arcs.

The second step in this modification consists of removing (from $G_i \boxtimes G_j$) the vertices $(v_i, v_j) \in V_{i,j}$ and the arcs a with $tail(a) = (v_i, v_j)$, in the case that (v_i, v_j) has *level* > 0 in $G_i \Box G_j$ but *level* 0 in $G_i \boxtimes G_j$. This is then repeated in the newly obtained graph, and so on, until there are no more vertices at *level* 0 in the current graph that are at *level* > 0 in $G_i \Box G_j$. This finds its motivation in the fact that in our applications, the states that are represented by such vertices can never be reached, so are irrelevant.

The resulting graph is called the *vertex-removing synchronised product* (VRSP for short) of G_i and G_j, and denoted as $G_i \boxslash G_j$. For $k \geq 3$, the VRSP $G_1 \boxslash G_2 \boxslash \cdots \boxslash G_k$ is defined recursively as $((G_1 \boxslash G_2) \boxslash \cdots) \boxslash G_k$. The VRSP is commutative, but not associative in general, in contrast to the Cartesian product. These properties are not relevant for the decomposition results that follow. However, for these results it is relevant to introduce counterparts of graph isomorphism and graph contraction that apply to our types of graphs. We define these counterparts in the next section.

4. Graph isomorphism and graph contraction

The isomorphism we introduce in this section is an analogue of a known concept for unlabelled graphs, but involves statements on the labels.

We assume that two different arcs with the same head and tail have different labels; otherwise, we replace such multiple arcs by one arc with that label, because these arcs represent exactly the same action at the same stage of a process.

Formally, an *isomorphism* from a graph G to a graph H consists of two bijections $\phi : V(G) \to V(H)$ and $\rho : A(G) \to A(H)$ such that for all $a \in A(G)$, one has $\mu(a) = (u, v)$ if and only if $\mu(\rho(a)) = (\phi(u), \phi(v))$ and $\lambda(a) = \lambda(\rho(a))$. Since we assume that two different arcs with the same head and tail have different labels, however, the bijection ρ is superfluous. The reason is that, if (ϕ, ρ) is an isomorphism, then ρ is completely determined by ϕ and the labels. In fact, if (ϕ, ρ) is an isomorphism and $\mu(a) = (u, v)$ for an arc $a \in A(G)$, then $\rho(a)$ is the unique arc $b \in A(H)$ with $\mu(b) = (\phi(u), \phi(v))$ and label $\lambda(b) = \lambda(a)$. Thus, we may define an isomorphism from G to H as a bijection $\phi : V(G) \to V(H)$ such that there exists an arc $a \in A(G)$ with $\mu(a) = (u, v)$ if and only if there exists an arc $b \in A(H)$ with $\mu(b) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$.

Next, we define what we mean by contraction. Let X be a nonempty proper subset of $V(G)$, and let $Y = V(G) \setminus X$. By *contracting* X we mean replacing X by a new vertex $\tilde{x}$, deleting all arcs with both ends in X, replacing each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X$ and $v \in Y$ by an arc c with $\mu(c) = (\tilde{x}, v)$ and $\lambda(c) = \lambda(a)$, and replacing each arc $b \in A(G)$ with $\mu(b) = (u, v)$ for $u \in Y$ and $v \in X$ by an arc d with $\mu(d) = (u, \tilde{x})$ and $\lambda(d) = \lambda(b)$. We denote the resulting graph as G/X, and say that G/X is the *contraction of G with respect to X*.

Now, we have all the necessary ingredients for stating and proving our first decomposition result.

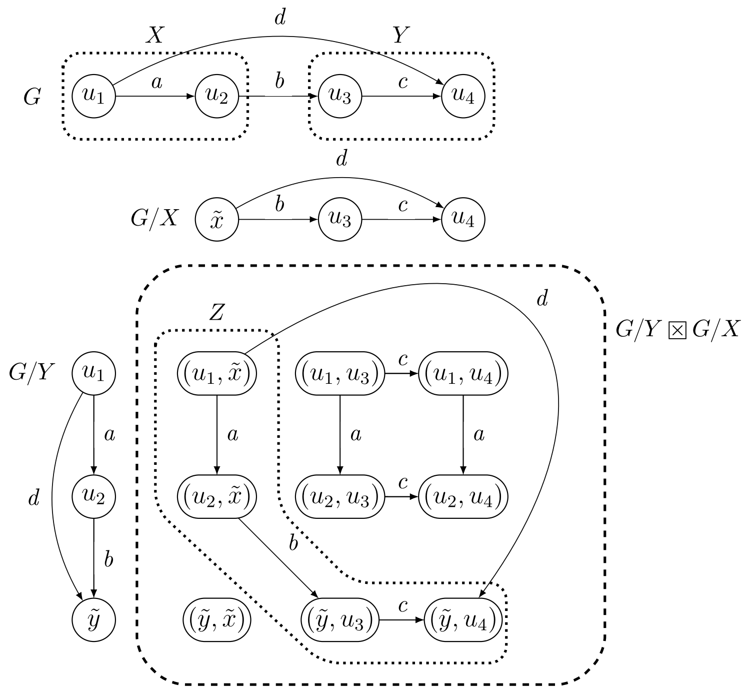

Fig. 1. Decomposition of $G \cong G/Y \boxbslash G/X$. The set Z from the proof of Theorem 1 and the graph isomorphic to G induced by Z in $G/Y \boxtimes G/X$ are indicated within the dotted region (except for the arc with label d).

5. A graph decomposition based on the VRSP

We start this section by presenting and proving our first decomposition theorem, of which an illustrative small example is given in Fig. 1. We assume that the graphs we want to decompose are connected; if not, we can apply our decomposition result to the components separately. Let us start by explaining the example of Fig. 1 first.

At the top of Fig. 1 we depicted a small graph G, together with a partition of its vertex set into two nonempty sets X and Y, such that $[X, Y]$ contains forward arcs only. Note that, in the figure we indicated labels (label pairs) $\lambda(a)$, etc. just by a, etc. The rest of the figure shows the graph G/Y on the left, the graph G/X at the top, and the graph $G/Y \boxtimes G/X$; this graph is the result of maintaining the asynchronous arcs and replacing the synchronising arcs by synchronous arcs in the Cartesian product $G/Y \square G/X$. The set of vertices that remain after the second step in the modification are indicated by Z (the vertices in the dotted region). In this example, it is clear that Z induces a graph isomorphic to G. So, in this example $G \cong G/Y \boxbslash G/X$. Our first decomposition theorem states sufficient conditions for this conclusion to hold in general. We will show that none of these conditions can be omitted without violating the conclusion.

Theorem 1. *Let G be a graph, let X be a nonempty proper subset of $V(G)$, and let $Y = V(G) \setminus X$. Suppose that all the arcs of $[X, Y]$ have distinct labels and that the arcs of G/X and G/Y corresponding to the arcs of $[X, Y]$ are the only synchronising arcs of G/X and G/Y. If $S'(G) \subseteq X$ and $[X, Y]$ has no backward arcs, then $G \cong G/Y \boxbslash G/X$.*

Proof. It clearly suffices to define a mapping $\phi : V(G) \to V(G/Y \boxbslash G/X)$ and to prove that ϕ is an isomorphism from G to $G/Y \boxbslash G/X$.

Let $\tilde{x}$ and $\tilde{y}$ be the new vertices replacing the sets X and Y when defining G/X and G/Y, respectively. Consider the mapping $\phi : V(G) \to V(G/Y \boxbslash G/X)$ defined by

$\phi(v) = (v, \tilde{x})$ for all $v \in X$ and $\phi(w) = (\tilde{y}, w)$ for all $w \in Y$.

Then ϕ is obviously a bijection if $V(G/Y \boxbslash G/X) = Z$, where Z is defined as $Z = \{(v, \tilde{x}) \mid v \in X\} \cup \{(\tilde{y}, w) \mid w \in Y\}$. We are going to show this later by arguing that all the other vertices of $G/Y \square G/X$ will disappear from $G/Y \boxtimes G/X$. But first we are going to prove the following claim.

Claim 1. *The subgraph of $G/Y \boxtimes G/X$ induced by Z is isomorphic to G.*

Proof. Obviously, ϕ is a bijection from $V(G)$ to Z. It remains to show that this bijection preserves the arcs and their labels. By the definition of the Cartesian product, for each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X$ and $v \in X$, there exists an arc b in

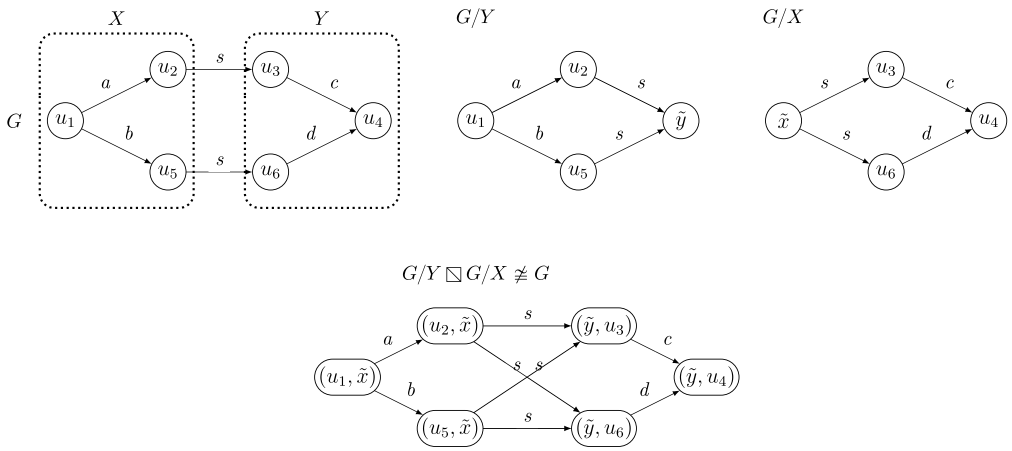

Fig. 2. Failing decomposition of G into G/Y and G/X, when there exist two different arcs with identical labels in $[X, Y]$.

$G/Y \boxtimes G/X$ with $\mu(b) = ((u, \tilde{x}), (v, \tilde{x})) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. This is because the arc $a \notin [X, Y]$, and hence a is not a synchronising arc of G/Y with respect to G/X (by hypothesis). Likewise, for each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in Y$ and $v \in Y$, there exists an arc b in $G/Y \boxtimes G/X$ with $\mu(b) = ((\tilde{y}, u), (\tilde{y}, v)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Next, consider an arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X$ and $v \in Y$. For such an arc, in $G/Y \Box G/X$ there exist four arcs with label $\lambda(a)$, namely the arcs with $\mu = ((u, \tilde{x}), (\tilde{y}, \tilde{x}))$, $\mu = ((\tilde{y}, \tilde{x}), (\tilde{y}, v))$, $\mu = ((u, \tilde{x}), (u, v))$, and $\mu = ((u, v), (\tilde{y}, v))$. In $G/Y \Box G/X$, these four arcs are replaced by one arc b with $\mu(b) = ((u, \tilde{x}), (\tilde{y}, v)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Since there are no backward arcs in $[X, Y]$, the above arcs are the only arcs in $G/Y \boxtimes G/X$ induced by the vertices of Z. This completes the proof of Claim 1. $\Box$

We continue with the proof of Theorem 1. It remains to show that all other vertices of $G/Y \Box G/X$, except for the vertices of Z, disappear from $G/Y \boxtimes G/X$. This is clear for the vertex $(\tilde{y}, \tilde{x})$: all the arcs of $G/Y \Box G/X$ corresponding to the arcs of $[X, Y]$ are synchronising arcs of G/Y and G/X, so they disappear from $G/Y \boxtimes G/X$. Hence, $(\tilde{y}, \tilde{x})$ has in-degree 0 (and out-degree 0) in $G/Y \boxtimes G/X$, while it has *level* > 0 in $G/Y \Box G/X$. For the other vertices, the argument is as follows.

The vertex set of $G/Y \Box G/X$ consists of $Z \cup \{(\tilde{y}, \tilde{x})\}$ and the vertex set $X \times Y$. We will argue that all vertices of $X \times Y$ will eventually disappear from $G/Y \boxtimes G/X$.

First of all, we claim that all $(u, v) \in X \times Y$ have *level* > 0 in $G/Y \Box G/X$. This is obvious if u has *level* > 0 in $G[X]$ or v has *level* > 0 in $G[Y]$. Now let $(u, v) \in X \times Y$ such that u has *level* 0 in $G[X]$ and v has *level* 0 in $G[Y]$. Then the claim follows from the fact that v has at least one in-arc from a vertex in X, since $S'(G) \subseteq X$. In fact, since v has only in-arcs from vertices in X and u has no in-arcs at all, (u, v) has *level* 0 in $G/Y \boxtimes G/X$. This is because all arcs $(p, v) \in A(G)$ are in $[X, Y]$, hence they correspond to synchronising arcs in G/Y with respect to G/X. Concluding, all vertices $(u, v) \in X \times Y$ such that u has *level* 0 in $G[X]$ and v has *level* 0 in $G[Y]$ disappear from $G/Y \boxtimes G/X$, together with all the arcs with tail (u, v) for all such vertices $(u, v) \in X \times Y$. If after this first step there are still vertices of $X \times Y$ left in $G/Y \boxtimes G/X$, we can repeat the above arguments step by step for such remaining vertices $(u, v) \in X \times Y$ for which (u, v) has the lowest level in what has remained from $G/Y \boxtimes G/X$. Since $G/Y \boxtimes G/X$ is acyclic, it is clear that all vertices of $X \times Y$ disappear one by one from $G/Y \boxtimes G/X$. This completes the proof of Theorem 1. $\Box$

We are next going to provide some examples to show that none of the essential conditions in Theorem 1 can be omitted without violating the conclusion. First of all, it is clear that we need a proper partition of $V(G)$ into nonempty sets; otherwise the contractions cannot be carried out and the whole discussion is meaningless. In fact, the result is only meaningful if both of the partite sets have at least two vertices. For the other conditions, we show by small examples that they are essential for the validity of the conclusion.

One of the requirements is that the arcs of $[X, Y]$ have distinct labels. The example in Fig. 2 clearly shows that we cannot omit this requirement. Note that all the other conditions of Theorem 1 are met by this example graph.

The next requirement is that the arcs of G/X and G/Y corresponding to the arcs of $[X, Y]$ are the only synchronising arcs of G/X and G/Y. The examples in Fig. 3 show that this requirement cannot be omitted, without violating the conclusion, if we keep satisfying the other conditions of Theorem 1.

The case where $G[Y]$ and $[X, Y]$ have arcs with the same label is similar to the example in the right half of Fig. 3, by symmetry arguments.

Another essential requirement is that $S'(G) \subseteq X$, as the example graph of Fig. 4 shows. Here, $S'(G) \nsubseteq X$, but X still contains a vertex of $S'(G)$.

For the final requirement that $[X, Y]$ has no backward arcs, the situation is a bit different. In principle, the decomposition would still work fine if all the other conditions are met, but the problem here is that contraction may lead to graphs with directed cycles. So, such situations do not yield useful results in the context of our applications, and formally we did not define

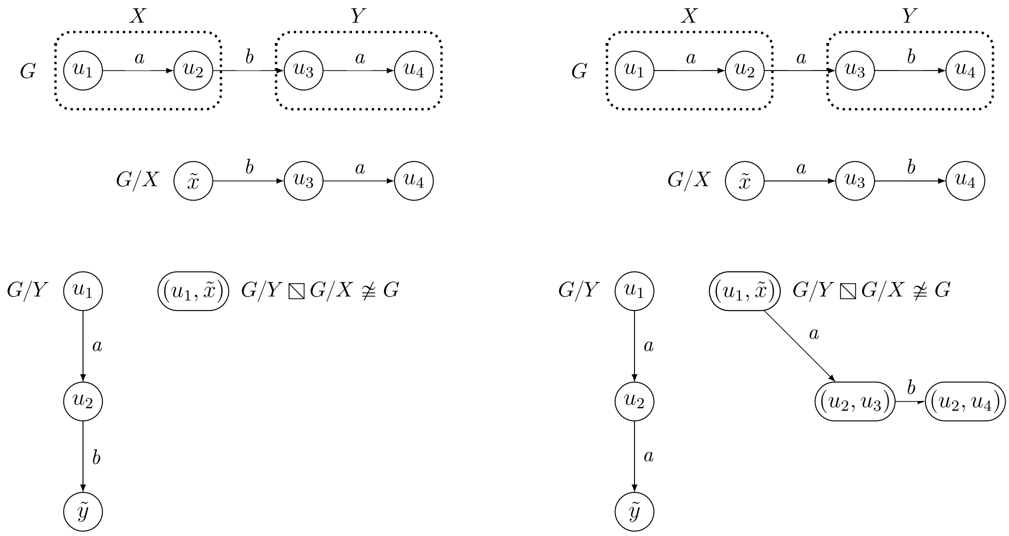

Fig. 3. Failing decomposition of G into G/Y and G/X, when $G[X]$ and $G[Y]$ (left example) or $G[X]$ and $[X, Y]$ (right example) have arcs with the same label.

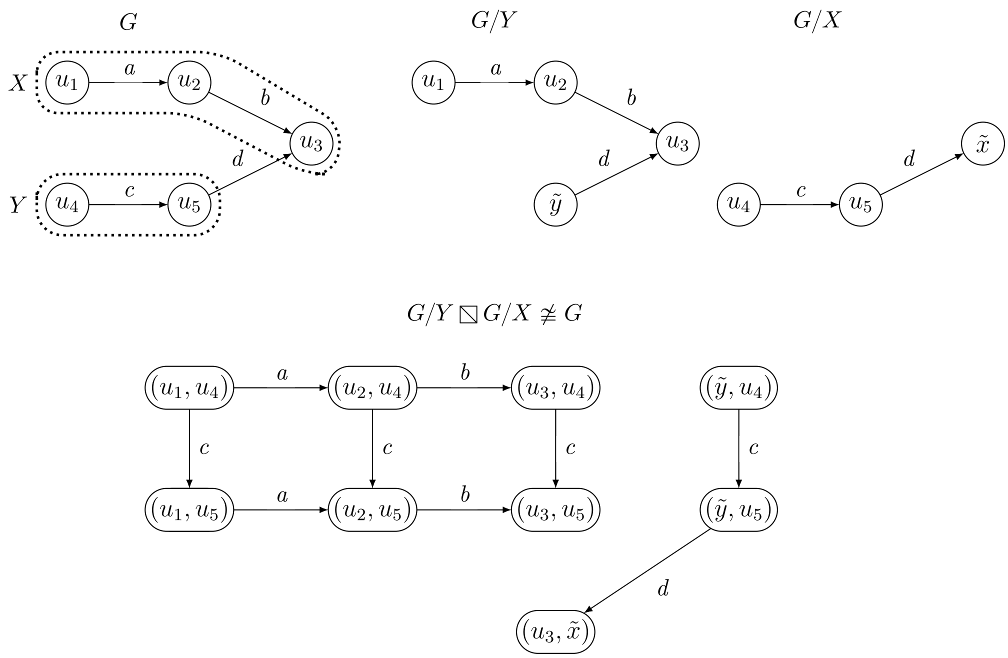

Fig. 4. Failing decomposition of G into G/Y and G/X, where $S'(G) \nsubseteq X$ and X contains a vertex of $S'(G)$.

the VRSP for such graphs. An example is shown in Fig. 5. Without giving the details, we observe that the decomposition is valid, i.e., $G \cong G/Y \boxslash G/X$, but both G/Y and G/X contain directed cycles (of length 2). Note, that the graph G in this figure admits another partition ($X = \{u_1, u_2\}$, $Y = \{u_3, u_4\}$) that satisfies all the conditions of Theorem 1. In fact, any acyclic directed graph has a partition of its vertex set into sets X and Y such that $[X, Y]$ only contains forward arcs, simply obtained by partitioning the graph according to the levels of its vertices: putting all vertices at $level < j$ in X and all vertices at $level \geq j$ in Y for a suitable value of j. Clearly, such partitions are easy to find, also algorithmically.

To conclude this section, we add a few remarks on the use of Theorem 1. Although this theorem has been presented for one (connected) graph G, we recall that the motivation behind the theorem is that the result enables us to replace a graph $G = \sum_{i=1}^{\omega(G)} G_i$ representing $\omega(G)$ processes by a new graph $G' = \sum_{i=1}^{\omega(G')} G'_i$ if at least one of the components G_i of G satisfies the conditions of the theorem. Obviously, Theorem 1 has quite a few restrictive requirements, so it is rather easy to come up with examples for which the theorem cannot be applied. On the other hand, there are cases in which we can still apply the

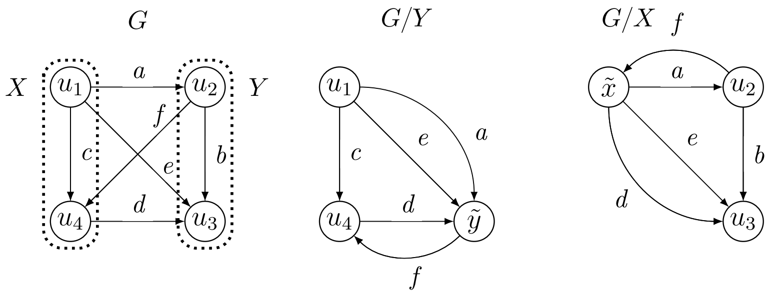

Fig. 5. Decomposition of G into G/Y and G/X, where $[X, Y]$ contains both forward and backward arcs.

theorem if the conditions are not met, e.g., if $[X, Y]$ has some backward arcs but G/X and G/Y are acyclic. In the next section, we present an alternative decomposition method for cases in which Theorem 1 cannot be applied.

6. A second decomposition result

In the second decomposition result, we are dealing with a partition of the vertex set into three instead of two nonempty sets. From this alternative partition, we again derive two new graphs: one by subsequently contracting two of the sets of the partition, and one by contracting only the third set. Formally, we define this as follows.

Let X_1, X_2 and Y be three disjoint nonempty subsets of $V(G)$, such that $Y = V(G) \setminus (X_1 \cup X_2)$. We use $G/X_1/X_2$ as shorthand for $(G/X_1)/X_2$. In the second decomposition theorem, we give a number of conditions that together guarantee that $G \cong G/Y \boxbslash G/X_1/X_2$. An example of this decomposition in which Y is a cut set that separates X_1 and X_2 in G is given in Fig. 6. A second example, in which Y is not a cut set is given in Fig. 7. As with Theorem 1, it is not difficult to give examples to show that none of the sufficient conditions can be omitted without violating the conclusion. We will not present such examples here, but they have appeared in the thesis of the first author [3].

Theorem 2. *Let G be a graph, and let X_1, X_2 and $Y = V(G) \setminus (X_1 \cup X_2)$ be three disjoint nonempty subsets of $V(G)$. Suppose that all the arcs of $[X_1, Y]$ have distinct labels, all the arcs of $[Y, X_2]$ have distinct labels, all the arcs of $[X_1, X_2]$ have distinct labels, the arcs of $[X_1, X_2]$ have no labels in common with any arcs in $[X_1, Y] \cup [Y, X_2]$, and that the arcs of $G/X_1/X_2$ and G/Y corresponding to the arcs of $[X_1, Y] \cup [Y, X_2] \cup [X_1, X_2]$ are the only synchronising arcs of $G/X_1/X_2$ and G/Y. If $S'(G) \subseteq X_1$, and $[X_1, Y]$, $[Y, X_2]$ and $[X_1, X_2]$ have no backward arcs, then $G \cong G/Y \boxbslash G/X_1/X_2$.*

Proof. The proof is analogous to the proof of Theorem 1. We present it here for completeness. It suffices to define a mapping $\phi : V(G) \to V(G/Y \boxbslash G/X_1/X_2)$ and to prove that ϕ is an isomorphism from G to $G/Y \boxbslash G/X_1/X_2$.

Let $\tilde{x}_1, \tilde{x}_2$ and $\tilde{y}$ be the new vertices replacing the sets X_1, X_2 and Y when defining $G/X_1/X_2$ and G/Y, respectively. Consider the mapping $\phi : V(G) \to V(G/Y \boxbslash G/X_1/X_2)$ defined by $\phi(u) = (u, \tilde{x}_1)$ for all $u \in X_1$, $\phi(v) = (v, \tilde{x}_2)$ for all $v \in X_2$ and $\phi(w) = (\tilde{y}, w)$ for all $w \in Y$.

Then ϕ is clearly a bijection if $V(G/Y \boxbslash G/X_1/X_2) = Z$, where Z is defined as $Z = \{(u, \tilde{x}_1) \mid u \in X_1\} \cup \{(v, \tilde{x}_2) \mid v \in X_2\} \cup \{(\tilde{y}, w) \mid w \in Y\}$. We are going to show this later by arguing that all the other vertices of $G/Y \square G/X_1/X_2$ will disappear from $G/Y \boxtimes G/X_1/X_2$. But first we are going to prove the following claim.

Claim 2. *The subgraph of $G/Y \boxtimes G/X_1/X_2$ induced by Z is isomorphic to G.*

Proof. Obviously, ϕ is a bijection from $V(G)$ to Z. It remains to show that this bijection preserves the arcs and their labels. By the definition of the Cartesian product, for each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X_1$ and $v \in X_1$, there exists an arc b in $G/Y \boxtimes G/X_1/X_2$ with $\mu(b) = ((u, \tilde{x}_1), (v, \tilde{x}_1)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Likewise, for each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in Y$ and $v \in Y$, there exists an arc b in $G/Y \boxtimes G/X_1/X_2$ with $\mu(b) = ((\tilde{y}, u), (\tilde{y}, v)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$, and for each arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X_2$ and $v \in X_2$, there exists an arc b in $G/Y \boxtimes G/X_1/X_2$ with $\mu(b) = ((u, \tilde{x}_2), (v, \tilde{x}_2)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Next, first consider an arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X_1$ and $v \in Y$. For such an arc, in $G/Y \square G/X_1/X_2$ there exist four arcs with label $\lambda(a)$, namely the arcs with $\mu = ((u, \tilde{x}_1), (\tilde{y}, \tilde{x}_1))$, $\mu = ((\tilde{y}, \tilde{x}_1), (\tilde{y}, v))$, $\mu = ((u, \tilde{x}_1), (u, v))$, and $\mu = ((u, v), (\tilde{y}, v))$. In $G/Y \boxtimes G/X_1/X_2$, these four arcs are replaced by one arc b with $\mu(b) = ((u, \tilde{x}_1), (\tilde{y}, v)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Secondly, consider an arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in Y$ and $v \in X_2$. For such an arc, in $G/Y \square G/X_1/X_2$ there also exist four arcs with label $\lambda(a)$, namely the arcs with $\mu = ((\tilde{y}, u), (v, u))$, $\mu = ((v, u), (v, \tilde{x}_2))$, $\mu = ((\tilde{y}, u), (\tilde{y}, \tilde{x}_2))$, and $\mu = ((\tilde{y}, \tilde{x}_2), (v, \tilde{x}_2))$. In $G/Y \boxtimes G/X_1/X_2$, these four arcs are replaced by one arc b with $\mu(b) = ((\tilde{y}, u), (v, \tilde{x}_2)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Thirdly, consider an arc $a \in A(G)$ with $\mu(a) = (u, v)$ for $u \in X_1$ and $v \in X_2$. For such an arc, in $G/Y \square G/X_1/X_2$ there also exist four arcs with label $\lambda(a)$, namely the arcs with $\mu = ((u, \tilde{x}_1), (u, \tilde{x}_2))$, $\mu = ((u, \tilde{x}_1), (v, \tilde{x}_1))$, $\mu = ((v, \tilde{x}_1), (v, \tilde{x}_2))$, and $\mu = ((u, \tilde{x}_2), (v, \tilde{x}_2))$. In $G/Y \boxtimes G/X_1/X_2$,

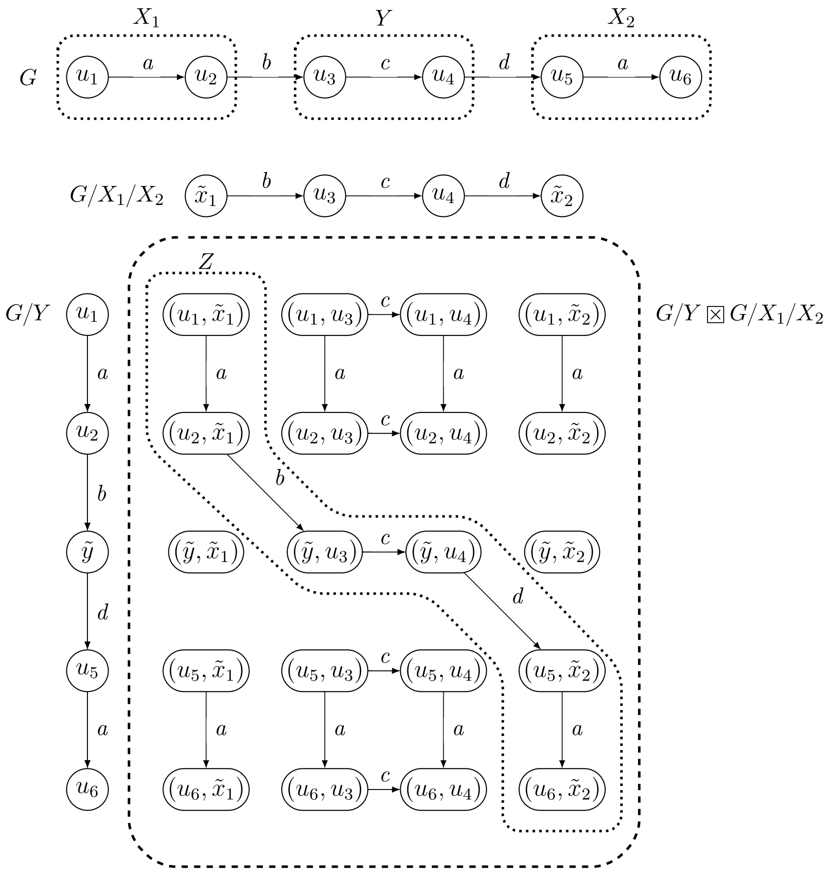

Fig. 6. Decomposition of G into G/Y and $G/X_1/X_2$, where Y is a cut set that separates X_1 and X_2 in G.

these four arcs are replaced by one arc b with $\mu(b) = ((u, \tilde{x}_1), (v, \tilde{x}_2)) = (\phi(u), \phi(v))$ and $\lambda(b) = \lambda(a)$. Since there are no backward arcs in $[X_1, Y]$, $[Y, X_2]$ and $[X_1, X_2]$, the above arcs are the only arcs in $G/Y \boxtimes G/X_1/X_2$ induced by the vertices of Z. This completes the proof of Claim 2. $\square$

We continue with the proof of Theorem 2. It remains to show that all other vertices of $G/Y \boxtimes G/X_1/X_2$, except for the vertices of Z, disappear from $G/Y \boxtimes G/X_1/X_2$. This is clear for the vertex $(\tilde{y}, \tilde{x}_1)$: all the arcs of $G/Y \Box G/X_1/X_2$ corresponding to the arcs of $[X_1, Y]$ are synchronising arcs of G/Y and $G/X_1/X_2$, so they disappear from $G/Y \boxtimes G/X_1/X_2$. Hence, $(\tilde{y}, \tilde{x}_1)$ has in-degree 0 in $G/Y \boxtimes G/X_1/X_2$, while it has *level* > 0 in $G/Y \Box G/X_1/X_2$. For the other vertices, the argument is as follows.

The vertex set of $G/Y \Box G/X_1/X_2$ consists of the union of $Z \cup \{(\tilde{y}, \tilde{x}_1), (\tilde{y}, \tilde{x}_2)\}$ and the vertex sets $(X_1 \cup X_2) \times Y$, $X_1 \times \{\tilde{x}_2\}$ and $X_2 \times \{\tilde{x}_1\}$. We will argue that all vertices of $(X_1 \cup X_2) \times Y$, $X_1 \times \{\tilde{x}_2\}$ and $X_2 \times \{\tilde{x}_1\}$, as well as the vertex $(\tilde{y}, \tilde{x}_2)$ will eventually disappear from $G/Y \boxtimes G/X_1/X_2$.

Firstly, we claim that all $(u, v) \in X_1 \times Y$ have *level* > 0 in $G/Y \Box G/X_1/X_2$. This is obvious if u has *level* > 0 in $G[X_1]$ or v has *level* > 0 in $G[Y]$. Now let $(u, v) \in X_1 \times Y$ such that u has *level* 0 in $G[X_1]$ and v has *level* 0 in $G[Y]$. Then the claim follows from the fact that v has at least one in-arc from a vertex in X_1, since $S'(G) \subseteq X_1$. In fact, since v has only in-arcs from vertices in X_1 and u has no in-arcs at all, (u, v) has *level* 0 in $G/Y \boxtimes G/X_1/X_2$. Hence, all vertices $(u, v) \in X_1 \times Y$ such that u has *level* 0 in $G[X_1]$ and v has *level* 0 in $G[Y]$ disappear from $G/Y \boxtimes G/X_1/X_2$, together with all the arcs with tail (u, v) for all such vertices $(u, v) \in X_1 \times Y$. If after this first step there are still vertices of $X_1 \times Y$ left in $G/Y \boxtimes G/X_1/X_2$, we can repeat the above arguments step by step for such remaining vertices $(u, v) \in X_1 \times Y$ for which (u, v) has the lowest level in what has remained from $G/Y \boxtimes G/X_1/X_2$. Since $G/Y \boxtimes G/X_1/X_2$ is acyclic, it is clear that all vertices of $X_1 \times Y$ disappear one by one from $G/Y \boxtimes G/X_1/X_2$. Now, since $(\tilde{y}, \tilde{x}_2)$ has possibly only in-arcs from vertices $(u, v) \in X_1 \times Y$, $(\tilde{y}, \tilde{x}_2)$ will disappear as well.

Next, we claim that all $(u, v) \in X_2 \times Y$ have *level* > 0 in $G/Y \Box G/X_1/X_2$. This is obvious if u has *level* > 0 in $G[X_2]$ or v has *level* > 0 in $G[Y]$. Now let $(u, v) \in X_2 \times Y$ such that u has *level* 0 in $G[X_2]$ and v has *level* 0 in $G[Y]$. Then the claim follows from the fact that u has at least one in-arc from a vertex in Y, since $[Y, X_2]$ has only forward arcs. In fact, since u has only in-arcs from vertices in Y and v has no in-arcs at all, (u, v) has *level* 0 in $G/Y \boxtimes G/X_1/X_2$. Hence, all vertices $(u, v) \in X_2 \times Y$ such that u has *level* 0 in $G[X_2]$ and v has *level* 0 in $G[Y]$ disappear from $G/Y \boxtimes G/X_1/X_2$, together with all the arcs with tail (u, v) for all such vertices $(u, v) \in X_2 \times Y$. If after this first step there are still vertices of $X_2 \times Y$ left in $G/Y \boxtimes G/X_1/X_2$, we can

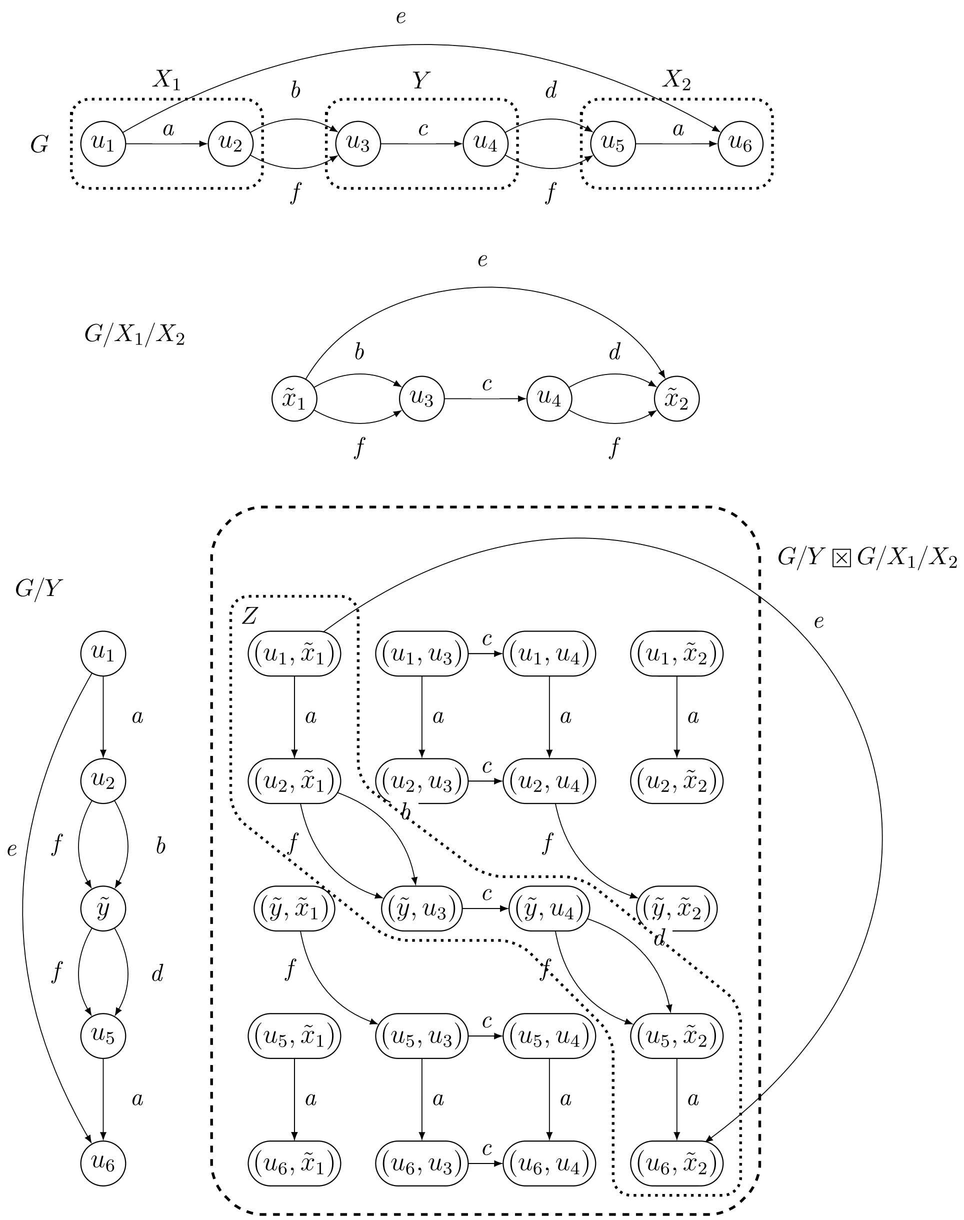

Fig. 7. Decomposition of G into G/Y and $G/X_1/X_2$, where Y does not separate X_1 and X_2 in G.

repeat the above arguments step by step for such remaining vertices $(u, v) \in X_2 \times Y$ for which (u, v) has the lowest level in what has remained from $G/Y \boxtimes G/X_1/X_2$. Since $G/Y \boxtimes G/X_1/X_2$ is acyclic, it is clear that all vertices of $X_2 \times Y$ disappear one by one from $G/Y \boxtimes G/X_1/X_2$.

We continue with the claim that all $(u, \tilde{x}_1) \in X_2 \times \{\tilde{x}_1\}$ have *level* > 0 in $G/Y \Box G/X_1/X_2$. This is obvious if u has *level* > 0 in $G[X_2]$. Now let $(u, \tilde{x}_1) \in X_2 \times \{\tilde{x}_1\}$ such that u has *level* 0 in $G[X_2]$. Then the claim follows from the fact that u has at least one in-arc from a vertex in Y, since $[Y, X_2]$ has only forward arcs. In fact, since u has only in-arcs from vertices in Y and $\tilde{x}_1$ has no in-arcs at all, $(u, \tilde{x}_1)$ has *level* 0 in $G/Y \boxtimes G/X_1/X_2$. Hence, all vertices $(u, \tilde{x}_1) \in X_2 \times \{\tilde{x}_1\}$ such that u has *level* 0 in $G[X_2]$ disappear from $G/Y \boxtimes G/X_1/X_2$, together with all the arcs with tail $(u, \tilde{x}_1)$ for all such vertices $(u, \tilde{x}_1) \in X_2 \times \{\tilde{x}_1\}$. If after this first step there are still vertices of $X_2 \times \{\tilde{x}_1\}$ left in $G/Y \boxtimes G/X_1/X_2$, we can repeat the above arguments step by step for such remaining vertices $(u, \tilde{x}_1) \in X_2 \times \{\tilde{x}_1\}$ for which $(u, \tilde{x}_1)$ has the lowest level in what has remained from $G/Y \boxtimes G/X_1/X_2$. Since $G/Y \boxtimes G/X_1/X_2$ is acyclic, it is clear that all vertices of $X_2 \times \{\tilde{x}_1\}$ disappear one by one from $G/Y \boxtimes G/X_1/X_2$.

Finally, we claim that all $(u, \tilde{x}_2) \in X_1 \times \{\tilde{x}_2\}$ have *level* > 0 in $G/Y \Box G/X_1/X_2$. This is obvious if u has *level* > 0 in $G[X_1]$. Now let $(u, \tilde{x}_2) \in X_1 \times \{\tilde{x}_2\}$ such that u has *level* 0 in $G[X_1]$. Then the claim follows from the fact that $\tilde{x}_2$ has at least one in-arc from a vertex in Y, since $[Y, X_2]$ has only forward arcs and $S'(G) \subseteq X_1$ by hypothesis. Noting that $\tilde{x}_2$ has only in-arcs from vertices in Y, and all $u \in S'(G) \subseteq X_1$ have no in-arcs at all, clearly for all $u \in S'(G) \subseteq X_1$, $(u, \tilde{x}_2)$ has *level* 0 in $G/Y \boxtimes G/X_1/X_2$. Hence, all vertices $(u, \tilde{x}_2) \in X_1 \times \{\tilde{x}_2\}$ such that u has *level* 0 in $G[X_1]$ disappear from $G/Y \boxtimes G/X_1/X_2$, together with all the arcs with tail $(u, \tilde{x}_2)$ for all such vertices $(u, \tilde{x}_2) \in X_1 \times \{\tilde{x}_2\}$. If after this first step there are still vertices of $X_1 \times \{\tilde{x}_2\}$ left in $G/Y \boxtimes G/X_1/X_2$, we can repeat the above arguments step by step for such remaining vertices $(u, \tilde{x}_2) \in X_1 \times \{\tilde{x}_2\}$ for which $(u, \tilde{x}_2)$ has the lowest level in what has remained from $G/Y \boxtimes G/X_1/X_2$. Since $G/Y \boxtimes G/X_1/X_2$ is acyclic, it is clear that all vertices of $X_1 \times \{\tilde{x}_2\}$ disappear one by one from $G/Y \boxtimes G/X_1/X_2$.

This completes the proof of Theorem 2. □

7. Applications for undirected graphs

We developed the decomposition tools of the previous sections for labelled acyclic directed multigraphs, since these graphs appeared as natural models for the processes and actions in the application area of robotics. In this section, we will show how the tools can be applied for decomposing undirected graphs.

The idea is simple. Let $G = (V, E)$ be a connected undirected (simple or multi)graph. We can orient G, i.e., give directions to the edges of E (replacing each edge $uv \in E$ by one arc a with $\mu(a) = (u, v)$ or $\mu(a) = (v, u)$), in such a way that the resulting directed graph is acyclic. One way to do this, is by just starting at an arbitrary vertex $v \in V$, replacing all edges incident with v by out-arcs of v (so v is not on any directed cycle in the resulting directed graph), considering the graph $G - v$, and repeating this procedure until no edges are left in the remaining graph. We clearly end up with an acyclic directed graph. The ordering in which we choose the vertices in this procedure implies that for every arc of the resulting directed graph, its tail precedes its head in the ordering. Such an ordering is known as a *topological ordering* or a *topological sort* of the vertices of the resulting directed graph. Obviously, in general an acyclic directed graph can have different topological sorts, and it is a known fact that a directed graph is acyclic if and only if it admits a topological sort (See, e.g., [2, Exercise 2.1.11]).

Once we have this connected directed acyclic graph D, we can use any of its suitable arc cuts for our purpose of defining a decomposition. One of the obvious choices would be an arc cut $[X, Y]$ consisting of the arcs between vertices at *level* $\leq j$ and *level* $\geq j + 1$ for a suitable choice of j, i.e., a choice such that $|X| \geq 2$ and $|Y| \geq 2$. But many other options are possible, in general. It is not difficult to see that any connected graph on at least four vertices admits such a decomposition. If we then assign different labels to all the arcs of D, the arcs in $[X, Y]$ are the only synchronising arcs of D/Y and D/X, and all the conditions of Theorem 1 are satisfied. This means that D can be decomposed into D/Y and D/X. This decomposition implies a decomposition for the associated undirected graph G. Such decompositions might turn out to be useful, e.g., as an ingredient of induction proofs within structural graph theory or of recursive methods within algorithmic graph theory. These potential applications are beyond the scope of this paper.

Acknowledgements

The authors would like to express their gratitude to the anonymous reviewers for their useful suggestions and comments. The research of the first author has been funded by the InHolland University of Applied Sciences, Alkmaar, The Netherlands.

References

[1] C. Baier, J.P. Katoen, Principles of Model Checking, MIT Press, ISBN: 978-0-262-02649-9, 2008.
[2] J.A. Bondy, U.S.R. Murty, Graph Theory, Springer, Berlin, 2008.
[3] A.H. Boode, On the automation of periodic hard real-time processes: a graph-theoretical approach (Ph.D. thesis), University of Twente, 2018, http://dx.doi.org/10.3990/1.9789036545518.
[4] A.H. Boode, J.F. Broenink, Performance of periodic real-time processes: a vertex-removing synchronised graph product, in: Communicating Process Architectures 2014, Oxford, UK, 36th WoTUG conference on concurrent and parallel programming, Open Channel Publishing Ltd., Bicester, 2014, pp. 119–138.
[5] A.H. Boode, H.J. Broersma, J.F. Broenink, Improving the performance of periodic real-time processes: a graph theoretical approach, in: Communicating Process Architectures 2013, Edinburgh, UK, 35th WoTUG conference on concurrent and parallel programming, Open Channel Publishing Ltd., Bicester, 2013, pp. 57–79.
[6] A.H. Boode, H.J. Broersma, J.F. Broenink, On a directed tree problem motivated by a newly introduced graph product, Electr. J of Graph Theory and Appl. 3 (2015) 162–181.
[7] R. Hammack, W. Imrich, S. Klavžar, Handbook of Product Graphs. Discrete Mathematics and its Applications (Boca Raton), second ed., CRC Press, Boca Raton, FL, 2011.
[8] C.A.R. Hoare, Communicating sequential processes, Commun. ACM 21 (1978) 666–677.
[9] J. Magee, J. Kramer, Concurrency: State Models & Java Programs, John Wiley & Sons, Inc, New York, NY, USA, ISBN: 0-471-98710-7, 1999.
[10] R. Milner, Communication and Concurrency, Prentice-Hall, Inc, Upper Saddle River, NJ, USA, ISBN: 0-13-115007-3, 1989.
[11] S. Wöhrle, W. Thomas, Model checking synchronized products of infinite transition systems, in: Proc. 19th LICS, IEEE Comp. Soc, IEEE Computer Society Press, 2004, pp. 2–11.