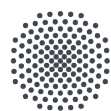
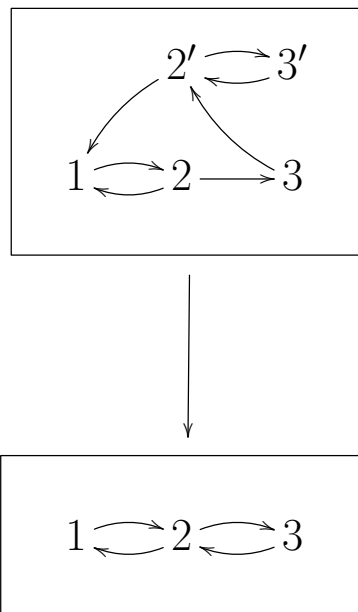


On the Bisson-Tsemo model category of graphs

Jannik Hess

Master Thesis

January 2022



University of Stuttgart
Germany

Contents

0	Introduction	5
0.1	Graphs	5
0.2	Graph morphisms	5
0.3	A model category structure on \mathbf{Gph} by Bisson and Tsemo	6
0.4	Proof that \mathbf{Gph} is a model category	8
0.5	A sufficient condition for a graph morphism to be a quasiisomorphism	10
0.6	Examples and counterexamples	11
0.6.1	Examples	11
0.6.2	Counterexamples	17
1	Preliminaries	19
1.1	Preliminaries on categories	19
1.1.1	The properties (2 of 6) and (2 of 3)	19
1.1.2	Pushout and Pullback	21
1.1.3	Lifting properties	28
1.1.4	Subsets of $\mathbf{Mor}(\mathcal{C})$ being closed under retracts	34
1.2	Preliminaries on sets	38
1.2.1	Elementary constructions and properties	38
1.2.2	Pushouts in \mathbf{Set}	40
1.2.3	Pullbacks in \mathbf{Set}	45
1.2.4	Colimit of a countable chain in \mathbf{Set}	47
2	Graphs	51
2.1	Definitions for graphs and graph morphisms	51
2.2	Thin graphs	60
2.3	Pushout and pullback of graphs	64
2.3.1	Pushout of graphs	64
2.3.2	Coproducts	69
2.3.3	Pullback of graphs	75
2.4	Colimit of a countable chain in \mathbf{Gph}	78
2.5	Tree graphs	88
3	Properties of graph morphisms	93
3.1	Quasiisomorphisms	93
3.2	Fibrations and fibrant graphs	98
3.3	Acyclic fibrations	104
3.4	Cofibrations and cofibrant graphs	106
3.5	Bifibrant graphs	113
3.6	Acyclic cofibrations	113
3.7	Summary of some notations	133
4	Factorization of graph morphisms	135
5	Subsets of $\mathbf{Mor}(\mathbf{Gph})$ and their lifting sets	147
6	\mathbf{Gph} is a model category	155
7	A sufficient condition for a graph morphism to be a quasiisomorphism	165

8	Duality	173
9	Some examples and counterexamples	177
9.1	Some examples for quasiisomorphisms	177
9.2	Some examples of graph morphisms related to the sufficient condition of Proposition 210	211
9.3	Some inequalities of subsets of $\text{Mor}(\text{Gph})$	215
9.4	Counterexamples for model categories	219
9.4.1	Elementary counterexamples	219
9.4.2	Counterexamples for pushouts and pullbacks	225
9.5	Counterexamples in Gph	234
10	Algorithmic treatment of graphs	239
10.1	Implementation of graphs	239
10.2	Implementation of graph morphisms	242
10.3	Calculating a pushout and a pullback of graphs	251
10.3.1	Calculating a pushout of graphs	251
10.3.2	Calculating a pullback of graphs	253
10.3.3	Calculating induced morphisms of pushouts and pullbacks of graphs	254
10.4	Calculating tree graphs	257
10.5	Testing properties of graph morphisms	261
10.6	Testing the sufficient condition of Proposition 210 for graph morphisms	264
10.7	Functions to calculate examples in section 9.1	266
10.8	Functions to calculate more examples	273
10.9	More useful functions	276
A	Explanation for electronic appendix	279

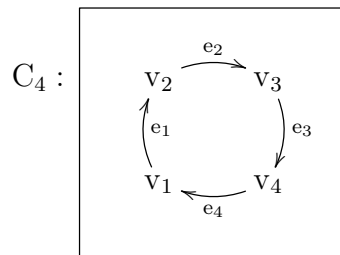
Chapter 0

Introduction

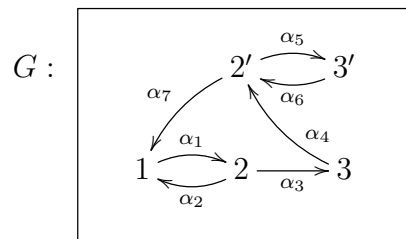
0.1 Graphs

A graph G consists of a set of vertices V_G and of a set of edges E_G together with a source map $s_G : E_G \rightarrow V_G$ and a target map $t_G : E_G \rightarrow V_G$, mapping an edge to its source respectively to its target; cf. Definition 45.⁽¹⁾ Pictorially, we represent a graph by writing out the vertices of G and by drawing an arrow for each edge, pointing from its source to its target.

For example, we have the following cyclic graph.



Or, for example, we have the following graph.



So e.g. the edge α_3 has source $\alpha_3 s_G = 2$ and target $\alpha_3 t_G = 3$.

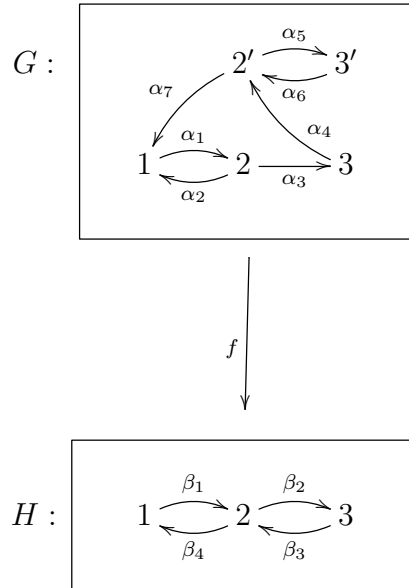
0.2 Graph morphisms

A graph morphism $f : G \rightarrow H$ between graphs G and H consists of a map $V_f : V_G \rightarrow V_H$ on the vertices of the graphs and a map $E_f : E_G \rightarrow E_H$ on the edges of the graphs such that

⁽¹⁾In the literature, graphs in this sense are also called “directed graphs”.

$E_f s_H = s_G V_f$ and $E_f t_H = t_G V_f$; cf. Definition 54.

For instance, we have the following graph morphism $f : G \rightarrow H$, mapping the vertices and the edges in a vertical way; cf. Example 215.



We have e.g. $\alpha_4 E_f t_H = \beta_3 t_H = 2 = 2' V_f = \alpha_4 t_G V_f$.

The category of graphs and graph morphisms is denoted by Gph .

We denote the set of graph morphisms from G to H by $(G, H)_{\text{Gph}}$; cf. Definition 64.

Given a graph morphism $f : G \rightarrow H$ and a graph K , we have the map

$$(K, f)_{\text{Gph}} : (K, G)_{\text{Gph}} \rightarrow (K, H)_{\text{Gph}} : g \mapsto gf ;$$

cf. Definition 68.

0.3 A model category structure on Gph by BISSON and TSEMO

BISSON and TSEMO define data for a model category structure on the category Gph as follows.

A graph morphism $f : G \rightarrow H$ is called a quasiisomorphism if the map

$$(C_k, f)_{\text{Gph}} : (C_k, G)_{\text{Gph}} \rightarrow (C_k, H)_{\text{Gph}}$$

is bijective for $k \geq 1$. In other words, we require each graph morphism $C_k \rightarrow H$ to have a unique lift $C_k \rightarrow G$ along $G \xrightarrow{f} H$. Let $\text{Qis} \subseteq \text{Mor}(\text{Gph})$ denote the subset of quasiisomorphisms; cf. Definition 115. A quasiisomorphism is written $G \xrightarrow{\sim} H$. E.g. the example displayed in §0.2 is a quasiisomorphism.

For a graph G and a vertex $v \in V_G$, we denote by $G(v, *) := \{e \in E_G : e s_G = v\}$ the set of edges with source v .

For a graph morphism $f : G \rightarrow H$ and a vertex $v \in V_G$, we denote

$$\begin{aligned} E_{f,v} &:= E_f|_{G(v,*)}^{H(vV_f,*)} : G(v,*) \rightarrow H(vV_f,*) \\ &e \mapsto eE_f . \end{aligned}$$

A graph morphism $f : G \rightarrow H$ is called a fibration if the map

$$E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$$

is surjective for $v \in V_G$; cf. Definition 127.(1). Let $\text{Fib} \subseteq \text{Mor}(\text{Gph})$ denote the subset of fibrations. A fibration is written $G \twoheadrightarrow H$.

A graph morphism $f : G \rightarrow H$ is called an etale fibration if the map

$$E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$$

is bijective for $v \in V_G$; cf. Definition 127.(2). This notion will play a role in a sufficient condition on a morphism to be a quasiisomorphism; cf. §0.5.

A graph morphism $f : G \rightarrow H$ is called an acyclic cofibration if the properties (AcCofib 1–5) hold; cf. Definition 162. Pictorially, an acyclic cofibration is obtained as follows. Let G be a graph. Glue some trees at their roots to vertices of G to obtain the graph H . Then the inclusion morphism $\iota : G \rightarrow H$ is an acyclic cofibration. Every acyclic cofibration is essentially obtained in this way.

Let $\text{AcCofib} \subseteq \text{Mor}(\text{Gph})$ denote the subset of acyclic cofibrations. An acyclic cofibration is written $G \twoheadrightarrow H$.

Let $\text{AcFib} := \text{Qis} \cap \text{Fib} \subseteq \text{Mor}(\text{Gph})$ denote the subset of acyclic fibrations. An acyclic fibration is written $G \dashrightarrow H$. E.g. the example displayed in §0.2 is an acyclic fibration.

A graph morphism $f : G \rightarrow H$ is called a cofibration if it has the left lifting property with respect to AcFib . That is, $f : G \rightarrow H$ is a cofibration if, given a commutative quadrangle

$$\begin{array}{ccc} G & \xrightarrow{a} & X \\ f \downarrow & & \downarrow g \\ H & \xrightarrow{b} & Y \end{array}$$

in Gph , there exists a graph morphism $h : H \rightarrow X$ such that the diagram

$$\begin{array}{ccc} G & \xrightarrow{a} & X \\ f \downarrow & \nearrow h & \downarrow g \\ H & \xrightarrow{b} & Y \end{array}$$

commutes.

Let $\text{Cofib} \subseteq \text{Mor}(\text{Gph})$ denote the subset of cofibrations. A cofibration is written $G \dashrightarrow H$.

Then Gph , together with Qis , Fib and Cofib , is a Quillen closed model category; cf. [3, Cor. 4.8], [4, Ch. I, §1, Def. 1; Ch. I, §5, Def. 1], Definition 198.(4), Proposition 204.

Moreover, for the set AcCofib , which is defined via (AcCofib 1–5) in Definition 162, we actually have the equality $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$; cf. Lemma 185.

0.4 Proof that Gph is a model category

Following BISSON and TSEMO [3, Cor. 4.8], we will show that Gph is a Quillen closed model category, using the data explained in §0.3; cf. Definition 198.(4).

Note that we have defined Cofib, Qis and AcCofib separately; cf. §0.3. We will show that $\text{Cofib} \cap \text{Qis} = \text{AcCofib}$, but this will only be possible having factorization of morphisms in acyclic cofibrations and fibrations at our disposal; cf. Lemma 185.

First, we show that Qis, Fib and Cofib are closed under composition and under retracts.

Moreover, Qis, Fib and AcFib are shown to be stable under pullbacks, and Cofib and AcCofib are shown to be stable under pushouts.

In Assertion 251, we give an example that shows that Qis is not stable under pushouts along cofibrations.

Suppose given subsets $M, N \subseteq \text{Mor}(\text{Gph})$. We write $M \boxtimes N$ if for every commutative quadrangle

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ m \downarrow & & \downarrow n \\ Y & \xrightarrow{b} & Y' \end{array}$$

in Gph with $m \in M$ and $n \in N$, there exists a morphism $h : Y \rightarrow X'$ in Gph such that

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ m \downarrow & \nearrow h & \downarrow n \\ Y & \xrightarrow{b} & Y' \end{array}$$

commutes.

By definition, we have $\text{Cofib} \boxtimes \text{AcFib}$.

To show that $\text{AcCofib} \boxtimes \text{Fib}$, a given acyclic cofibration is factored in successive inclusion morphisms, in each of which single-edge-trees are glued to the subgraph; cf. Lemma 174. Then to construct the required lift, the definition of a fibration can be used directly.

We have to show that each morphism $f : X \rightarrow Y$ can be factored into an acyclic cofibration followed by a fibration.

To this end, a resolution $p : F \rightarrow Y$ of Y is constructed, where F is a disjoint union of trees. This resolution is glued, via pushout, to the discrete subgraph \dot{X} of X .

$$\begin{array}{ccc} \dot{X} & \xrightarrow{\circ} & F \\ \downarrow & & \downarrow \\ X & \xrightarrow{\circ} & \tilde{F} \\ & \searrow f & \downarrow \\ & & Y \end{array}$$

$\begin{array}{c} \nearrow p \\ \downarrow \\ \searrow \end{array}$

Using this factorization property, we are now able to show that we have in fact $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$.

We have to show that each morphism $f : X \rightarrow Y$ can be factored into a cofibration followed by an acyclic fibration.

Using a disjoint union \tilde{X} of X with cyclic graphs, we factor f into a cofibration $X \xrightarrow{c} \tilde{X}$ followed by a graph morphism $\tilde{f} : \tilde{X} \rightarrow Y$ that becomes surjective under $(C_k, -)_{\text{Gph}}$ for $k \geq 1$.

$$\begin{array}{ccc} & \tilde{X} & \\ c \nearrow & & \searrow \tilde{f} \\ X & \xrightarrow{f} & Y \end{array}$$

Now we consider $\tilde{f} : \tilde{X} \rightarrow Y$. We iteratively glue cycles in the source graph using a pushout construction, where the definition of the sets M' and M ensure that cycles that map to the same cycle in Y are glued together.

$$\begin{array}{ccccc} \coprod_{M'} C_n & \xrightarrow{\quad} & \coprod_M C_n & & \\ \downarrow & & \downarrow & \searrow c & \\ \tilde{X}_i & \xrightarrow{g_{i,i+1}} & \tilde{X}_{i+1} & & Y \\ \uparrow g_{0,i} & \nearrow g_{0,i+1} & \searrow \tilde{f}_i & \nearrow \tilde{f}_{i+1} & \\ \tilde{X} & \xrightarrow{\tilde{f}} & & & Y \end{array}$$

This yields a factorization over the direct limit \tilde{X}_∞ as follows.

$$\begin{array}{ccc} & \tilde{X}_\infty & \\ g_{0,\infty} \nearrow & & \searrow \tilde{f}_\infty \\ \tilde{X} & \xrightarrow{\tilde{f}} & Y \end{array}$$

Using a factorization of \tilde{f}_∞ into an acyclic cofibration $h : \tilde{X}_\infty \xrightarrow{\circ} \tilde{\tilde{X}}$ followed by a fibration $\tilde{\tilde{f}} : \tilde{\tilde{X}} \rightarrow Y$, which then is an acyclic fibration, we obtain

$$\begin{array}{ccc} & \tilde{X}_\infty & \xrightarrow{h} \tilde{\tilde{X}} \\ g_{0,\infty} \nearrow & & \searrow \tilde{f}_\infty \\ \tilde{X} & \xrightarrow{\tilde{f}} & Y \\ c \nearrow & & \searrow \tilde{\tilde{f}} \\ X & \xrightarrow{f} & Y \end{array}$$

as required.

0.5 A sufficient condition for a graph morphism to be a quasiisomorphism

Suppose given graphs G and H .

Suppose given a graph morphism $f : G \rightarrow H$.

An edge e_H of H is called unitargeting with respect to f if we have

$$|\{\tilde{e} \text{ t}_G : \tilde{e} \in \mathbf{E}_G, \tilde{e} \mathbf{E}_f = e_H\}| = 1,$$

that is, if its preimage in G has a unique target; cf. Definition 206.

Consider the following property.

(Uni) For $n \geq 1$ and each graph morphism $u : C_n \rightarrow H$, there exists $i \in \mathbb{Z}/n\mathbb{Z}$ such that $e_i \mathbf{E}_u \in \mathbf{E}_H$ is unitargeting with respect to f .

Pictorially speaking, f satisfies (Uni) if every cycle in H contains at least a unitargeting edge. In practice, one removes all unitargeting edges from H to obtain a subgraph \tilde{H} , and one verifies that $(C_n, \tilde{H})_{\text{Gph}} = \emptyset$ for $n \geq 1$ in order to verify that (Uni) holds.

We show that if $f : G \rightarrow H$ is an etale fibration and satisfies (Uni), then it is a quasiisomorphism; cf. Proposition 210.⁽²⁾

E.g. the graph morphism $f : G \rightarrow H$ in §0.2 is a quasiisomorphism since it verifies this sufficient condition as follows.

First, $f : G \rightarrow H$ is an etale fibration; cf. Definition 127.(2). For instance, $G(2', *) = \{\alpha_7, \alpha_5\}$ maps bijectively to $H(2, *) = \{\beta_4, \beta_2\}$ via $\mathbf{E}_{f,2'}$ since $\alpha_7 \mapsto \beta_4$ and $\alpha_5 \mapsto \beta_2$.

Second, we show that f satisfies (Uni). The edge β_3 is unitargeting since $\alpha_4 \text{ t}_G = \alpha_6 \text{ t}_G = 2'$. The edge β_1 is unitargeting since α_1 is its only preimage. The edge β_4 is unitargeting since $\alpha_2 \text{ t}_G = \alpha_7 \text{ t}_G = 1$. But the edge β_2 is not unitargeting since $\alpha_3 \text{ t}_G = 3 \neq 3' = \alpha_5 \text{ t}_G$. Obtaining \tilde{H} by removing the unitargeting edges in H there is just the edge β_2 left in \tilde{H} , i.e. $\mathbf{E}_{\tilde{H}} = \{\beta_2\}$. So there does not exist any graph morphism $C_n \rightarrow \tilde{H}$ for $n \geq 1$.

Hence the graph morphism $f : G \rightarrow H$ in §0.2 is a quasiisomorphism.

There is a dual counterpart to this sufficient condition, obtainable as follows.

Given a graph G , we define the opposite graph G^{op} by letting $s_{G^{\text{op}}} := t_G$ and $t_{G^{\text{op}}} := s_G$.

Given a graph morphism $f : G \rightarrow H$ we define the opposite graph morphism

$$f^{\text{op}} = (\mathbf{V}_{f^{\text{op}}}, \mathbf{E}_{f^{\text{op}}}) : G^{\text{op}} \rightarrow H^{\text{op}}$$

by $\mathbf{V}_{f^{\text{op}}} := \mathbf{V}_f : \mathbf{V}_G \rightarrow \mathbf{V}_H$ and $\mathbf{E}_{f^{\text{op}}} := \mathbf{E}_f : \mathbf{E}_G \rightarrow \mathbf{E}_H$.

Then $f^{\text{op}} : G^{\text{op}} \rightarrow H^{\text{op}}$ is a quasiisomorphism if and only if $f : G \rightarrow H$ is a quasiisomorphism.

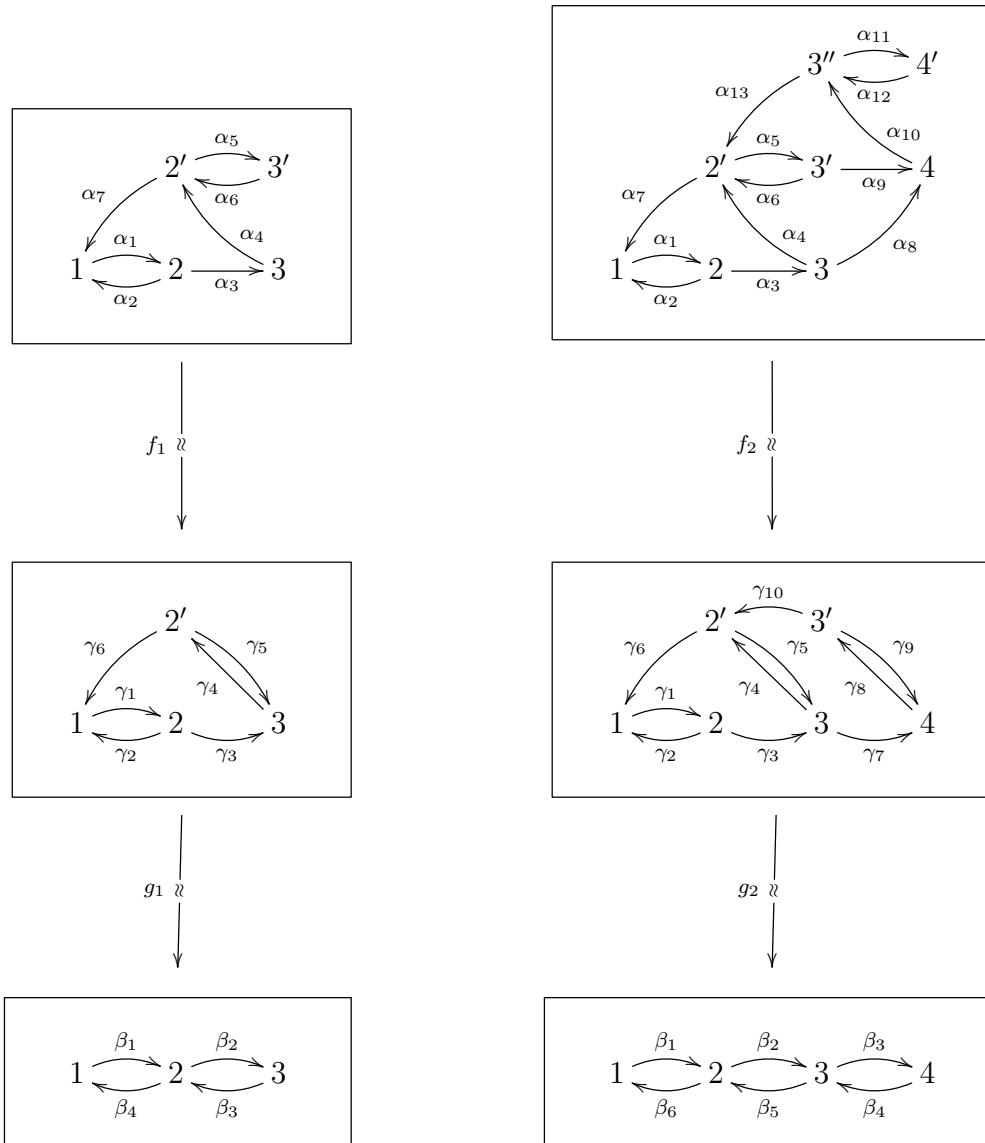
⁽²⁾Special thanks to Konrad Unger and Lukas Wiedmann who asked persistently if thinness is actually needed here. This led to a removal of this unnecessary condition.

0.6 Examples and counterexamples

0.6.1 Examples

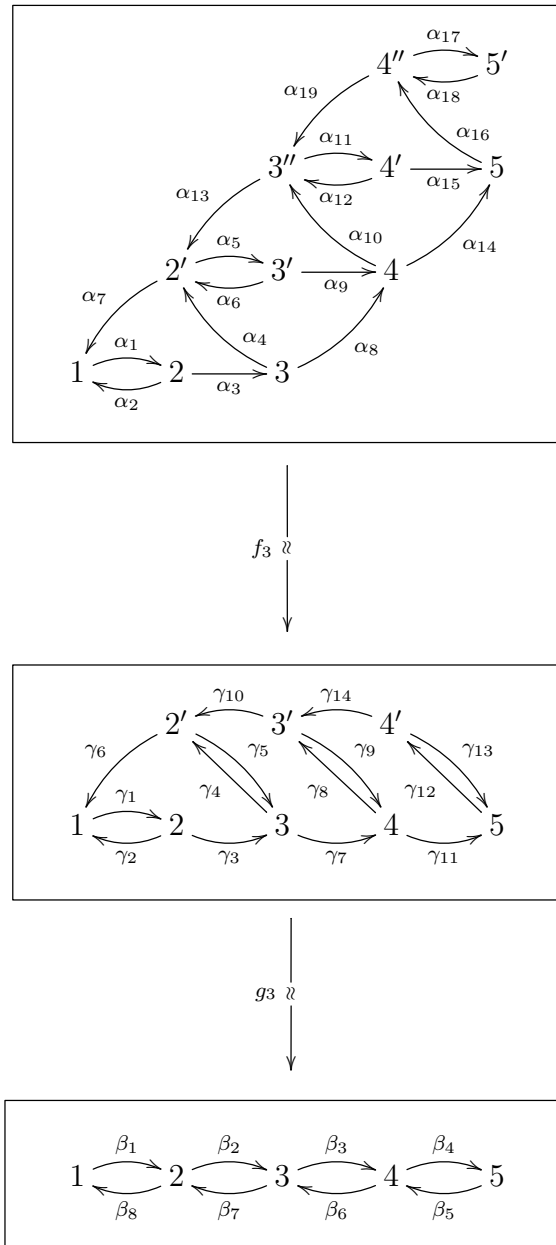
For sake of illustration, we show several examples for quasiisomorphisms obtained by the sufficient condition in Proposition 210, which we verify via Magma [2] in §9.1 using the functions given in §10.7.

We map the vertices and the edges in a vertical way.



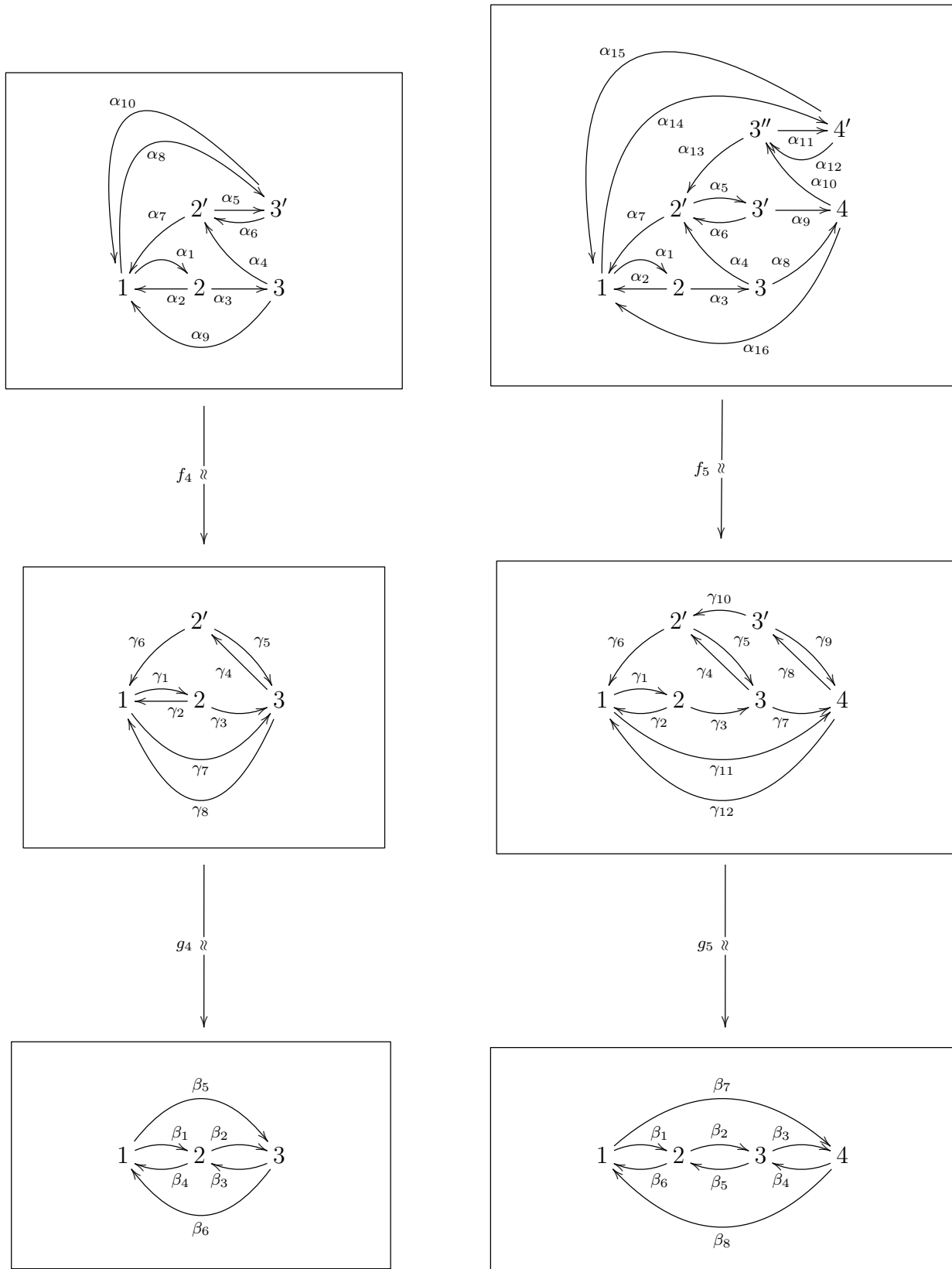
Here, $2V_{f_1} = 2$ and $2'V_{f_1} = 2'$. Moreover, $2V_{f_2} = 2$, $2'V_{f_2} = 2'$, $3V_{f_2} = 3$, $3'V_{f_2} = 3$, $3''V_{f_2} = 3'$.

Varying the target graph and adjusting the other graphs accordingly, we get the following example.



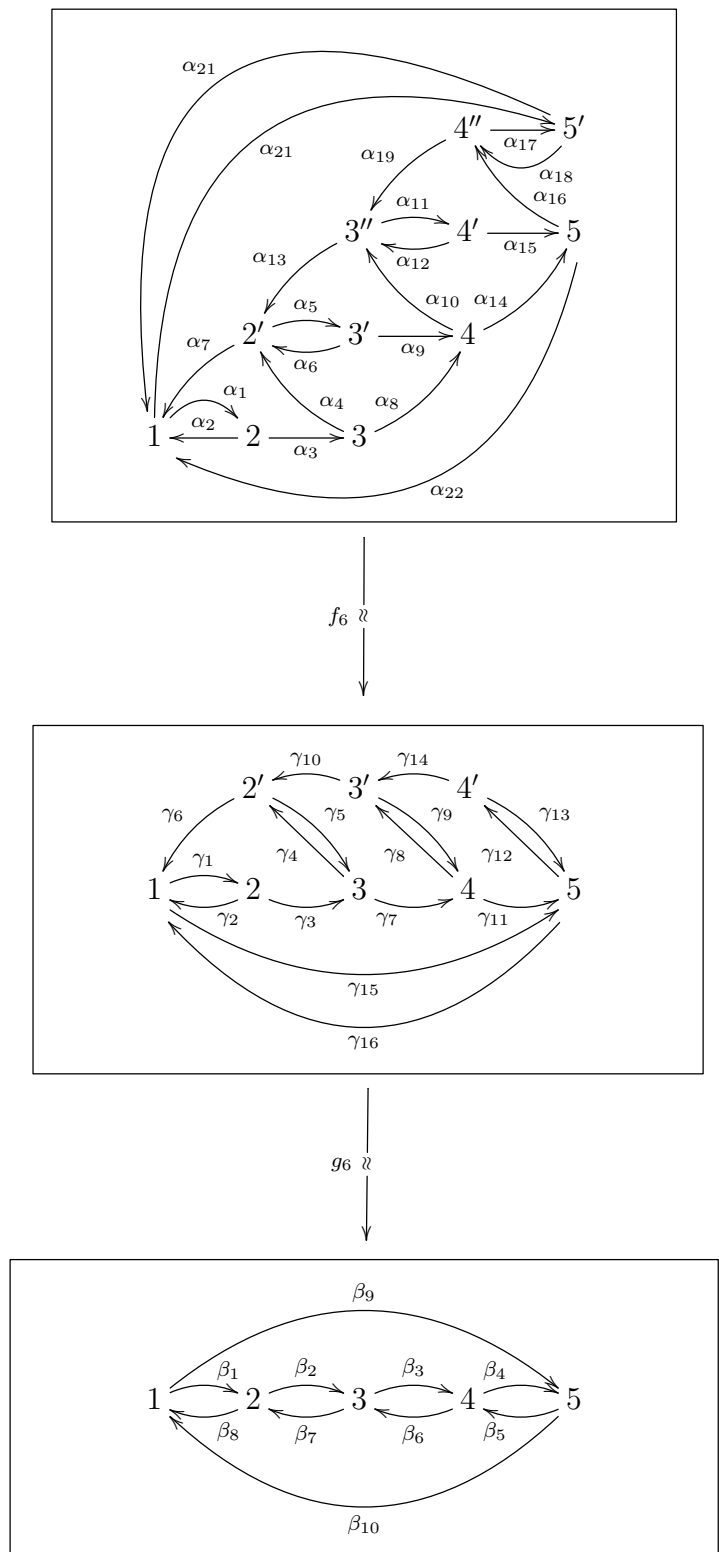
Here, $2V_{f_3} = 2$, $2'V_{f_3} = 2'$, $3V_{f_3} = 3$, $3'V_{f_3} = 3$, $3''V_{f_3} = 3'$, $4V_{f_3} = 4$, $4'V_{f_3} = 4$, $4''V_{f_3} = 4'$.

In the target graphs of the first two examples, we add two edges and adjust the other two graphs to get quasiisomorphisms as follows.



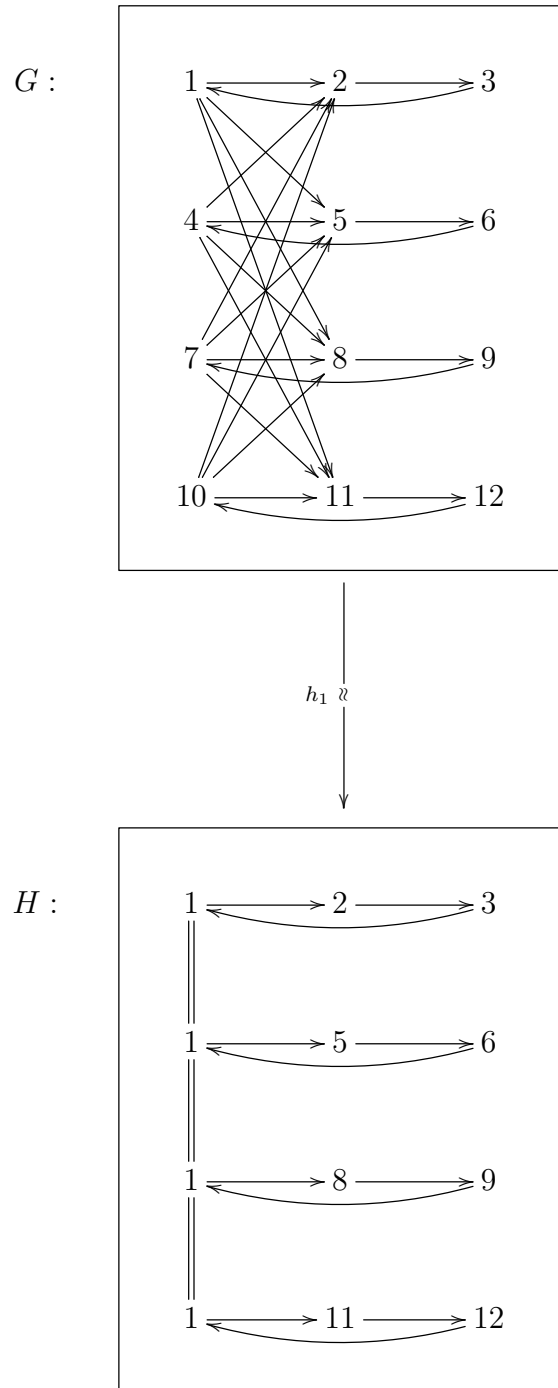
Here, $2V_{f_4} = 2$, $2'V_{f_4} = 2'$. Moreover $2V_{f_5} = 2$, $2'V_{f_5} = 2'$, $3V_{f_5} = 3$, $3'V_{f_5} = 3$, $3''V_{f_5} = 3'$.

We can enlarge this example further to obtain the following quasiisomorphisms.



Here, $2V_{f_6} = 2$, $2'V_{f_6} = 2'$, $3V_{f_6} = 3$, $3'V_{f_6} = 3$, $3''V_{f_6} = 3'$, $4V_{f_6} = 4$, $4'V_{f_6} = 4$, $4''V_{f_6} = 4'$.

In the graph G in the following example we need to have edges from the vertices 1, 4, 7 and 10 to the vertices 2, 5, 8 and 11 to obtain an étale fibration. Since in addition the edges $2 \rightarrow 3$, $5 \rightarrow 6$, $8 \rightarrow 9$ and $11 \rightarrow 12$ of H are unitargeting, we conclude that h_1 is a quasiisomorphism; cf. §0.5.



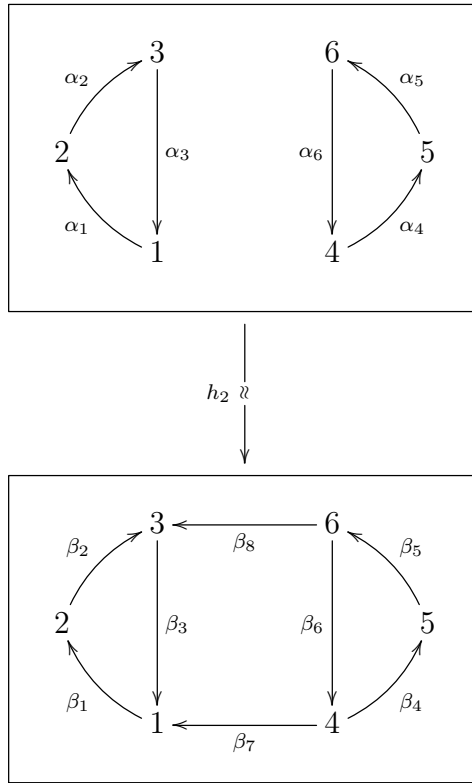
We let

$$\begin{aligned}
 1V_{h_1} &:= 1 & 2V_{h_1} &:= 2 & 3V_{h_1} &:= 3 & 4V_{h_1} &:= 1 & 5V_{h_1} &:= 5 & 6V_{h_1} &:= 6 \\
 7V_{h_1} &:= 1 & 8V_{h_1} &:= 8 & 9V_{h_1} &:= 9 & 10V_{h_1} &:= 1 & 11V_{h_1} &:= 11 & 12V_{h_1} &:= 12 .
 \end{aligned}$$

Here, the graph H has only a single vertex named 1, displayed four times for sake of clarity.

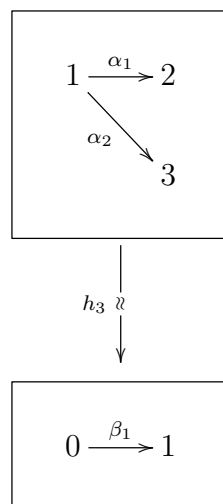
The following graph morphism between fibrant graphs is neither an acyclic fibration nor an acyclic cofibration but nevertheless a quasiisomorphism; cf. Example 226.

To prove this, we can not apply Proposition 210.



Here, $1 V_{h_2} = 1$, $2 V_{h_2} = 2$, $3 V_{h_2} = 3$, $4 V_{h_2} = 4$, $5 V_{h_2} = 5$, $6 V_{h_2} = 6$.

The following graph morphism is an acyclic fibration but not an etale fibration.



So even for a fibration, the sufficient condition of Proposition 210 is not necessary for it to be a quasiisomorphism.

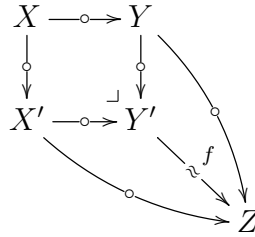
0.6.2 Counterexamples

We show by an example that the set of quasiisomorphisms is not stable under pushouts along cofibrations; cf. Assertion 251.

We show by an example that the pushout of two cofibrant graphs is not necessarily cofibrant; cf. Assertion 255.

We show by an example that the set of cofibrations is not stable under pullbacks; cf. Assertion 253.

We give an example of a pushout and an induced morphism as follows, in which the induced morphism f is not an acyclic cofibration, only a quasiisomorphism; cf. Assertion 254.



We give an example $f : G \rightarrow H$ for which $(C_1, f)_{\text{Gph}}$ and $(C_2, f)_{\text{Gph}}$ are bijective, for which there exists no injective graph morphism $C_k \rightarrow G$ or $C_k \rightarrow H$ for $k \geq 3$, but which is not a quasiisomorphism; cf. Assertion 258.

Conventions

- (1) Given $a, b \in \mathbb{Z}$, we write $[a, b] := \{z \in \mathbb{Z} : a \leq z \leq b\}$. In particular, $[a, b] = \emptyset$ if $a > b$.
- (2) Given $a \in \mathbb{Z}$, we write $\mathbb{Z}_{\geq a} := \{z \in \mathbb{Z} : a \leq z\}$, etc.
- (3) We set $\mathbb{N} := \mathbb{Z}_{\geq 1}$.
- (4) We compose on the right. So given maps $X \xrightarrow{f} Y \xrightarrow{g} Z$, their composite is denoted by $X \xrightarrow{fg=f \cdot g} Z$ and maps $x \in X$ to $x(f \cdot g) = (xf)g$.
- (5) Given a set X , “for $x \in X$ ” means “for all $x \in X$ ”.
- (6) Given a finite set X , we denote by $|X|$ its cardinality.
- (7) Given sets X and Y , the set of maps from X to Y is denoted by Y^X .
- (8) Given a set M , a relation $R \subseteq M \times M$, the equivalence relation (\sim) generated by R and an element $m \in M$, we write $[m]_{(\sim)}$ for the equivalence class of m , i.e. the set of all elements $n \in M$ with $(m, n) \in R$.
- (9) The symbol `%%` in a comment in Magma code refers to a function that is used in this line.
- (10) The label **Reminder** indicates a summary of notations or definitions we remind of.
- (11) The label **Assertion** indicates an assertion that we **falsify** by a counterexample.
- (12) Whenever necessary, we restrict the consideration to a given universe in the sense of Bourbaki [1].

Chapter 1

Preliminaries

1.1 Preliminaries on categories

Let \mathcal{C} be a category.

1.1.1 The properties (2 of 6) and (2 of 3)

Definition 1 Suppose given a subset $Q \subseteq \text{Mor}(\mathcal{C})$.

(1) We say that Q *satisfies* (2 of 6) if the following property holds.

Suppose given a commutative diagram

$$\begin{array}{ccc} & X' & \xrightarrow{gh} & Y' \\ & \nearrow f & \searrow g & \nearrow h \\ X & \xrightarrow{fg} & Y & \end{array}$$

in \mathcal{C} .

Then the composites $X \xrightarrow{fg} Y$ and $X' \xrightarrow{gh} Y'$ are in Q if and only if f, g and h are in Q .

(2) We say that Q *satisfies* (2 of 3) if the following property holds.

Suppose given a commutative diagram

$$\begin{array}{ccc} & Y & \\ & \nearrow f & \searrow g \\ X & \xrightarrow{fg} & Z \end{array}$$

in \mathcal{C} .

Then (i, ii, iii) hold.

- (i) If $f, g \in Q$ then $fg \in Q$.
- (ii) If $f, fg \in Q$ then $g \in Q$.
- (iii) If $g, fg \in Q$ then $f \in Q$.

Lemma 2 Suppose given a subset $Q \subseteq \text{Mor}(\mathcal{C})$ such that $\text{id}_X \in Q$ for $X \in \text{Ob}(\mathcal{C})$.

If Q satisfies (2 of 6), then Q satisfies (2 of 3).

Proof. Suppose given a commutative diagram

$$\begin{array}{ccc} & Y & \\ f \nearrow & & \searrow g \\ X & \xrightarrow{fg} & Z \end{array}$$

in \mathcal{C} . We have to show that (i, ii, iii) from Definition 1 hold.

Ad (i). Suppose that $f, g \in Q$. We have to show that $fg \in Q$.

We have the following commutative diagram.

$$\begin{array}{ccccc} & & Y & \xrightarrow{g} & Z \\ & f \nearrow & & \searrow g & \\ X & \xrightarrow{fg} & Z & \nearrow \text{id}_Z & \\ & & & & \end{array}$$

Since $f, g, \text{id}_Z \in Q$, the morphism $fg : X \rightarrow Z$ is an element of Q by (2 of 6); cf. Definition 1.(1).

Ad (ii). Suppose that $f, fg \in Q$. We have to show that $g \in Q$.

We have the following commutative diagram.

$$\begin{array}{ccccc} & & X & \xrightarrow{fg} & Z \\ & \text{id}_X \nearrow & & \searrow f & \\ X & \xrightarrow{fg} & Y & \nearrow g & \\ & & & & \end{array}$$

Since $f, fg \in Q$, the morphism $g : Y \rightarrow Z$ is an element of Q by (2 of 6); cf. Definition 1.(1).

Ad (iii). Suppose that $g, fg \in Q$. We have to show that $f \in Q$.

We have the following commutative diagram.

$$\begin{array}{ccccc} & & Y & \xrightarrow{g} & Z \\ & f \nearrow & & \searrow g & \\ X & \xrightarrow{fg} & Z & \nearrow \text{id}_Z & \\ & & & & \end{array}$$

Since $g, fg \in Q$, the morphism $f : X \rightarrow Y$ is an element of Q by (2 of 6); cf. Definition 1.(1). \square

Remark 3 The subset $\text{Iso}(\mathcal{C}) \subseteq \text{Mor}(\mathcal{C})$ satisfies (2 of 6).

Proof. Suppose given the following commutative diagram in \mathcal{C} .

$$\begin{array}{ccc} & X' & \xrightarrow{gh} Y' \\ f \nearrow & & \searrow g \\ X & \xrightarrow{fg} & Y \\ & & \nearrow h \end{array}$$

Suppose that $f, g, h \in \text{Iso}(\mathcal{C})$.

Then $fg, gh \in \text{Iso}(\mathcal{C})$.

Conversely, suppose that $fg, gh \in \text{Iso}(\mathcal{C})$.

Then $((fg)^{-1} \cdot f) \cdot g = \text{id}_Y$ and $g \cdot (h \cdot (gh)^{-1}) = \text{id}_{X'}$.

Thus $g \cdot ((fg)^{-1} \cdot f) = g \cdot ((fg)^{-1} \cdot f) \cdot g \cdot (h \cdot (gh)^{-1}) = g \cdot (h \cdot (gh)^{-1}) = \text{id}_{X'}$.

Therefore, g is an isomorphism with $g^{-1} = (fg)^{-1} \cdot f$.

Hence $f = (fg) \cdot g^{-1}$ and $h = g^{-1} \cdot (gh)$ are isomorphisms. \square

Remark 4 Suppose given a subset $Q \subseteq \text{Mor}(\mathcal{C})$ such that $\text{id}_X \in Q$ for $X \in \text{Ob}(\mathcal{C})$.

Suppose that Q satisfies (2 of 6).

Then $\text{Iso}(\mathcal{C}) \subseteq Q$.

Proof. Let $g : X \rightarrow Y$ be an isomorphism in \mathcal{C} .

We have the following commutative diagram in \mathcal{C} .

$$\begin{array}{ccc} & X & \xrightarrow{\text{id}_X} X \\ g^{-1} \nearrow & & \searrow g \\ Y & \xrightarrow{\text{id}_Y} & Y \\ & & \nearrow g^{-1} \end{array}$$

Since $\text{id}_X, \text{id}_Y \in Q$, we have $g \in Q$ by (2 of 6); cf. Definition 1.(1). \square

1.1.2 Pushout and Pullback

Definition 5 Suppose given a quadrangle

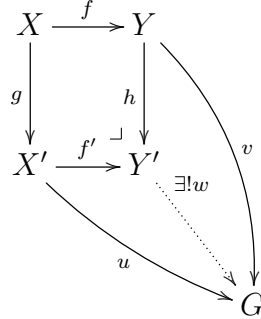
$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in \mathcal{C} .

It is called a *pushout* if (Pushout 1–2) hold.

(Pushout 1) We have $g \cdot f' = f \cdot h$.

(Pushout 2) Suppose given an object $G \in \text{Ob}(\mathcal{C})$ and morphisms $u : X' \rightarrow G$ and $v : Y \rightarrow G$ in $\text{Mor}(\mathcal{C})$ such that $f \cdot v = g \cdot u$. Then there exists a unique morphism $w : Y' \rightarrow G$ in $\text{Mor}(\mathcal{C})$ such that $f' \cdot w = u$ and $h \cdot w = v$.

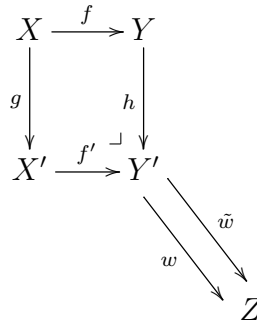


To indicate that this quadrangle is a pushout, we write

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array} .$$

Then we also say that f' is a pushout of f .

Remark 6 Suppose given



in \mathcal{C} such that $f'w = f'\tilde{w}$ and $hw = h\tilde{w}$.

Then we have $w = \tilde{w}$.

Proof. Let $v := hw = h\tilde{w}$ and $u := f'w = f'\tilde{w}$.

We have $fv = fhv = gf'w = gu$.

Thus there exists exactly one morphism $w' : Y' \rightarrow Z$ such that $hw' = v$ and $f'w' = u$.

But we have $hw = v$ and $f'w = u$, and $h\tilde{w} = v$ and $f'\tilde{w} = u$.

So $w = w' = \tilde{w}$.

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow h \\
 X' & \xrightarrow{f'} & Y' \\
 & \searrow u & \downarrow w \\
 & & Z
 \end{array}
 \begin{array}{l}
 \swarrow v \\
 \downarrow \tilde{w}
 \end{array}$$

□

Remark 7 Suppose given

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

and

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow \tilde{h} \\
 X' & \xrightarrow{\tilde{f}'} & \tilde{Y}'
 \end{array}$$

in \mathcal{C} .

Then there exists an isomorphism $\tilde{w} : \tilde{Y}' \xrightarrow{\sim} Y'$ such that $\tilde{f}'\tilde{w} = f'$ and $\tilde{h}\tilde{w} = h$.

Proof. Because Y' is a pushout there exists, by (Pushout 2), a unique morphism $w : Y' \rightarrow \tilde{Y}'$ such that $hw = \tilde{h}$ and $f'w = \tilde{f}'$.

Because \tilde{Y}' is a pushout there exists, by (Pushout 2), a unique morphism $\tilde{w} : \tilde{Y}' \rightarrow Y'$ such that $\tilde{f}'\tilde{w} = f'$ and $\tilde{h}\tilde{w} = h$.

We have

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow h \\
 X' & \xrightarrow{f'} & Y' \\
 & \searrow \tilde{f}' & \downarrow w \\
 & & \tilde{Y}'
 \end{array}
 \begin{array}{l}
 \swarrow \tilde{h} \\
 \downarrow \tilde{w}
 \end{array}$$

We have to show that $w\tilde{w} = \text{id}_{Y'}$ and $\tilde{w}w = \text{id}_{\tilde{Y}'}$.

We have $hw\tilde{w} = \tilde{h}\tilde{w} = h$ and $f'w\tilde{w} = \tilde{f}'\tilde{w} = f'$. So we have $w\tilde{w} = \text{id}_{Y'}$; cf. Remark 6.

We have $\tilde{h}\tilde{w}w = hw = \tilde{h}$ and $\tilde{f}'\tilde{w}w = f'w = \tilde{f}'$. So we have $\tilde{w}w = \text{id}_{\tilde{Y}'}$; cf. Remark 6. □

Remark 8 Suppose given

$$\begin{array}{ccccc} X & \xrightarrow{f} & Y & \xrightarrow{g} & Z \\ u \downarrow & & v \downarrow & & w \downarrow \\ X' & \xrightarrow{f'} & Y' & \xrightarrow{g'} & Z' \end{array}$$

in \mathcal{C} . Then

$$\begin{array}{ccc} X & \xrightarrow{fg} & Z \\ u \downarrow & & \downarrow w \\ X' & \xrightarrow{f'g'} & Z' \end{array}$$

Proof.

Ad (Pushout 1).

We have to show that $u(f'g') \stackrel{!}{=} (fg)w$.

We have $uf'g' = fvg' = fgw$; cf. Definition 5.

Ad (Pushout 2).

Suppose given an object $G \in \text{Ob}(\mathcal{C})$ and morphisms $a : X' \rightarrow G$ and $b : Z \rightarrow G$ such that $u \cdot a = (fg) \cdot b$. We have to show that there exists exactly one morphism $d : Z' \rightarrow G$ such that $(f'g') \cdot d = a$ and $w \cdot d = b$. Because Y' is a pushout there exists exactly one morphism $c : Y' \rightarrow G$ such that $f' \cdot c = a$ and $v \cdot c = g \cdot b$. Because Z' is a pushout and because of $v \cdot c = g \cdot b$ there exists exactly one morphism $d : Z' \rightarrow G$ such that $g' \cdot d = c$ and $w \cdot d = b$. So there exists a morphism $d : Z' \rightarrow G$ such that $f'g' \cdot d = f'c = a$ and $w \cdot d = b$.

$$\begin{array}{ccccc} X & \xrightarrow{f} & Y & \xrightarrow{g} & Z \\ u \downarrow & & v \downarrow & & w \downarrow \\ X' & \xrightarrow{f'} & Y' & \xrightarrow{g'} & Z' \end{array} \begin{array}{l} \searrow b \\ \searrow \exists! d \\ \searrow \exists! c \\ \searrow a \end{array}$$

It remains to show uniqueness.

Let $\tilde{d} : Z' \rightarrow G$ in $\text{Mor}(\mathcal{C})$ be a morphism such that $a = f'g'\tilde{d}$ and such that $b = w\tilde{d}$.

We have to show that $d \stackrel{!}{=} \tilde{d}$.

Recall that $c : Y' \rightarrow G$ is the unique morphism with $f' \cdot c = a$ and $v \cdot c = g \cdot b$.

We have $f' \cdot (g'\tilde{d}) = a$ and $v \cdot (g'\tilde{d}) = gw\tilde{d} = gb$. Because of the uniqueness of the morphism c we get $g'\tilde{d} = c$.

Recall that $d : Z' \rightarrow G$ is the unique morphism with $g'd = c$ and $wd = b$.

But $g'\tilde{d} = c$ and $w\tilde{d} = b$. So $d = \tilde{d}$. □

Definition 9 Suppose given a quadrangle

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in \mathcal{C} .

It is called a *pullback* if (Pullback 1–2) hold.

(Pullback 1) We have $f \cdot h = g \cdot f'$.

(Pullback 2) Suppose given an object $G \in \text{Ob}(\mathcal{C})$ and morphisms $u : G \rightarrow Y$ and $v : G \rightarrow X'$ in $\text{Mor}(\mathcal{C})$ such that $u \cdot h = v \cdot f'$. Then there exists a unique morphism $w : G \rightarrow X$ in $\text{Mor}(\mathcal{C})$ such that $w \cdot f = u$ and $w \cdot g = v$.

$$\begin{array}{ccccc} G & & & & \\ & \searrow u & & & \\ & & X & \xrightarrow{f} & Y \\ & \swarrow \exists! w & \downarrow g & & \downarrow h \\ & & X' & \xrightarrow{f'} & Y' \\ & \swarrow v & & & \end{array}$$

To indicate that this quadrangle is a pullback, we write

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & \lrcorner & \downarrow h \\ X' & \xrightarrow{f'} & Y' . \end{array}$$

Then we also say that f is a pullback of f' .

The following remarks on pullbacks have dual counterparts; cf. Remarks 6, 7 and 8. We carry out the proofs nonetheless.

Remark 10 Suppose given

$$\begin{array}{ccccc} Z & & & & \\ & \searrow \tilde{w} & & & \\ & \swarrow w & & & \\ & & X & \xrightarrow{f} & Y \\ & & \downarrow g & & \downarrow h \\ & & X' & \xrightarrow{f'} & Y' \end{array}$$

in \mathcal{C} such that $wg = \tilde{w}g$ and $wf = \tilde{w}f$.

Then we have $w = \tilde{w}$.

Proof. Let $v := wg = \tilde{w}g$ and $u := wf = \tilde{w}f$.

We have $vf' = wgf' = wfh = uh$.

Thus there exists exactly one morphism $w' : Z \rightarrow X$ such that $w'g = v$ and $w'f = u$.

But we have $wg = v$ and $wf = u$, and $\tilde{w}g = v$ and $\tilde{w}f = u$.

So $w = w' = \tilde{w}$.

$$\begin{array}{ccccc}
 Z & & & & \\
 \swarrow & \searrow & & \searrow & \\
 & X & \xrightarrow{f} & Y & \\
 \swarrow & \downarrow g & \lrcorner & \downarrow h & \\
 & X' & \xrightarrow{f'} & Y' & \\
 \swarrow & & & & \\
 & & & &
 \end{array}$$

□

Remark 11 Suppose given

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & \lrcorner & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

and

$$\begin{array}{ccc}
 \tilde{X} & \xrightarrow{\tilde{f}} & Y \\
 \tilde{g} \downarrow & \lrcorner & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

in \mathcal{C} .

Then there exists an isomorphism $\tilde{w} : X \xrightarrow{\sim} \tilde{X}$ such that $\tilde{w}\tilde{f} = f$ and $\tilde{w}\tilde{g} = g$.

Proof. Because X is a pushout there exists, by (Pushout 2), a unique morphism $w : \tilde{X} \rightarrow X$ such that $wf = \tilde{f}$ and $wg = \tilde{g}$.

Because \tilde{X} is a pushout there exists, by (Pushout 2), a unique morphism $\tilde{w} : X \rightarrow \tilde{X}$ such that $\tilde{w}\tilde{f} = f$ and $\tilde{w}\tilde{g} = g$.

We have

$$\begin{array}{ccccc}
 \tilde{X} & & & & \\
 \swarrow \tilde{w} & & & & \searrow \tilde{f} \\
 & X & \xrightarrow{f} & Y & \\
 \downarrow w & \lrcorner & & \lrcorner & \downarrow h \\
 & X & \xrightarrow{f} & Y & \\
 \downarrow \tilde{g} & \downarrow g & & \downarrow h & \\
 & X' & \xrightarrow{f'} & Y' &
 \end{array}$$

We have to show that $\tilde{w}w = \text{id}_X$ and $w\tilde{w} = \text{id}_{\tilde{X}}$.

We have $w\tilde{w}\tilde{f} = wf = \tilde{f}$ and $w\tilde{w}\tilde{g} = wg = \tilde{g}$. So we have $w\tilde{w} = \text{id}_{\tilde{X}}$; cf. Remark 10.

We have $\tilde{w}wf = \tilde{w}\tilde{f} = f$ and $\tilde{w}wg = \tilde{w}\tilde{g} = g$. So we have $\tilde{w}w = \text{id}_X$; cf. Remark 10. \square

Remark 12 Suppose given

$$\begin{array}{ccccc}
 X & \xrightarrow{f} & Y & \xrightarrow{g} & Z \\
 \downarrow u & \lrcorner & \downarrow v & \lrcorner & \downarrow w \\
 X' & \xrightarrow{f'} & Y' & \xrightarrow{g'} & Z'
 \end{array}$$

in \mathcal{C} . Then

$$\begin{array}{ccc}
 X & \xrightarrow{fg} & Z \\
 \downarrow u & \lrcorner & \downarrow w \\
 X' & \xrightarrow{f'g'} & Z' .
 \end{array}$$

Proof.

Ad (Pullback 1).

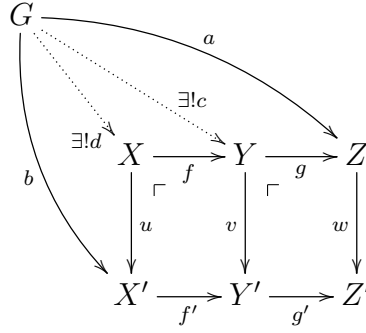
We have to show that $(fg)w \stackrel{!}{=} u(f'g')$.

We have $fgw = fvg' = u f'g'$; cf. Definition 9.

Ad (Pullback 2).

Suppose given an object $G \in \text{Ob}(\mathcal{C})$ and morphisms $a : G \rightarrow Z$ and $b : G \rightarrow X'$ such that $a \cdot w = b \cdot (f'g')$. We have to show that there exists exactly one morphism $d : G \rightarrow X$ such that $d \cdot (fg) = a$ and $d \cdot u = b$. Because Y is a pullback there exist exactly one morphism $c : G \rightarrow Y$ such that $c \cdot g = a$ and $c \cdot v = b \cdot f'$. Because X is a pullback and because of $c \cdot v = b \cdot f'$ there exists exactly one morphism $d : G \rightarrow X$ such that $d \cdot u = b$ and $d \cdot f = c$. So there exists a

morphism $d : G \rightarrow X$ such that $d \cdot u = b$ and $d \cdot fg = cg = a$.



It remains to show uniqueness.

Let $\tilde{d} : G \rightarrow X$ in $\text{Mor}(\mathcal{C})$ be a morphism such that $a = \tilde{d}fg$ and such that $b = \tilde{d}u$.

We have to show that $d \stackrel{!}{=} \tilde{d}$.

Recall that $c : G \rightarrow Y$ is the unique morphism with $c \cdot g = a$ and $c \cdot v = b \cdot f'$.

We have $(\tilde{d}f) \cdot g = a$ and $\tilde{d} \cdot u = b$. Because of the uniqueness of the morphism c we get $\tilde{d}f = c$.

Recall that $d : G \rightarrow X$ is the unique morphism with $df = c$ and $du = b$.

But $\tilde{d}f = c$ and $\tilde{d}u = b$. So $d = \tilde{d}$. \square

1.1.3 Lifting properties

Let \mathcal{C} be a category.

Definition 13 Suppose given a set of morphisms $M \subseteq \text{Mor}(\mathcal{C})$.

(1) For a morphism $f : X \rightarrow Y$ we define the *left lifting property* (LLP_M) as follows.

(LLP_M) For each morphism $f' : X' \rightarrow Y'$ in M and all morphisms $u : X \rightarrow X'$ and $v : Y \rightarrow Y'$ in $\text{Mor}(\mathcal{C})$ with $fv = uf'$, there exists a morphism $h : Y \rightarrow X'$ such that $fh = u$ and $hf' = v$.

$$\begin{array}{ccc} X & \xrightarrow{u} & X' \\ f \downarrow & \nearrow h & \downarrow f' \\ Y & \xrightarrow{v} & Y' \end{array}$$

The morphism f may or may not satisfy (LLP_M).

(2) Let the *left-liftable set* $\square M$ of M be defined as follows.

$$\square M := \{f \in \text{Mor}(\mathcal{C}) : f \text{ satisfies } (\text{LLP}_M)\}$$

For short, $\square M$ is also called M *left-lift*.

Definition 14 Suppose given a set of morphisms $M \subseteq \text{Mor}(\mathcal{C})$.

(1) For a morphism $f : X \rightarrow Y$ we define the *right lifting property* (RLP_M) as follows.

(RLP_M) For each morphism $f' : X' \rightarrow Y'$ in M and all morphisms $u : X' \rightarrow X$ and $v : Y' \rightarrow Y$ in $\text{Mor}(\mathcal{C})$ with $uf = f'v$, there exists a morphism $h : Y' \rightarrow X$ such that $f'h = u$ and $hf = v$.

$$\begin{array}{ccc} X' & \xrightarrow{u} & X \\ f' \downarrow & \nearrow h & \downarrow f \\ Y' & \xrightarrow{v} & Y \end{array}$$

The morphism f may or may not satisfy (RLP_M).

(2) Let the *right-liftable set* M^\square of M be defined as follows.

$$M^\square := \{f \in \text{Mor}(\mathcal{C}) : f \text{ satisfies } (\text{RLP}_M)\}$$

For short, the set M^\square is also called M *right-lift*.

Definition 15 Suppose given subsets $M, N \subseteq \text{Mor}(\mathcal{C})$.

We write $M \square N$ if $M \subseteq \square N$, or, equivalently, if $M^\square \supseteq N$; cf. Definitions 13 and 14.

If $M \square N$, we say M *lift* N .

Here, we use “lift” as a preposition.

Remark 16 Suppose given subsets $M \subseteq N \subseteq \text{Mor}(\mathcal{C})$.

Then we have $M^\square \supseteq N^\square$ and we have $\square M \supseteq \square N$; cf. Definitions 13 and 14.

Remark 17 Suppose given a set of morphisms $M \subseteq \text{Mor}(\mathcal{C})$.

Then we have $M \subseteq (\square M)^\square$ and $M \subseteq \square(M^\square)$.

Proof.

Since we have $\square M \subseteq \square M$ we have $\square M \square M$ and thus $M \subseteq (\square M)^\square$; cf. Definition 15.

Since we have $M^\square \subseteq M^\square$ we have $M \square M^\square$ and thus $M \subseteq \square(M^\square)$; cf. Definition 15. \square

Note that in general, we do **not** have $(\square M)^\square \subseteq M$. For instance, if \mathcal{C} has an object X , then $\emptyset \not\square \{\text{id}_X\} \subseteq \text{Mor}(\mathcal{C})^\square = (\square \emptyset)^\square$.

Note that in general, we do **not** have $\square(M^\square) \subseteq M$. For instance, if \mathcal{C} has an object X , then $\emptyset \not\square \{\text{id}_X\} \subseteq \square \text{Mor}(\mathcal{C}) = \square(\emptyset^\square)$.

Remark 18 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

We have $\text{Iso}(\mathcal{C}) \subseteq \square M$ and $\text{Iso}(\mathcal{C}) \subseteq M^\square$.

Proof.

Suppose given a commutative diagram

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ \wr \downarrow f & & \downarrow m \\ Y & \xrightarrow{b} & Y' \end{array}$$

in \mathcal{C} with $f \in \text{Iso}(\mathcal{C})$ and $m \in M \subseteq \text{Mor}(\mathcal{C})$.

We have the morphism $f^{-1}a : Y \rightarrow X'$ with $ff^{-1}a = a$ and with $f^{-1}am = f^{-1}fb = b$.

So we have the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ \wr \downarrow f & \nearrow f^{-1}a & \downarrow m \\ Y & \xrightarrow{b} & Y' \end{array}$$

And so the isomorphism $f : X \rightarrow Y$ is in $\square M$.

So we have $\text{Iso}(\mathcal{C}) \subseteq \square M$.

Now suppose given a commutative diagram

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ m \downarrow & & \wr \downarrow f \\ Y & \xrightarrow{b} & Y' \end{array}$$

in \mathcal{C} with $f \in \text{Iso}(\mathcal{C})$ and $m \in M \subseteq \text{Mor}(\mathcal{C})$.

We have the morphism $bf^{-1} : Y \rightarrow X'$ with $mbf^{-1} = af f^{-1} = a$ and with $bf^{-1}f = b$.

So we have the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ m \downarrow & \nearrow bf^{-1} & \wr \downarrow f \\ Y & \xrightarrow{b} & Y' \end{array}$$

And so the isomorphism $f : X \rightarrow Y$ is in M^\square .

So we have $\text{Iso}(\mathcal{C}) \subseteq M^\square$. □

Remark 19 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

Suppose given morphisms $f : X \rightarrow Y$ and $g : Y \rightarrow Z$ in $\square M$.

Then the composite $fg : X \rightarrow Z$ is also in $\square M$.

Proof. Suppose given a commutative diagram as follows, where $m \in M$.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ f \downarrow & & \downarrow m \\ Y & & \\ g \downarrow & & \downarrow \\ Z & \xrightarrow{b} & Z' \end{array}$$

We have to show that there exists a morphism $h : Z \rightarrow X'$ such that $(fg)h \stackrel{!}{=} a$ and $hm \stackrel{!}{=} b$.

Since $f(gb) = am$ and since the morphism $f : X \rightarrow Y$ is in ${}^{\square}M$, there exists a morphism $k : Y \rightarrow X'$ such that $fk = a$ and $km = gb$.

Since $km = gb$ and since the morphism $g : Y \rightarrow Z$ is in ${}^{\square}M$, there exists a morphism $h : Z \rightarrow X'$ such that $gh = k$ and $hm = b$.

So we have $(fg)h = fk = a$ and $hm = b$.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ f \downarrow & \nearrow k & \downarrow m \\ Y & & \\ g \downarrow & \nearrow h & \downarrow \\ Z & \xrightarrow{b} & Z' \end{array}$$

□

Remark 20 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

Suppose given morphisms $f : X \rightarrow Y$ and $g : Y \rightarrow Z$ in M^{\square} .

Then the composite $fg : X \rightarrow Z$ is also in M^{\square} .

Proof. Suppose given a commutative diagram as follows, where $m \in M$.

$$\begin{array}{ccc} X' & \xrightarrow{a} & X \\ m \downarrow & & \downarrow f \\ Z' & \xrightarrow{b} & Z \\ & & \downarrow g \\ & & Y \end{array}$$

We have to show that there exists a morphism $h : Z' \rightarrow X$ such that $h(fg) \stackrel{!}{=} b$ and $mh \stackrel{!}{=} a$.

Since $(af)g = mb$ and since the morphism $g : Y \rightarrow Z$ is in M^{\square} , there exists a morphism $k : Z' \rightarrow Y$ such that $kg = b$ and $mk = af$.

Since $mk = af$ and since the morphism $f : X \rightarrow Y$ is in M^{\square} , there exists a morphism $h : Z' \rightarrow X$ such that $hf = k$ and $mh = a$.

So we have $h(fg) = kg = b$ and $mh = a$.

$$\begin{array}{ccc}
 X' & \xrightarrow{a} & X \\
 \downarrow m & \nearrow h & \downarrow f \\
 & & Y \\
 & \nearrow k & \downarrow g \\
 Z' & \xrightarrow{b} & Z
 \end{array}$$

□

Remark 21 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

We have $\square M \subseteq \text{Mor}(\mathcal{C})$; cf. Definition 13.

The set $\square M$ is stable under pushouts; cf. Definition 82.

Proof. Suppose given a pushout

$$\begin{array}{ccc}
 X & \xrightarrow{h} & Y \\
 f \downarrow & & \downarrow g \\
 X' & \xrightarrow{h'} & Y'
 \end{array}
 \quad \lrcorner$$

in \mathcal{C} with $f : X \rightarrow X'$ in $\square M$.

We have to show that the morphism $g : Y \rightarrow Y'$ is in $\square M$, i.e. that the morphism g satisfies (LLP_M) ; cf. Definition 13.

Suppose given a commutative diagram as follows, where $m \in M$.

$$\begin{array}{ccccc}
 X & \xrightarrow{h} & Y & \xrightarrow{p} & Z \\
 f \downarrow & & \downarrow g & & \downarrow m \\
 X' & \xrightarrow{h'} & Y' & \xrightarrow{q} & Z'
 \end{array}
 \quad \lrcorner$$

We have to show that there exists a morphism $\tilde{k} : Y' \rightarrow Z$ such that $g\tilde{k} = p$ and $\tilde{k}m = q$.

Because the morphism $f : X \rightarrow X'$ is in $\square M$ there exists a morphism $k : X' \rightarrow Z$ such that $fk = hp$ and $km = h'q$.

Because Y' is a pushout and $fk = hp$, there exists a morphism $\tilde{k} : Y' \rightarrow Z$ such that $h'\tilde{k} = k$ and $g\tilde{k} = p$.

So we have $g\tilde{k} = p$. It remains to show that $\tilde{k}m \stackrel{!}{=} q$.

We have $h'\tilde{k}m = km = h'q$ and $g\tilde{k}m = pm = gq$.

Cancelling h' and g simultaneously using Remark 6, we obtain $\tilde{k}m = q$.

$$\begin{array}{ccccc}
 X & \xrightarrow{h} & Y & \xrightarrow{p} & Z \\
 f \downarrow & & \downarrow g & & \downarrow m \\
 X' & \xrightarrow{h'} & Y' & \xrightarrow{q} & Z'
 \end{array}
 \quad \lrcorner$$

$\begin{array}{ccc}
 & \nearrow k & \\
 & \searrow \tilde{k} & \\
 & &
 \end{array}$

□

Remark 22 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

We have $M^\square \subseteq \text{Mor}(\mathcal{C})$; cf. Definition 13.

The set M^\square is stable under pullbacks; cf. Definition 96.

Proof. Suppose given a pullback

$$\begin{array}{ccc} Y & \xrightarrow{f} & Z \\ g \downarrow & \lrcorner & \downarrow h \\ Y' & \xrightarrow{f'} & Z' \end{array}$$

in \mathcal{C} with $h : Z \rightarrow Z'$ in M^\square .

We have to show that the morphism $g : Y \rightarrow Y'$ is in M^\square , i.e. that the morphism g satisfies (RLP_M) ; cf. Definition 14.

Suppose given a commutative diagram as follows, where $m \in M$.

$$\begin{array}{ccccc} X & \xrightarrow{p} & Y & \xrightarrow{f} & Z \\ m \downarrow & \lrcorner & \downarrow g & \lrcorner & \downarrow h \\ X' & \xrightarrow{q} & Y' & \xrightarrow{f'} & Z' \end{array}$$

We have to show that there exists a morphism $\tilde{k} : X' \rightarrow Z$ such that $m\tilde{k} = p$ and $\tilde{k}g = q$.

Because the morphism $h : Z \rightarrow Z'$ is in M^\square there exists a morphism $k : X' \rightarrow Z$ such that $mk = pf$ and $kh = qf'$.

Because Y is a pullback and $kh = pf'$, there exists a morphism $\tilde{k} : X' \rightarrow Y$ such that $\tilde{k}f = k$ and $\tilde{k}g = q$.

So we have $\tilde{k}g = q$. It remains to show that $m\tilde{k} \stackrel{!}{=} p$.

We have $m\tilde{k}f = mk = pf$ and $m\tilde{k}g = mq = pg$.

Cancelling f and g simultaneously using Remark 10, we obtain $m\tilde{k} = p$.

$$\begin{array}{ccccc} X & \xrightarrow{a} & Y & \xrightarrow{f} & Z \\ m \downarrow & \lrcorner & \downarrow g & \lrcorner & \downarrow h \\ X' & \xrightarrow{q} & Y' & \xrightarrow{f'} & Z' \end{array}$$

$\begin{array}{ccc} \nearrow k & & \nearrow \\ \searrow \tilde{k} & & \searrow \end{array}$

□

1.1.4 Subsets of $\text{Mor}(\mathcal{C})$ being closed under retracts

Definition 23 A subset of morphisms $M \subseteq \text{Mor}(\mathcal{C})$ is called *closed under retracts* if the following property (C) holds.

(C) Suppose given a commutative diagram

$$\begin{array}{ccc} & G' & \xrightarrow{f'} & H' \\ & \uparrow p & & \uparrow q \\ \text{id}_{G'} & \left(\begin{array}{ccc} G & \xrightarrow{f} & H \end{array} \right) & & \text{id}_{H'} \\ & \uparrow i & & \uparrow j \\ & G' & \xrightarrow{f'} & H' \end{array}$$

in \mathcal{C} such that $f : G \rightarrow H$ in M .

Then the morphism $f' : G' \rightarrow H'$ is in M .

Remark 24 The subset of isomorphisms $\text{Iso}(\mathcal{C}) \subseteq \text{Mor}(\mathcal{C})$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in \mathcal{C} as follows.

$$\begin{array}{ccc} & G' & \xrightarrow{f'} & H' \\ & \uparrow p & & \uparrow q \\ \text{id}_{G'} & \left(\begin{array}{ccc} G & \xrightarrow{\sim} & H \end{array} \right) & & \text{id}_{H'} \\ & \uparrow i & & \uparrow j \\ & G' & \xrightarrow{f'} & H' \end{array}$$

We have to show that the morphism $f' : G' \rightarrow H'$ is an isomorphism.

We show that we have $f'^{-1} = j f^{-1} p : H' \rightarrow G'$.

We have $j f^{-1} p \cdot f' = j f^{-1} f q = j q = \text{id}_{H'}$ and $f' \cdot j f^{-1} p = i f f^{-1} p = i p = \text{id}_{G'}$.

So the morphism $f' : G' \rightarrow H'$ is an isomorphism. □

Remark 25 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

The subset $\square M \subseteq \text{Mor}(\mathcal{C})$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in \mathcal{C} as follows, where $f \in \square M$.

$$\begin{array}{ccc} & G' & \xrightarrow{f'} & H' \\ & \uparrow p & & \uparrow q \\ \text{id}_{G'} & \left(\begin{array}{ccc} G & \xrightarrow{f} & H \end{array} \right) & & \text{id}_{H'} \\ & \uparrow i & & \uparrow j \\ & G' & \xrightarrow{f'} & H' \end{array}$$

We have to show that the morphism $f' : G' \rightarrow H'$ is in $\square M$.

Suppose given a commutative diagram in \mathcal{C} as follows, where $m \in M$.

$$\begin{array}{ccc}
 G'' & \xrightarrow{m} & H'' \\
 \uparrow u & & \uparrow v \\
 G' & \xrightarrow{f'} & H' \\
 \uparrow p & & \uparrow q \\
 G & \xrightarrow{f} & H \\
 \uparrow i & & \uparrow j \\
 G' & \xrightarrow{f'} & H'
 \end{array}
 \begin{array}{c}
 \text{id}_{G'} \\
 \left. \begin{array}{c} \uparrow \\ \uparrow \\ \uparrow \end{array} \right\} \\
 \left. \begin{array}{c} \downarrow \\ \downarrow \\ \downarrow \end{array} \right\} \\
 \text{id}_{H'}
 \end{array}$$

So we have to show that there exists a morphism $h : H' \rightarrow G''$ such that $hm = v$ and $f'h = u$; cf. Definition 13.

Since the morphism $f : G \rightarrow H$ is in $\square M$, there exists a morphism $k : H \rightarrow G''$ such that $km = qv$ and $fk = pu$.

We let $h := jk : H' \rightarrow G''$.

We have $hm = jkm = jqv = v$ and we have $f'h = f'jk = ifk = ipu = u$.

So the morphism $f' : G' \rightarrow H'$ is in $\square M$. □

Remark 26 Suppose given a subset $M \subseteq \text{Mor}(\mathcal{C})$.

The subset $M^\square \subseteq \text{Mor}(\mathcal{C})$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in \mathcal{C} as follows, where $f \in M^\square$.

$$\begin{array}{ccc}
 G' & \xrightarrow{f'} & H' \\
 \uparrow p & & \uparrow q \\
 G & \xrightarrow{f} & H \\
 \uparrow i & & \uparrow j \\
 G' & \xrightarrow{f'} & H'
 \end{array}
 \begin{array}{c}
 \text{id}_{G'} \\
 \left. \begin{array}{c} \uparrow \\ \uparrow \end{array} \right\} \\
 \left. \begin{array}{c} \downarrow \\ \downarrow \end{array} \right\} \\
 \text{id}_{H'}
 \end{array}$$

We have to show that the morphism $f' : G' \rightarrow H'$ is in M^\square .

Suppose given a commutative diagram in \mathcal{C} as follows, where $m \in M$.

$$\begin{array}{ccc}
 G' & \xrightarrow{f'} & H' \\
 \uparrow p & & \uparrow q \\
 G & \xrightarrow{f} & H \\
 \uparrow i & & \uparrow j \\
 G' & \xrightarrow{f'} & H' \\
 \uparrow u & & \uparrow v \\
 G'' & \xrightarrow{m} & H''
 \end{array}
 \begin{array}{c}
 \text{id}_{G'} \\
 \left. \begin{array}{c} \uparrow \\ \uparrow \\ \uparrow \end{array} \right\} \\
 \left. \begin{array}{c} \downarrow \\ \downarrow \\ \downarrow \end{array} \right\} \\
 \text{id}_{H'}
 \end{array}$$

We have to show that there exists a morphism $h : H'' \rightarrow G'$ such that $hf' = v$ and $mh = u$; cf. Definition 14.

Since the morphism $f : G \rightarrow H$ is in M^\square , there exists a morphism $k : H'' \rightarrow G$ such that $kf = vj$ and $mk = ui$.

We let $h := kp : H'' \rightarrow G'$.

We have $hf' = kpf' = kfq = vjq = v$ and we have $mh = mkp = uip = u$.

So the morphism $f' : G' \rightarrow H'$ is in M^\square . □

Definition 27 Suppose given subset $M, N \subseteq \text{Mor}(\mathcal{C})$.

We write $M \hat{\square} N$ if the properties (C 1–3) hold.

(C 1) $M = \square N$.

(C 2) $M^\square = N$.

(C 3) For a graph morphism $f : X \rightarrow Z$ there exist graph morphisms $m : X \rightarrow Y$ in M and $n : Y \rightarrow Z$ in N such that

$$\begin{array}{ccc} & Y & \\ m \nearrow & & \searrow n \\ X & \xrightarrow{f} & Z \end{array}$$

commutes.

If $M \hat{\square} N$, we say M *closed-lift* N .

Here, we use “closed-lift” as a preposition.

Remark 28 Suppose given subsets $M, N \subseteq \text{Mor}(\mathcal{C})$ such that $M \square N$.

We have $M \hat{\square} N$ if and only if M and N are closed under retracts and (C 3) holds; cf. Definition 27.

Proof.

First, suppose that $M \hat{\square} N$.

Left-liftable sets are closed under retracts; cf. Remark 25.

Right-liftable sets are closed under retracts; cf. Remark 26.

Second, suppose that M and N are closed under retracts and that (C 3) holds.

We have to show that $M \stackrel{!}{=} \square N$ and that $M^\square \stackrel{!}{=} N$.

Recall that $M \square N$ is equivalent to $M \subseteq \square N$ and to $M^\square \supseteq N$.

We have to show that $M \stackrel{!}{\supseteq} \square N$.

Suppose given a graph morphism $f : X \rightarrow Y$ in $\square N$. So $f \square N$.

We have to show that $f \stackrel{!}{\in} M$.

Since (C 3) holds, we have a commutative diagram in Gph as follows, where $m \in M$ and $n \in N$.

$$\begin{array}{ccc} X & \xrightarrow{m} & Z \\ f \downarrow & & \downarrow n \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

Since $f \boxtimes N$, we have a commutative diagram in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{m} & Z \\ f \downarrow & \nearrow h & \downarrow n \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

We consider the following commutative diagram in Gph.

$$\begin{array}{ccccc} & & \text{id}_X & & \\ & & \curvearrowright & & \\ X & \xrightarrow{\text{id}_X} & X & \xrightarrow{\text{id}_X} & X \\ f \downarrow & & \downarrow m & & \downarrow f \\ Y & \xrightarrow{h} & Z & \xrightarrow{n} & Y \\ & & \text{id}_Y & & \end{array}$$

Since M is closed under retracts, we obtain $f \in M$.

We have to show that $M^\boxtimes \stackrel{!}{\subseteq} N$.

Suppose given a graph morphism $f : X \rightarrow Y$ in M^\boxtimes . So $M \boxtimes f$.

We have to show that $f \stackrel{!}{\in} N$.

Since (C 3) holds, we have a commutative diagram in Gph as follows, where $m \in M$ and $n \in N$.

$$\begin{array}{ccc} X & \xrightarrow{\text{id}_X} & X \\ m \downarrow & & \downarrow f \\ Z & \xrightarrow{n} & Y \end{array}$$

Since $M \boxtimes f$, we have a commutative diagram in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{\text{id}_X} & X \\ m \downarrow & \nearrow h & \downarrow f \\ Z & \xrightarrow{n} & Y \end{array}$$

We consider the following commutative diagram in Gph.

$$\begin{array}{ccccc} & & \text{id}_X & & \\ & & \curvearrowright & & \\ X & \xrightarrow{m} & Z & \xrightarrow{h} & X \\ f \downarrow & & \downarrow n & & \downarrow f \\ Y & \xrightarrow{\text{id}_Y} & Y & \xrightarrow{\text{id}_Y} & Y \\ & & \text{id}_Y & & \end{array}$$

Since N is closed under retracts, we obtain $f \in N$. □

1.2 Preliminaries on sets

1.2.1 Elementary constructions and properties

Remark 29 Suppose given sets $A \subseteq B \subseteq C$.

We have $C \setminus A = (C \setminus B) \dot{\cup} (B \setminus A)$.

Remark 30 Suppose given sets X, Y, Z .

Suppose given maps $f : X \rightarrow Y$, $g : Y \rightarrow Z$ with g injective.

We have $Z \setminus Xfg = (Z \setminus Yg) \dot{\cup} (Y \setminus Xf)g$.

Proof. We have to show that $Z \setminus Xfg \stackrel{!}{=} (Z \setminus Yg) \dot{\cup} (Yg \setminus Xfg)$, since $Xfg \subseteq Yg \subseteq Z$.

Because g is injective we have $Yg \setminus Xfg = (Y \setminus Xf)g$:

We show $Yg \setminus Xfg \stackrel{!}{=} (Y \setminus Xf)g$.

First, we show $Yg \setminus Xfg \subseteq (Y \setminus Xf)g$

Suppose given $z \in Yg \setminus Xfg$.

There exists $y \in Y$ such that $z = yg$. Then $y \notin Xf$ because of $z = yg \notin Xfg$.

So we have $z = yg \in (Y \setminus Xf)g$.

Second, we show $Yg \setminus Xfg \supseteq (Y \setminus Xf)g$.

Suppose given $z \in (Y \setminus Xf)g$.

There exists $y \in Y \setminus Xf$ such that $z = yg \in Yg$.

We *assume* that $z \in Xfg$. So we have $z = yg \in Xfg$.

There exists $x \in X$ such that $yg = xfg$. Because g is injective we have $y = xf$, which is a *contradiction*.

So we have $z \notin Xfg$. Hence $z \in Yg \setminus Xfg$. □

Remark 31 Suppose given a surjective map $f : X \rightarrow Y$.

Then $f : X \rightarrow Y$ is an epimorphism in Set.

Proof. Suppose given maps: $u, u' : Y \rightarrow Z$ such that $fu = fu'$.

We have to show that $u \stackrel{!}{=} u'$.

Suppose given $y \in Y$.

Since $f : X \rightarrow Y$ is surjective there exists an element $x \in X$ such that $xf = y$.

Since $fu = fu'$ we have $yu = xfu = xfu' = yu'$.

So we conclude that $u = u'$. □

Definition 32 Suppose given a set I .

Suppose given sets A_i for $i \in I$.

Then the *coproduct* or *disjoint union* $\coprod_{i \in I} A_i$ of the sets A_i for $i \in I$ is defined as follows.

$$\coprod_{i \in I} A_i := \bigcup_{i \in I} \{(i, a) : a \in A_i\}.$$

Suppose given $k \in I$. We let

$$\begin{aligned} \iota_k : A_k &\rightarrow \coprod_{i \in I} A_i \\ a &\mapsto (k, a) \end{aligned}$$

If $I = [1, n]$ for some $n \geq 1$, we often write $A_1 \sqcup A_2 \sqcup \dots \sqcup A_n := \coprod_{i \in [1, n]} A_i$.

Example 33 Suppose given sets A and B .

Then we have the coproduct $A \sqcup B$ of A and B as follows; cf. Definition 32.

$$A \sqcup B = \{(1, a) : a \in A\} \cup \{(2, b) : b \in B\}$$

We have

$$\begin{aligned} \iota_1 : A &\rightarrow A \sqcup B \\ a &\mapsto (1, a) \end{aligned}$$

and

$$\begin{aligned} \iota_2 : B &\rightarrow A \sqcup B \\ b &\mapsto (2, b). \end{aligned}$$

Remark 34 Suppose given a set M .

Suppose given a relation $R \subseteq M \times M$.

Let $(\sim) \subseteq M \times M$ the equivalence relation generated by R .

Let

$$\begin{aligned} \rho : M &\rightarrow M/(\sim) \\ m &\mapsto [m]_{(\sim)}. \end{aligned}$$

Suppose given a set M .

Suppose given a map $f : M \rightarrow N$ such that $xf = yf$ for $(x, y) \in R$.

Then there exists a unique map $\bar{f} : M/(\sim) \rightarrow N$ such that $\rho \bar{f} = f$, i.e. we have the following commutative diagram.

$$\begin{array}{ccc} M & \xrightarrow{f} & N \\ \rho \downarrow & \nearrow \exists! \bar{f} & \\ M/(\sim) & & \end{array}$$

Remark 35 Let Q be the subset of bijective maps of $\text{Mor}(\text{Set})$. Then Q satisfies (2 of 6); cf. Definition 1.(1).

Proof. Suppose given sets X, X', Y and Y' .

Suppose given maps $f : X \rightarrow X', g : X' \rightarrow Y$ and $h : Y \rightarrow Y'$.

We have the following commutative diagram in Set .

$$\begin{array}{ccccc} & & X' & \xrightarrow{gh} & Y' \\ & f \nearrow & & \searrow g & \\ X & \xrightarrow{fg} & Y & \xleftarrow{h} & Y' \end{array}$$

We have to show that the composites $fg : X \rightarrow Y$ and $gh : X' \rightarrow Y'$ are bijective if and only if f, g and h are bijective.

First, suppose that fg and gh are bijective.

Since fg is bijective, the map g is surjective.

Since gh is bijective, the map g is injective.

So g is bijective.

Hence $f = (fg) \cdot g^{-1}$ and $h = g^{-1} \cdot (gh)$ are bijective.

Now suppose that f, g and h are bijective.

Then the composites fg and gh are bijective. □

1.2.2 Pushouts in Set

Construction 36 Suppose given the diagram

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \\ X' & & \end{array}$$

in Set .

We consider the coproduct $X' \sqcup Y$; cf. Definition 32.

On this set we have the relation $R := \{((1, xg), (2, xf)) : x \in X\} \subseteq (X' \sqcup Y) \times (X' \sqcup Y)$.

Let (\sim) be the equivalence relation generated by R .

So we have the set of equivalence classes $Y' := (X' \sqcup Y) /_{(\sim)}$.

Let

$$\begin{array}{ccc} Y & \xrightarrow{h} & Y' \\ y & \mapsto & [(2, y)]_{(\sim)} \end{array}$$

and

$$\begin{array}{ccc} X' & \xrightarrow{f'} & Y' \\ x' & \mapsto & [(1, x')]_{(\sim)}. \end{array}$$

Then, in Set, we have the pushout

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}.$$

Proof.

Ad (Pushout 1).

We have $xfh = [(2, xf)]_{(\sim)} = [(1, xg)]_{(\sim)} = xgf'$ for $x \in X$. So $fh = gf'$.

Ad (Pushout 2).

Suppose given a set Z together with maps $u : X' \rightarrow Z$ and $v : Y \rightarrow Z$ such that $fv = gu$.

We have to show that there exists exactly one map $w : Y' \rightarrow Z$ with $f'w = u$ and $hw = v$.

Existence.

We define the map

$$\begin{aligned} \hat{w} : X \sqcup Y &\rightarrow Z \\ (1, x') &\mapsto (1, x')\hat{w} := x'u \\ (2, y) &\mapsto (2, y)\hat{w} := yv. \end{aligned}$$

Suppose given $((1, xg), (2, xf)) \in R$, where $x \in X$. Then $(1, xg)\hat{w} = xgu = xfv = (2, xf)\hat{w}$.

Using $Y' = (X \sqcup Y)_{/(\sim)}$, where (\sim) is the equivalence relation generated by R , we obtain the map

$$\begin{aligned} w : Y' &\rightarrow Z \\ [(1, x')]_{(\sim)} &\mapsto [(1, x')]_{(\sim)}w := x'u \\ [(2, y)]_{(\sim)} &\mapsto [(2, y)]_{(\sim)}w := yv. \end{aligned}$$

$$\begin{array}{ccc} X \sqcup Y & \xrightarrow{\hat{w}} & Z \\ \downarrow & \searrow \exists_! w & \\ Y' = (X \sqcup Y)_{/(\sim)} & & \end{array}$$

Then for $x \in X'$ we have $xf'w = [(1, x')]_{(\sim)}w = x'u$. So we have in fact $f'w = u$.

Then for $y \in Y$ we have $yhw = [(2, y)]_{(\sim)}w = yv$. So we have in fact $hw = v$.

Uniqueness.

Suppose given $\tilde{w} : Y' \rightarrow Z$ with $f'\tilde{w} = u$ and $h\tilde{w} = v$.

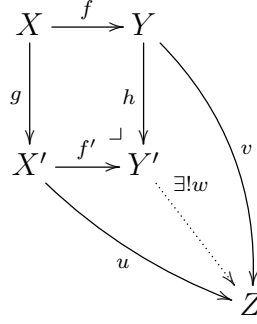
Suppose given $y' \in Y'$.

For $y' \in Y'$ we have $x' \in X$ such that $[(1, x')]_{(\sim)} = y'$ or we have $y \in Y$ such that $[(2, y)]_{(\sim)} = y'$.

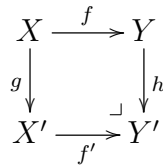
If $y' = [(1, x')]_{(\sim)}$ with $x' \in X$, then we obtain $y'\tilde{w} = [(1, x')]_{(\sim)}\tilde{w} = x'f'\tilde{w} = x'u$.

If $y' = [(2, y)]_{(\sim)}$ with $y \in Y$, then we obtain $y'\tilde{w} = [(2, y)]_{(\sim)}\tilde{w} = yh\tilde{w} = yv$.

Since this holds for every such \tilde{w} , this shows uniqueness.



Remark 37 Suppose given the pushout

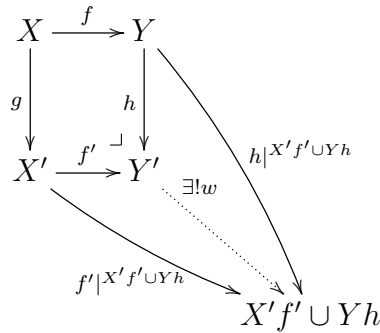


in Set.

Then we have $Y' = X'f' \cup Yh$.

Proof. It suffices to show that $Y' \stackrel{!}{\subseteq} X'f' \cup Yh$.

According to (Pushout 2) there exists a unique map $w : Y' \rightarrow X'f' \cup Yh$ such that $hw = h|_{X'f' \cup Yh}$ and $f'w = f|_{X'f' \cup Yh}$.



For $y' \in Y$ we calculate $y'w$.

If $y' \in X'f'$, then we write $y' = x'f'$ for some $x' \in X'$. We obtain $y'w = x'f'w = x'(f'|_{X'f' \cup Yh}) = x'f' = y'$.

If $y' \in Yh$, then we write $y' = yh$ for some $y \in Y$. We obtain $y'w = yhw = y(h|_{X'f' \cup Yh}) = yh = y'$.

So $y'w = y'$ in both cases.

In particular, $y' = y'w \in X'f' \cup Yh$.

So we have $Y' \subseteq X'f' \cup Yh$ and thus we have $Y' = X'f' \cup Yh$. □

Remark 38 Suppose given

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \\ X' & & \end{array}$$

in Set with an injective map $f : X \rightarrow X'$.

Let $Y' := (X' \setminus Xf) \sqcup Y$.

Let

$$a' : X' \rightarrow Y' \\ x' \mapsto \begin{cases} (2, xa) & \text{if } x' = xf \in Xf \text{ for a unique } x \in X \\ (1, x') & \text{if } x' \in X' \setminus Xf. \end{cases}$$

and we let

$$g : Y \rightarrow (X' \setminus Xf) \sqcup Y \\ y \mapsto (2, y)$$

Then

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

is a pushout in Set.

Proof.

Suppose given $x \in X$. We have $xf a' = (2, xa) = x a g$. So we have $f a' = a g$.

Now we show that the commutative diagram

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

is a pushout in Set.

Suppose given

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array} \begin{array}{c} \searrow v \\ \downarrow \\ \searrow u \end{array} T$$

in Set such that $av = fu$.

We have to show that there exists a unique map $w : Y' \rightarrow T$ such that $a'w = u$ and $gw = v$.

Uniqueness. This follows from $X'a' \cup Yg = Y'$.

We show the *existence* of a map $w : Y' \rightarrow T$ such that $a'w = u$ and $gw = v$.

We let

$$\begin{aligned} w : \quad Y' &\rightarrow T \\ (1, x') &\mapsto x'u \\ (2, y) &\mapsto yv . \end{aligned}$$

So we have

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \\ & \searrow u & \downarrow w \\ & & T \end{array}$$

We have to show that $a'w \stackrel{!}{=} u$.

Suppose given $x' \in X'$.

$$\begin{aligned} \text{We have } x'a'w &= \left\{ \begin{array}{ll} (2, xa)w & \text{if } \exists! x \in X : x' = xf \\ (1, x')w & \text{if } x' \in X' \setminus Xf \end{array} \right\} = \left\{ \begin{array}{ll} xav & \text{if } \exists! x \in X : x' = xf \\ x'u & \text{if } x' \in X' \setminus Xf \end{array} \right\} \stackrel{fu=av}{=} \\ \left\{ \begin{array}{ll} xfu & \text{if } \exists! x \in X : x' = xf \\ x'u & \text{if } x' \in X' \setminus Xf \end{array} \right\} &= \left\{ \begin{array}{ll} x'u & \text{if } x' \in Xf \\ x'u & \text{if } x' \in X' \setminus Xf \end{array} \right\} = x'u. \end{aligned}$$

We have to show that $f'w \stackrel{!}{=} v$.

For $y \in Y$ we have $yf'w = (2, y)w = yv$. □

Remark 39 Suppose given

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

in Set.

If the map $f : X \rightarrow X'$ is injective, then the map $g : Y \rightarrow Y'$ is injective.

Proof.

Without loss of generality, the pushout is constructed in the way as in Remark 38; cf. Remark 7.

Then the map g is in fact injective. □

Remark 40 Suppose given

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

in Set.

If the map $f : X \rightarrow X'$ is surjective, then the map $g : Y \rightarrow Y'$ is surjective.

Proof.

Without loss of generality, the pushout is constructed in the way as in Construction 36; cf. Remark 7.

So we have $Y' = (X' \sqcup Y) /_{(\sim)}$ where (\sim) is the equivalence relation generated by $\{(1, xf), (2, xa) : x \in X\}$.

Suppose given $y' \in Y'$.

We have to show that there exists $y \in Y$ such that $yg = y'$.

Case: There exists $y \in Y$ such that $y' = [(2, y)]_{(\sim)}$.

Then we have $yg = [(2, y)]_{(\sim)} = y'$.

Case: There exists $x' \in X'$ such that $y' = [(1, x')]_{(\sim)}$.

Since $f : X \rightarrow X'$ is surjective there exists $x \in X$ such that $xf = x'$.

So we have $y' = [(1, x')]_{(\sim)} = [(1, xf)]_{(\sim)} \stackrel{\text{Def. } (\sim)}{=} [(2, xa)]_{(\sim)} \stackrel{\text{Def. } g}{=} (xa)g$. □

Without Construction 36 we can use $Y' = X'a' \cup Yg$ by Remark 37 to prove this Remark 40.

1.2.3 Pullbacks in Set

Construction 41 Suppose given the diagram

$$\begin{array}{ccc} & & Y \\ & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Set.

Define the set $X := \{(x', y) \in X' \times Y : x'f' = yh\}$.

Let

$$\begin{array}{ccc} X & \xrightarrow{g} & X' \\ (x', y) & \mapsto & x' \end{array}$$

and

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ (x', y) & \mapsto & y. \end{array}$$

Then, in Set , we have the pullback

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & \lrcorner & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}.$$

Proof. We have $(x', y)fh = yh = x'f' = (x', y)gf'$ for $(x', y) \in X$. So $fh = gf'$. \square

Universal property.

Suppose given a set Z together with two maps $u : Z \rightarrow Y$ and $v : Z \rightarrow X'$ such that $uh = vf'$.

We have to show that there exists exactly one map $w : Z \rightarrow X$ with $wf = u$ and $wg = v$.

Existence.

We define the map

$$\begin{aligned} w : Z &\rightarrow X \\ z &\mapsto zw := (zv, zu). \end{aligned}$$

The element (zv, zu) is in fact contained in X since $zv f' = zuh$.

Then $wf = u$ since for $z \in Z$ we have $zwf = (zv, zu)f = zu$.

Moreover, $wg = v$ since for $z \in Z$ we have $zwg = (zv, zu)g = zv$.

Uniqueness.

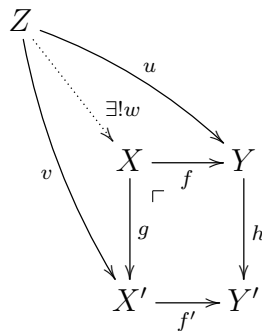
Suppose given $\tilde{w} : Z \rightarrow X$ with $\tilde{w}g = v$ and $\tilde{w}f = u$.

Suppose given $z \in Z$. Write $z\tilde{w} =: (x', y) \in X$.

We obtain $x' = z\tilde{w}g = zv$ and $y = z\tilde{w}f = zu$.

So $z\tilde{w} = (zv, zu)$.

Since this holds for every such \tilde{w} , this shows uniqueness.



1.2.4 Colimit of a countable chain in Set

Suppose given

$$X_0 \xrightarrow{f_{0,1}} X_1 \xrightarrow{f_{1,2}} X_2 \xrightarrow{f_{2,3}} \dots$$

in Set.

Let $f_{k,l} := f_{k,k+1} \cdot \dots \cdot f_{l-1,l} : X_k \rightarrow X_l$ for $0 \leq k \leq l$. In particular, $f_{k,k} = \text{id}_{X_k}$.

We define the relation (\sim) on $\bigsqcup_{i \geq 0} X_i$ as follows.

For $(i, x_i), (j, x_j) \in \bigsqcup_{i \geq 0} X_i$ we let

$$(i, x_i) \sim (j, x_j) :\Leftrightarrow \text{There exists } m \geq \max\{j, k\} \text{ such that } x_i f_{i,m} = x_j f_{j,m}.$$

The relation (\sim) is an equivalence relation:

For $(j, x_j) \in \bigsqcup_{i \geq 0} X_i$ we have $(j, x_j) \sim (j, x_j)$, since $x_j f_{j,j} = x_j \text{id}_{X_j} = x_j f_{j,j}$.

The relation (\sim) is symmetric by definition.

Suppose given $(j, x_j), (k, x_k), (l, x_l) \in \bigsqcup_{i \geq 0} X_i$ such that $(j, x_j) \sim (k, x_k)$ and $(k, x_k) \sim (l, x_l)$.

So there exist $m \geq \max\{j, k\}$ and $n \geq \max\{k, l\}$ such that $x_j f_{j,m} = x_k f_{k,m}$ and $x_k f_{k,n} = x_l f_{l,n}$.

Let $p := \max\{m, n\}$.

Then we have $x_j f_{j,p} = x_j f_{j,m} f_{m,p} = x_k f_{k,m} f_{m,p} = x_k f_{k,p} = x_k f_{k,n} f_{n,p} = x_l f_{l,n} f_{n,p} = x_l f_{l,p}$.

So the relation (\sim) is transitive.

Let $[j, x_j]$ be the equivalence class of (j, x_j) with respect to (\sim) .

Definition 42

(1) We define the *colimit* $X_\infty := \varinjlim_{i \geq 0} X_i := \bigsqcup_{i \geq 0} X_i / (\sim) = \{[j, x_j] : j \geq 0, x_j \in X_j\}$.

(2) For $k \geq 0$, we let

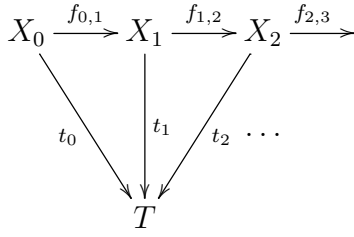
$$\begin{aligned} f_{k,\infty} : X_k &\rightarrow X_\infty \\ x_k &\mapsto [k, x_k]. \end{aligned}$$

Suppose given $0 \leq k \leq l$. Suppose given $x_k \in X_k$. We have $(k, x_k) \sim (l, x_k f_{k,l})$, since $x_k f_{k,l} = (x_k f_{k,l}) \text{id}_{X_l} = (x_k f_{k,l}) f_{l,l}$.

So we have $x_k f_{k,l} f_{l,\infty} = [l, x_k f_{k,l}] = [k, x_k] = x_k f_{k,\infty}$.

So we have $f_{k,l} f_{l,\infty} = f_{k,\infty}$.

Lemma 43 (universal property) Suppose given maps $t_j : X_j \rightarrow T$ for $j \geq 0$ such that $f_{j,j+1}t_{j+1} = t_j$ for $j \geq 0$, i.e. $f_{j,k}t_k = t_j$ for $0 \leq j \leq k$.



Then there exists a unique map $t_\infty : X_\infty \rightarrow T$ such that $f_{j,\infty}t_\infty = t_j$ for $j \geq 0$.

Proof.

Uniqueness.

Suppose given $t', t'' : X_\infty \rightarrow T$ such that $f_{j,\infty}t' = t_j$ and $f_{j,\infty}t'' = t_j$ for $j \geq 0$.

We have to show that $t' = t''$.

Let $[j, x_j] \in X_\infty$.

We have to show that $[j, x_j]t' \stackrel{!}{=} [j, x_j]t''$.

In fact, we have $[j, x_j]t' = x_j f_{j,\infty}t' = x_j t_j = x_j f_{j,\infty}t'' = [j, x_j]t''$.

Existence.

Let

$$\begin{aligned}
 t_\infty : X_\infty &\rightarrow T \\
 [j, x_j] &\mapsto x_j t_j .
 \end{aligned}$$

We show that this map is well-defined.

Suppose given $(j, x_j), (k, x_k) \in \bigsqcup_{i \geq 0} X_i$ such that $[j, x_j] = [k, x_k]$.

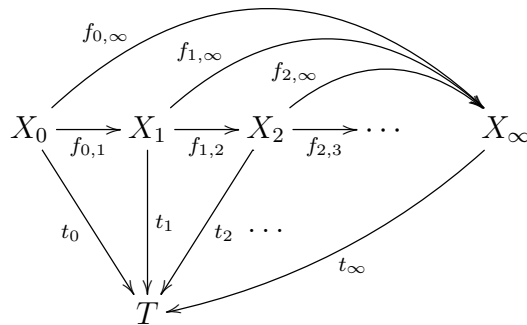
Then $x_j f_{j,m} = x_k f_{k,m}$ for some $m \geq \max\{j, k\}$.

We have to show that $x_j t_j \stackrel{!}{=} x_k t_k$.

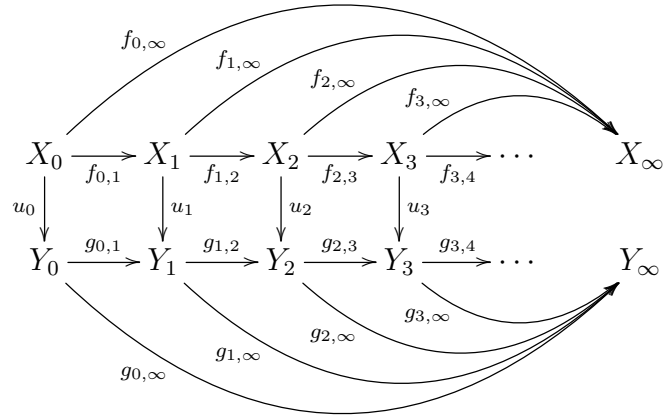
In fact, we have $x_j t_j = x_j f_{j,m} t_m = x_k f_{k,m} t_m = x_k t_k$.

Suppose given $j \geq 0$. Suppose given $x_j \in X_j$. We have $x_j f_{j,\infty} t_\infty = [j, x_j] t_\infty \stackrel{\text{Def. } t_\infty}{=} x_j t_j$.

So we have $f_{j,\infty} t_\infty = t_j$.



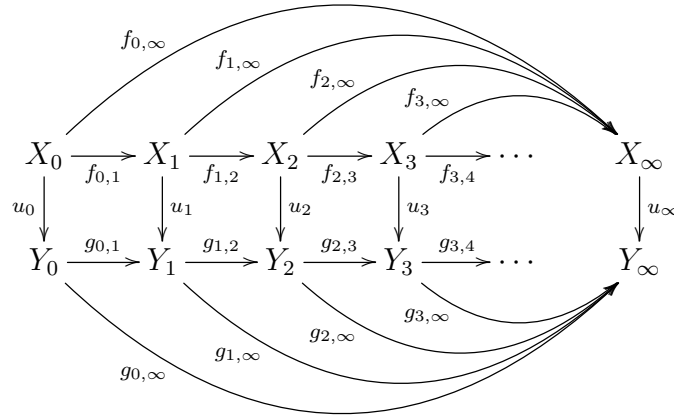
Definition 44 Suppose given a commutative diagram in Set as follows; cf. Definition 42.



We have $f_{j,j+1}u_{j+1}g_{j+1,\infty} = u_jg_{j,j+1}g_{j+1,\infty} = u_jg_{j,\infty}$ for $j \geq 0$.

So because of the universal property there exists a unique map $u_\infty : X_\infty \rightarrow Y_\infty$ such that $f_{j,\infty}u_\infty = u_jg_{j,\infty}$ for $j \geq 0$; cf. Lemma 43.

We let $\varinjlim_{i \geq 0} u_i := u_\infty$.



Chapter 2

Graphs

2.1 Definitions for graphs and graph morphisms

Definition 45 A graph $G = (V_G, E_G, s_G, t_G)$ is a tuple consisting of a set of vertices V_G and a set of edges E_G together with maps $s_G : E_G \rightarrow V_G$, the source map, and $t_G : E_G \rightarrow V_G$, the target map.

Remark 46 Suppose given a graph $G = (V_G, E_G, s_G, t_G)$. The elements of V_G are called *vertices* and the elements of E_G are called *edges*.

The element $e s_G \in V_G$ is called the *source* of $e \in E_G$.

The element $e t_G \in V_G$ is called the *target* of $e \in E_G$.

Pictorially, we represent G by writing out the vertices of G and by drawing an arrow for each edge e of G , pointing from its source $e s_G$ to its target $e t_G$.

Definition 47 Suppose given a graph $G = (V_G, E_G, s_G, t_G)$.

- (1) A graph $G' = (V_{G'}, E_{G'}, s_{G'}, t_{G'})$ is called a *subgraph* of G if $V_{G'} \subseteq V_G$, $E_{G'} \subseteq E_G$, $s_{G'} = s_G|_{E_{G'}}$, and $t_{G'} = t_G|_{E_{G'}}$.

To indicate that G' is a subgraph of G , we write $G' \subseteq G$.

- (2) Suppose given subgraphs $H \subseteq G$ and $K \subseteq G$. The *intersection* $H \cap K$ of the subgraphs H and K is the subgraph

$$H \cap K = (V_{H \cap K}, E_{H \cap K}, s_{H \cap K}, t_{H \cap K}) := (V_H \cap V_K, E_H \cap E_K, s_G|_{E_H \cap E_K}^{V_H \cap V_K}, t_G|_{E_H \cap E_K}^{V_H \cap V_K})$$

of G .

- (3) A subgraph $G' \subseteq G$ is called a *full subgraph* if $E_{G'} = \{e \in E_G : e s_G \in V_{G'} \text{ and } e t_G \in V_{G'}\}$, i.e. if for $v', w' \in V_{G'}$, each edge e of G having $e s_G = v'$ and $e t_G = w'$ is already an edge of G' .
- (4) A graph $G = (V_G, E_G, s_G, t_G)$ is called *finite* if the sets V_G and E_G both are finite.

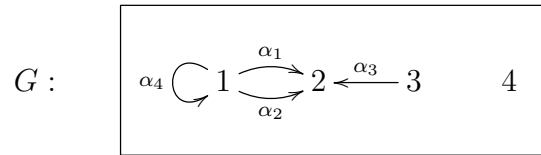
Remark 48 Suppose given a graph G .

- (1) To define a subgraph G' of G , it suffices to give subsets $E_{G'} \subseteq E_G$ and $V_{G'} \subseteq V_G$ such that $E_{G'} s_G \subseteq V_{G'}$ and $E_{G'} t_G \subseteq V_{G'}$.
- (2) To define a full subgraph G' of G , it suffices to give a subset $V_{G'} \subseteq V_G$.

Example 49 The graph G having $V_G = \{1, 2, 3, 4\}$ and $E_G = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ with

$$\begin{aligned}\alpha_1 s_G &= 1, & \alpha_1 t_G &= 2, \\ \alpha_2 s_G &= 1, & \alpha_2 t_G &= 2, \\ \alpha_3 s_G &= 3, & \alpha_3 t_G &= 2, \\ \alpha_4 s_G &= 1, & \alpha_4 t_G &= 1,\end{aligned}$$

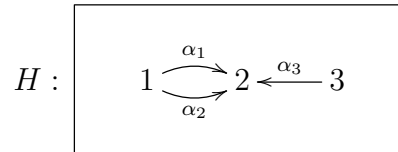
is represented the following way.



Note that the graph H having $V_H = \{1, 2, 3\}$ and $E_H = \{\alpha_1, \alpha_2, \alpha_3\}$ with

$$\begin{aligned}\alpha_1 s_H &= 1, & \alpha_3 t_H &= 2, \\ \alpha_2 s_H &= 1, & \alpha_3 t_H &= 2, \\ \alpha_3 s_H &= 3, & \alpha_3 t_H &= 2,\end{aligned}$$

which is represented as

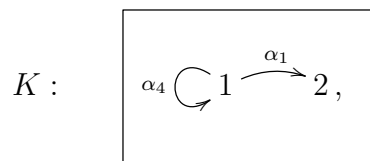


is a full subgraph of G ; cf. Definition 47.

Moreover, the graph K having $V_K = \{1, 2\}$ and $E_K = \{\alpha_1, \alpha_4\}$ with

$$\begin{aligned}\alpha_1 s_K &= 1, & \alpha_1 t_K &= 2, \\ \alpha_4 s_K &= 1, & \alpha_4 t_K &= 1,\end{aligned}$$

which is represented as

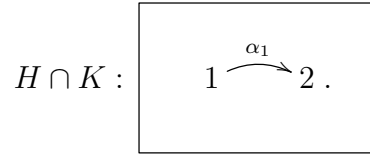


is a subgraph of G , but not a full subgraph.

As intersection, we obtain $H \cap K$ having $V_{H \cap K} = \{1, 2\}$ and $E_{H \cap K} = \{\alpha_1\}$ with

$$\alpha_1 s_{H \cap K} = 1, \quad \alpha_1 t_{H \cap K} = 2,$$

which is represented as



Remark 50 Suppose given a graph G .

Suppose given full subgraphs $G' \subseteq G$ and $G'' \subseteq G$ with $V_{G'} = V_{G''}$.

Then $G' = G''$.

Proof. We have to show that $G' \stackrel{!}{=} G''$. It remains to show that $E_{G'} \stackrel{!}{=} E_{G''}$ as subsets of E_G .

Write $V := V_{G'} = V_{G''}$. We have $E_{G'} = \{e \in E_G : e s_G \in V \text{ and } e t_G \in V\} = E_{G''}$. \square

Notation 51 Suppose given a graph G .

Suppose given vertices $v, w \in V_G$.

Suppose given subsets $V, W \subseteq V_G$.

(1) We denote $G(v, w) := \{e \in E_G : e s_G = v, e t_G = w\} \subseteq E_G$.

Write $G(v, *) := \{e \in E_G : e s_G = v\} \subseteq E_G$.

Write $G(*, w) := \{e \in E_G : e t_G = w\} \subseteq E_G$.

(2) We denote $G(V, W) := \{e \in E_G : e s_G \in V, e t_G \in W\} \subseteq E_G$.

Write $G(V, w) := G(V, \{w\})$ and $G(v, W) := G(\{v\}, W)$.

Write $G(V, *) := \{e \in E_G : e s_G \in V\} \subseteq E_G$.

Write $G(*, W) := \{e \in E_G : e t_G \in W\} \subseteq E_G$.

In particular, $G(v, *) = G(\{v\}, *)$ and $G(*, w) = G(*, \{w\})$.

Definition 52 Let $n \in \mathbb{N}$. We will define a graph C_n .

For $i \in \mathbb{Z}$ we often abbreviate $i := i + n\mathbb{Z} \in \mathbb{Z}/n\mathbb{Z}$.

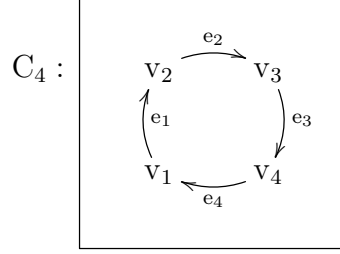
To denote vertices in C_n , we use symbols v_i , for $i \in \mathbb{Z}/n\mathbb{Z}$. To denote edges in C_n , we use symbols e_i , for $i \in \mathbb{Z}/n\mathbb{Z}$.

The *cyclic graph* $C_n = (V_{C_n}, E_{C_n}, s_{C_n}, t_{C_n})$ is the graph with $V_{C_n} := \{v_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$,

$E_{C_n} := \{e_i \mid i \in \mathbb{Z}/n\mathbb{Z}\}$ and with

$$e_i s_{C_n} = v_i, \quad e_i t_{C_n} = v_{i+1} \quad \text{for } i \in \mathbb{Z}/n\mathbb{Z}.$$

Example 53 For example the cyclic graph C_4 is represented the following way.



Note that $v_4 = v_0$ and $e_4 = e_0$.

Definition 54 Suppose given graphs $G = (V_G, E_G, s_G, t_G)$ and $H = (V_H, E_H, s_H, t_H)$.

A *graph morphism* $f = (V_f, E_f) : G \rightarrow H$ between G and H is a tuple consisting of a map on the vertices of the graphs $V_f : V_G \rightarrow V_H$ and a map on the edges of the graphs $E_f : E_G \rightarrow E_H$ such that (Morph 1–2) hold.

(Morph 1) We have $E_f s_H = s_G V_f$.

(Morph 2) We have $E_f t_H = t_G V_f$.

The graph morphism $f = (V_f, E_f)$ is called *injective* if V_f and E_f are injective.

The graph morphism $f = (V_f, E_f)$ is called *surjective* if V_f and E_f are surjective.

The graph morphism $f = (V_f, E_f)$ is called *bijective* if V_f and E_f are bijective.

Definition 55 Suppose given a graph $G = (V_G, E_G, s_G, t_G)$.

The graph morphism $\text{id}_G := (V_{\text{id}_G}, E_{\text{id}_G}) : G \rightarrow G$ with

$$\begin{aligned} V_{\text{id}_G} &:= \text{id}_{V_G} : V_G \rightarrow V_G \\ v &\mapsto v \end{aligned}$$

for $v \in V_G$ and with

$$\begin{aligned} E_{\text{id}_G} &:= \text{id}_{E_G} : E_G \rightarrow E_G \\ e &\mapsto e \end{aligned}$$

for $e \in E_G$ is called the *identity on G* .

Note that the identity id_G is bijective and thus a graph isomorphism.

Definition 56 Suppose given $n \in \mathbb{N}$ and $k, l \in \mathbb{N}$ with $k \leq l$. We will define a graph D_n , as well as graph morphisms $r_n : D_n \rightarrow C_n$ and $\iota_{k,l} : D_k \rightarrow D_l$.

- (1) To denote vertices in D_n , we use symbols \hat{v}_i , for $i \in [0, n]$. To denote edges in D_n , we use symbols \hat{e}_i , for $i \in [0, n-1]$.

The *direct graph* $D_n = (V_{D_n}, E_{D_n}, s_{D_n}, t_{D_n})$ is the graph with $V_{D_n} := \{\hat{v}_i \mid i \in [0, n]\}$,

$E_{D_n} := \{\hat{e}_i \mid i \in [0, n-1]\}$ and with

$$\hat{e}_i s_{D_n} = \hat{v}_i \text{ for } i \in [0, n-1], \quad \hat{e}_i t_{D_n} = \hat{v}_{i+1} \text{ for } i \in [0, n-1].$$

(2) We have the graph morphism

$$\begin{aligned} r_n &: D_n \rightarrow C_n \\ V_{r_n} &: \hat{v}_i \rightarrow v_{i+n\mathbb{Z}} = v_i \text{ for } i \in [0, n] \\ E_{r_n} &: \hat{e}_i \rightarrow e_{i+n\mathbb{Z}} = e_i \text{ for } i \in [0, n-1]. \end{aligned}$$

To verify that r_n is in fact a graph morphism, we have to show that $\hat{e}_i s_{D_n} V_{r_n} = \hat{e}_i E_{r_n} s_{C_n}$ and that $\hat{e}_i t_{D_n} V_{r_n} = \hat{e}_i E_{r_n} t_{C_n}$ for $i \in [0, n-1]$.

We have $\hat{e}_i s_{D_n} V_{r_n} = \hat{v}_i V_{r_n} = v_{i+n\mathbb{Z}} = e_{i+n\mathbb{Z}} s_{C_n} = \hat{e}_i E_{r_n} s_{C_n}$ for $i \in [0, n-1]$ and we have $\hat{e}_i t_{D_n} V_{r_n} = \hat{v}_{i+1} V_{r_n} = v_{i+1+n\mathbb{Z}} = e_{i+n\mathbb{Z}} t_{C_n} = \hat{e}_i E_{r_n} t_{C_n}$ for $i \in [0, n-1]$.

So $r_n : D_n \rightarrow C_n$ is in fact a graph morphism. \square

(3) We have the graph morphism

$$\begin{aligned} \iota_{k,l} &: D_k \rightarrow D_l \\ V_{\iota_{k,l}} &: \hat{v}_i \rightarrow \hat{v}_i \text{ for } i \in [0, k] \\ E_{\iota_{k,l}} &: \hat{e}_i \rightarrow \hat{e}_i \text{ for } i \in [0, k-1]. \end{aligned}$$

We often identify the direct graph D_k with the subgraph $D_k \iota_{k,l} \subseteq D_l$ of the direct graph D_l .

We often abbreviate $\iota_n := \iota_{0,n} : D_0 \rightarrow D_n$, where $V_{\iota_n} : \hat{v}_0 \mapsto \hat{v}_0$.

Example 57 For example the direct graph D_1 is represented the following way.

$$D_1 : \boxed{\hat{v}_0 \xrightarrow{\hat{e}_0} \hat{v}_1}$$

Example 58 For example the direct graph D_3 is represented the following way.

$$D_3 : \boxed{\hat{v}_0 \xrightarrow{\hat{e}_0} \hat{v}_1 \xrightarrow{\hat{e}_1} \hat{v}_2 \xrightarrow{\hat{e}_2} \hat{v}_3}$$

Remark 59 Suppose given a graph G .

Suppose given an edge $e \in E_G$.

Then there exists a unique graph morphism $f : D_1 \rightarrow G$ such that $\hat{e}_0 E_f = e$.

Proof.

Existence. Let $\hat{e}_0 E_f := e$, $\hat{v}_0 V_f := e s_G$ and $\hat{v}_1 V_f := e t_G$.

We have $\hat{e}_0 s_{D_1} V_f = \hat{v}_0 V_f = e s_G = \hat{e}_0 E_f s_G$ and $\hat{e}_0 t_{D_1} V_f = \hat{v}_1 V_f = e t_G = \hat{e}_0 E_f t_G$.

So f is in fact a graph morphism.

Uniqueness. Given $f : D_1 \rightarrow G$ such that $\hat{e}_0 E_f = e$, we necessarily have $\hat{v}_0 V_f = \hat{e}_0 s_{D_1} V_f = \hat{e}_0 E_f s_G = e s_G$ and $\hat{v}_1 V_f = \hat{e}_0 t_{D_1} V_f = \hat{e}_0 E_f t_G = e t_G$. \square

Definition 60 Suppose given graphs $X = (V_X, E_X, s_X, t_X)$, $Y = (V_Y, E_Y, s_Y, t_Y)$ and $Z = (V_Z, E_Z, s_Z, t_Z)$.

Suppose given graph morphisms $f = (V_f, E_f) : X \rightarrow Y$ and $g = (V_g, E_g) : Y \rightarrow Z$.

Let $f \cdot g := (V_f \cdot V_g, E_f \cdot E_g)$. We often write $fg := f \cdot g$.

Then $f \cdot g$ is also a graph morphism, called the *composite* of the graph morphisms f and g .

Proof. We need to show that fg is a graph morphism.

We have to show that (Morph 1) and (Morph 2) hold for fg .

So we have to show that $E_{fg} s_Z \stackrel{!}{=} s_X V_{fg}$ and that $E_{fg} t_Z \stackrel{!}{=} t_X V_{fg}$.

We have $E_{fg} s_Z = E_f \cdot E_g \cdot s_Z \stackrel{\text{(Morph 1) for } g}{=} E_f \cdot s_Y \cdot V_g \stackrel{\text{(Morph 1) for } f}{=} s_X \cdot V_f \cdot V_g = s_X \cdot V_{fg}$.

We have $E_{fg} t_Z = E_f \cdot E_g \cdot t_Z \stackrel{\text{(Morph 2) for } g}{=} E_f \cdot t_Y \cdot V_g \stackrel{\text{(Morph 2) for } f}{=} t_X \cdot V_f \cdot V_g = t_X \cdot V_{fg}$. \square

So

$$\begin{aligned} V_{f \cdot g} &= V_f \cdot V_g \\ E_{f \cdot g} &= E_f \cdot E_g . \end{aligned}$$

Definition 61 Suppose given graphs G and H . Suppose given a graph morphism $f : G \rightarrow H$.

The *image* Gf of the graph morphism f is the graph $(V_{Gf}, E_{Gf}, s_{Gf}, t_{Gf})$ consisting of the set of vertices $V_{Gf} := V_G V_f \subseteq V_H$, the set of edges $E_{Gf} := E_G E_f \subseteq E_H$, the source map $s_{Gf} := s_H \big|_{E_{Gf}}^{V_{Gf}}$ and the target map $t_{Gf} := t_H \big|_{E_{Gf}}^{V_{Gf}}$.

Note that the image Gf of the graph morphism f is a subgraph of H ; cf. Definitions 47 and 54.

Example 62 Let G be the graph having $V_G = \{1, 2, 3\}$ and $E_G = \{\alpha_1, \alpha_2\}$ with

$$\begin{aligned} \alpha_1 s_G &= 1, & \alpha_1 t_G &= 2, \\ \alpha_2 s_G &= 3, & \alpha_2 t_G &= 2, \end{aligned}$$

Let H be the graph having $V_H = \{1, 2, 3\}$ and $E_H = \{\beta_1, \beta_2\}$ with

$$\begin{aligned} \beta_1 s_H &= 1, & \beta_1 t_H &= 2, \\ \beta_2 s_H &= 2, & \beta_2 t_H &= 3, \end{aligned}$$

Now we have the following situation

$$G : \boxed{1 \xrightarrow{\alpha_1} 2 \xleftarrow{\alpha_2} 3}$$

$$H : \boxed{1 \xrightarrow{\beta_1} 2 \xrightarrow{\beta_2} 3}$$

Let $f = (V_f, E_f) : G \rightarrow H$ be the graph morphism with

$$1 V_f = 1, \quad 2 V_f = 2, \quad 3 V_f = 1,$$

and with

$$\begin{aligned} \alpha_1 E_f &= \beta_1, \\ \alpha_2 E_f &= \beta_1, \end{aligned}$$

Then the image Gf of the graph morphism f is the graph having $V_{Gf} = \{1, 2\}$ and $E_{Gf} = \{\beta_1\}$ with

$$\beta_1 s_{Gf} = 1, \quad \beta_1 t_{Gf} = 2,$$

The image Gf is represented as

$$Gf : \boxed{1 \xrightarrow{\beta_1} 2}$$

Definition 63 Suppose given a graph morphism $f = (V_f, E_f) : G \rightarrow H$.

Suppose given subgraphs $G' \subseteq G$ and $H' \subseteq H$ such that $G'f \subseteq H'$.

Let the *restriction* of f to G' and H' be defined as

$$f|_{G'}^{H'} := (V_f|_{V_{G'}^{H'}}, E_f|_{E_{G'}^{H'}})$$

Then $f|_{G'}^{H'} : G' \rightarrow H'$ is a graph morphism.

We have $V_{f|_{G'}^{H'}} = V_f|_{V_{G'}^{H'}}$ and $E_{f|_{G'}^{H'}} = E_f|_{E_{G'}^{H'}}$.

If $H' = H$, then we also write $f|_{G'} := f|_{G'}^H : G' \rightarrow H$.

If $G' = G$, then we also write $f|^{H'} := f|_G^{H'} : G \rightarrow H'$.

Note that, in particular, we can restrict the graph morphism f to the subgraph $Gf \subseteq H$ to obtain the surjective graph morphism $f|^{Gf} : G \rightarrow Gf$.

Moreover, we obtain the *inclusion morphism* $\text{id}_G|_{G'} : G' \rightarrow G$.

Definition 64 The *category of graphs* Gph consists of the set of objects

$$\text{Ob}(\text{Gph}) := \{G : G \text{ is a graph}\}$$

and the set of morphisms

$$\text{Mor}(\text{Gph}) := \{f : f \text{ is a graph morphism}\}.$$

Cf. Definitions 45 and 54.

The category Gph has the identity morphisms introduced in Definition 55 and carries the composition introduced in Definition 60.

Suppose given graphs $G, H \in \text{Ob}(\text{Gph})$. By

$$(G, H)_{\text{Gph}} := \{G \xrightarrow{f} H : f \text{ is a graph morphism}\} \subseteq \text{Mor}(\text{Gph})$$

we denote the set of graph morphisms from G to H . We often abbreviate $(G, H) := (G, H)_{\text{Gph}}$.

Remark 65 Suppose given graphs $G, H \in \text{Ob}(\text{Gph})$. A graph morphism $f : G \rightarrow H$ is an isomorphism if and only if it is bijective, i.e. if the map $V_f : V_G \rightarrow V_H$ and the map $E_f : E_G \rightarrow E_H$ both are bijective.

Proof. Suppose given a bijective graph morphism $f = (V_f, E_f) : G \rightarrow H$.

Then the map $V_f : V_G \rightarrow V_H$ is bijective and so there exists its inverse $V_f^{-1} : V_H \rightarrow V_G$.

Moreover, the map $E_f : E_G \rightarrow E_H$ is bijective and so there exists its inverse $E_f^{-1} : E_H \rightarrow E_G$.

Therefore $(V_f^{-1}, E_f^{-1}) : H \rightarrow G$ is a graph morphism with $ff^{-1} = \text{id}_G$ and $f^{-1}f = \text{id}_H$, since we have $E_f^{-1}s_G = E_f^{-1}s_G V_f V_f^{-1} = E_f^{-1}E_f s_H V_f^{-1} = s_H V_f^{-1}$ and $E_f^{-1}t_G = E_f^{-1}t_G V_f V_f^{-1} = E_f^{-1}E_f t_H V_f^{-1} = t_H V_f^{-1}$.

Furthermore, $(V_f, E_f)(V_f^{-1}, E_f^{-1}) = (V_f V_f^{-1}, E_f E_f^{-1}) = (\text{id}_{V_G}, \text{id}_{E_G}) = \text{id}_G$ and $(V_f^{-1}, E_f^{-1})(V_f, E_f) = (V_f^{-1} V_f, E_f^{-1} E_f) = (\text{id}_{V_H}, \text{id}_{E_H}) = \text{id}_H$. Thus $f^{-1} = (V_{f^{-1}}, E_{f^{-1}}) = (V_f^{-1}, E_f^{-1})$. So f is an isomorphism.

Conversely, if f is an isomorphism, then there exists a graph morphism $g : H \rightarrow G$ such that $fg = \text{id}_G$ and $gf = \text{id}_H$. Hence there exist maps $V_g : V_H \rightarrow V_G$ and $E_g : E_H \rightarrow E_G$ such that $V_f V_g = \text{id}_{V_G}$ and $V_g V_f = \text{id}_{V_H}$ and such that $E_f E_g = \text{id}_{E_G}$ and $E_g E_f = \text{id}_{E_H}$.

So both maps V_f and E_f are bijective. □

Remark 66 Suppose given an injective graph morphism $f : X \rightarrow Y$.

Then the restriction $f|^{Xf} : X \rightarrow Xf$ is a graph isomorphism.

Proof. We have to show that the graph morphism $f|^{Xf}$ is bijective; cf. Remark 65.

The graph morphism f is injective by supposition. Hence the graph morphism $f|^{Xf}$ is injective.

By construction, the graph morphism $f|^{Xf}$ is surjective.

So the graph morphism $f|^{Xf}$ is bijective. □

Definition 67 Suppose given $n \in \mathbb{N}$.

Suppose given $s \in \mathbb{Z}$.

We define the graph automorphism

$$\begin{aligned} a_s : C_n &\xrightarrow{\sim} C_n \\ e_i &\mapsto e_{i+s} \\ v_i &\mapsto v_{i+s} . \end{aligned}$$

In fact, for $i \in \mathbb{Z}/n\mathbb{Z}$ we have

$$(e_i E_{a_s}) s_{C_n} = e_{i+s} s_{C_n} = v_{i+s} = v_i V_{a_s} = (e_i s_{C_n}) V_{a_s} \text{ and}$$

$$(e_i E_{a_s}) t_{C_n} = e_{i+s} t_{C_n} = v_{i+s+1} = v_{i+1} V_{a_s} = (e_i t_{C_n}) V_{a_s} .$$

So a_s is a graph morphism; cf. Definition 54.

We have $a_s \cdot a_t = a_{s+t}$ for $s, t \in \mathbb{Z}$.

We have $a_0 = \text{id}_{C_n}$; cf. Definition 55.

In particular, we have $a_s \cdot a_{-s} = a_0 = \text{id}_{C_n}$ and $a_{-s} \cdot a_s = a_0 = \text{id}_{C_n}$, so that $a_{-s} = a_s^{-1}$ for $s \in \mathbb{Z}$.

So a_s is a graph automorphism.

Definition 68 Suppose given graphs G, H, X, Y .

Suppose given a graph morphism $f : G \rightarrow H$.

We have the map

$$\begin{array}{ccc} (X, G)_{\text{Gph}} & \xrightarrow{(X, f)_{\text{Gph}}} & (X, H)_{\text{Gph}} \\ g & \mapsto & g(X, f)_{\text{Gph}} := gf . \end{array}$$

We have the map

$$\begin{array}{ccc} (H, Y)_{\text{Gph}} & \xrightarrow{(f, Y)_{\text{Gph}}} & (G, Y)_{\text{Gph}} \\ g & \mapsto & g(f, Y)_{\text{Gph}} := fg . \end{array}$$

Remark 69 Suppose given a graphs X, Y and graph morphisms $f : G \rightarrow H, g : H \rightarrow K$ and their composite $fg : G \rightarrow K$.

We have

- (1) $(X, fg)_{\text{Gph}} = (X, f)_{\text{Gph}} \cdot (X, g)_{\text{Gph}}$
 $(X, \text{id}_G)_{\text{Gph}} = \text{id}_{(X, G)_{\text{Gph}}}$
- (2) $(fg, Y)_{\text{Gph}} = (g, Y)_{\text{Gph}} \cdot (f, Y)_{\text{Gph}}$
 $(\text{id}_K, Y)_{\text{Gph}} = \text{id}_{(K, Y)_{\text{Gph}}}$.

Proof.

Ad (1).

Suppose given a graph morphism $u : X \rightarrow G$.

We have $u(X, fg)_{\text{Gph}} = u(fg) = (uf)g = (u(X, f)_{\text{Gph}}) \cdot g = u((X, f)_{\text{Gph}} \cdot (X, g)_{\text{Gph}})$.

We have $u(X, \text{id}_G)_{\text{Gph}} = u \text{id}_G = u$.

Ad (2).

Suppose given a graph morphism $v : K \rightarrow Y$.

We have $v(fg, Y)_{\text{Gph}} = (fg)v = f(gv) = (gv)(f, Y)_{\text{Gph}} = v((g, Y)_{\text{Gph}} \cdot (f, Y)_{\text{Gph}})$.

We have $v(\text{id}_K, Y)_{\text{Gph}} = \text{id}_K v = v$. □

Remark 70

(1) The cyclic graph C_1 is the terminal object in the category Gph .

Let τ_X be the unique graph morphism $\tau_X : X \rightarrow C_1$.

We have $v \tau_X = v_0$ for $v \in V_X$ and we have $e \tau_X = e_0$ for $e \in E_X$.

(2) The *empty graph* $\emptyset := (V_\emptyset, E_\emptyset)$ with $V_\emptyset := \emptyset$ and $E_\emptyset := \emptyset$ is the initial object in the category Gph .

Let ι_X be the unique graph morphism $\iota_X : \emptyset \rightarrow X$.

Definition 71 Suppose given a graph G .

We will define the *discrete subgraph* $\dot{G} \subseteq G$.

We let $V_{\dot{G}} := V_G$ and $E_{\dot{G}} := \emptyset$.

We have the maps $s_{\dot{G}} : \emptyset \rightarrow V_{\dot{G}}$ and $t_{\dot{G}} : \emptyset \rightarrow V_{\dot{G}}$.

We have the inclusion morphism o_G from \dot{G} to G as follows.

$$\begin{array}{ccc} \dot{G} & \xrightarrow{o_G} & G \\ V_{o_G} : V_{\dot{G}} & \rightarrow & V_G \\ & x \mapsto & x \\ E_{o_G} : \emptyset & \rightarrow & E_G \end{array}$$

Remark 72 Suppose given a surjective graph morphism $f : G \rightarrow H$.

Then $f : G \rightarrow H$ is an epimorphism in Gph .

Proof. Suppose given graph morphisms $u, u' : H \rightarrow K$ such that $fu = fu'$.

We have to show that $u \stackrel{!}{=} u'$. I.e. we have to show that $V_u \stackrel{!}{=} V_{u'}$ and $E_u \stackrel{!}{=} E_{u'}$.

We have $V_f V_u = V_{fu} = V_{fu'} = V_f V_{u'}$.

Since V_f is surjective, it is epimorphic; cf. Remark 31. So $V_u = V_{u'}$.

We have $E_f E_u = E_{fu} = E_{fu'} = E_f E_{u'}$.

Since E_f is surjective, it is epimorphic; cf. Remark 31. So $E_u = E_{u'}$.

So we conclude $u = u'$. □

2.2 Thin graphs

Definition 73 A graph G is called *thin* if for $v, v' \in V_G$ we have

$$|\{e \in E_G : es_G = v, et_G = v'\}| \leq 1.$$

I.e. between two vertices there is at most one edge.

Equivalently, a graph G is thin if the map

$$\begin{array}{ccc} \mathbf{E}_G & \xrightarrow{(s_G, t_G)} & \mathbf{V}_G \times \mathbf{V}_G \\ e & \mapsto & (e s_G, e t_G) \end{array}$$

is injective.

Example 74 The cyclic graph C_n is thin for $n \in \mathbb{N}$; cf. Definitions 52 and 73.

For $i, j \in \mathbb{Z}/n\mathbb{Z}$ we have

$$|\{e_k \in \mathbf{E}_{C_n} : e_k s_{C_n} = v_i, e_k t_{C_n} = v_j\}| = |\{e_i \in \mathbf{E}_{C_n} : e_i t_{C_n} = v_j\}| = \begin{cases} 1 & \text{if } j = i + 1 \\ 0 & \text{if } j \neq i + 1. \end{cases}$$

Example 75 The direct graph D_n is thin for $n \in \mathbb{N}$; cf. Definitions 56 and 73.

In fact, for $i, j \in [0, n]$ we have $|\{\hat{e}_k \in \mathbf{E}_{D_n} : \hat{e}_k s_{D_n} = \hat{v}_i, \hat{e}_k t_{D_n} = \hat{v}_j\}| = \begin{cases} 1 & \text{if } j = i + 1 \\ 0 & \text{if } j \neq i + 1. \end{cases}$

Remark 76 Suppose given a thin graph G . Suppose given a subgraph $G' \subseteq G$.

Then G' is thin.

Proof. Suppose given vertices $v, v' \in \mathbf{V}_{G'} \subseteq \mathbf{V}_G$.

We have

$$\begin{array}{ccc} & |\{e' \in \mathbf{E}_{G'} : e' s_{G'} = v, e' t_{G'} = v'\}| & \stackrel{\text{Def. 47.(1)}}{=} |\{e' \in \mathbf{E}_{G'} : e' s_G = v, e' t_G = v'\}| \\ \mathbf{E}_{G'} \subseteq \mathbf{E}_G & & \\ \leq & |\{e \in \mathbf{E}_G : e s_G = v, e t_G = v'\}| & \leq 1. \end{array}$$

□

Remark 77 Suppose given a graph X . Suppose given a thin graph Y .

(1) Suppose given graph morphisms $f, g : X \rightarrow Y$ with $\mathbf{V}_f = \mathbf{V}_g$.

Then $f = g$.

In other words, the map

$$\begin{array}{ccc} (X, Y)_{\text{Gph}} & \rightarrow & (\mathbf{V}_X, \mathbf{V}_Y)_{\text{Set}} \\ f & \mapsto & \mathbf{V}_f. \end{array}$$

is injective.

(2) A map $u : \mathbf{V}_X \rightarrow \mathbf{V}_Y$ is called *monotone* if for each edge $e \in \mathbf{E}_X$ there exists an edge $\tilde{e} \in \mathbf{E}_Y$ such that $\tilde{e} s_Y = (e s_X)u$ and $\tilde{e} t_Y = (e t_X)u$.

We have a bijective map

$$\begin{array}{ccc} (X, Y)_{\text{Gph}} & \rightarrow & \{u \in (\mathbf{V}_X, \mathbf{V}_Y)_{\text{Set}} : u \text{ is monotone}\} \\ f & \mapsto & \mathbf{V}_f. \end{array}$$

Proof.

Ad (1). We have to show that $E_f \stackrel{!}{=} E_g$.

The graph Y is thin. So we have $|\{e \in E_Y : e_{s_Y} = v, e_{t_Y} = v'\}| \leq 1$ for $v, v' \in V_Y$.

We have $E_f t_Y = t_X V_f = t_X V_g = E_g t_Y$ as well as $E_f s_Y = s_X V_f = s_X V_g = E_g s_Y$.

Suppose given an edge $e_X \in E_X$.

We have $v_1 := e_X E_f s_Y = e_X E_g s_Y \in V_Y$ and $v_2 := e_X E_f t_Y = e_X E_g t_Y \in V_Y$.

We have $1 \geq |\{e \in E_Y : e_{s_Y} = v_1, e_{t_Y} = v_2\}| \geq |\{e_X E_f, e_X E_g\}|$ because Y is thin.

So we have $e_X E_f = e_X E_g$.

Therefore, $E_f = E_g$.

Ad (2). By (1), the map in question is injective. It remains to show that it is surjective.

Suppose given a monotone map $u : V_X \rightarrow V_Y$.

We let $V_f := u$.

For $e \in E_X$, we let $e E_f := \tilde{e}$, where \tilde{e} is the unique edge in E_Y having $\tilde{e}_{s_Y} = (e_{s_X})u$ and $\tilde{e}_{t_Y} = (e_{t_X})u$.

Then $e E_f s_Y = \tilde{e}_{s_Y} = (e_{s_X})u = e_{s_X} V_f$ and $e E_f t_Y = \tilde{e}_{t_Y} = (e_{t_X})u = e_{t_X} V_f$.

So $f = (V_f, E_f)$ is a graph morphism with $V_f = u$. □

Example 78 Suppose given a graph X .

Suppose given $n \in \mathbb{N}$.

Suppose given graph morphisms $f, g : X \rightarrow C_n$ with $V_f = V_g$.

Then $f = g$, since the cyclic graph C_n is thin; cf. Remarks 74 and 77.(1).

Remark 79 Suppose given a thin graph Y .

Suppose given $n \in \mathbb{N}$.

Suppose given a map $u : V_{C_n} \rightarrow V_Y$ such that $Y(v_i u, v_{i+1} u) \neq \emptyset$ for $i \in \mathbb{Z}/n\mathbb{Z}$.

Then there exists a unique graph morphism $f : C_n \rightarrow Y$ such that $V_f = u$.

Proof.

Uniqueness. This follows by Remark 77.(1).

Existence. Write $Y(v_i u, v_{i+1} u) =: \{\alpha_i\}$ for $i \in \mathbb{Z}/n\mathbb{Z}$.

Let $V_f := u$.

Let

$$\begin{aligned} E_f : E_{C_n} &\rightarrow E_Y \\ e_i &\mapsto \alpha_i \end{aligned}$$

Then $f := (V_f, E_f) : C_n \rightarrow Y$ is a graph morphism, since $e_i E_f s_Y = \alpha_i s_Y = v_i u = e_i s_{C_n} V_f$ and $e_i E_f t_Y = \alpha_i t_Y = v_{i+1} u = e_i t_{C_n} V_f$ for $i \in \mathbb{Z}/n\mathbb{Z}$. Cf. also Remark 77.(2). □

Remark 80 Suppose given a thin graph G .

Suppose given a graph morphism $f : G \rightarrow H$ such that the map V_f is injective.

Then the graph morphism $f : G \rightarrow H$ is injective, i.e. both maps V_f and E_f are injective.

Proof. Suppose given edges e and \tilde{e} in E_G such that $e E_f = \tilde{e} E_f$.

We have to show that $e \stackrel{!}{=} \tilde{e}$.

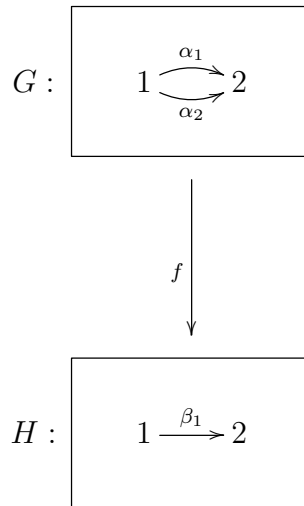
We have $e s_G V_f = e E_f s_H = \tilde{e} E_f s_H = \tilde{e} s_G V_f$.

Since the map $V_f : V_G \rightarrow V_H$ is injective, we have $e s_G = \tilde{e} s_G$.

Likewise we have $e t_G = \tilde{e} t_G$.

Since the graph G is thin, we conclude that $e = \tilde{e}$. □

Example 81 We consider the following graph morphism.



Here, $f = (V_f, E_f) : G \rightarrow H$ is the graph morphism mapping the vertices and the edges in a vertical way. I.e.

$$1 V_f = 1, \quad 2 V_f = 2$$

and

$$\alpha_1 E_f = \beta_1, \quad \alpha_2 E_f = \beta_1.$$

The map V_f is injective, while the map E_f is not injective. Note that the graph G is **not** thin.

2.3 Pushout and pullback of graphs

2.3.1 Pushout of graphs

Reminder 82 Suppose given a quadrangle

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Gph . It is called a pushout if (Pushout 1–2) hold; cf. Definition 5.

(Pushout 1) We have $g \cdot f' = f \cdot h$.

(Pushout 2) Suppose given a graph G and graph morphisms $u : X' \rightarrow G$ and $v : Y \rightarrow G$ such that $f \cdot v = g \cdot u$. Then there exists a unique graph morphism $w : Y' \rightarrow G$ such that $f' \cdot w = u$ and $h \cdot w = v$.

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array} \begin{array}{l} \searrow v \\ \downarrow \\ \exists! w \\ \downarrow \\ \searrow u \end{array} G$$

To indicate that this quadrangle is a pushout, we write

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array} \lrcorner$$

Then we also say that f' is a pushout of f .

Construction 83 Suppose given a diagram

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \\ X' & & \end{array}$$

in Gph .

We aim to construct a pushout

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}.$$

In particular, we have to construct maps $E_h : E_Y \rightarrow E_{Y'}$ and $E_{f'} : E_{X'} \rightarrow E_{Y'}$ and $V_h : V_Y \rightarrow V_{Y'}$ and $V_{f'} : V_{X'} \rightarrow V_{Y'}$.

We form pushouts in Set as follows.

$$\begin{array}{ccc} E_X & \xrightarrow{E_f} & E_Y \\ E_g \downarrow & & \downarrow E_h \\ E_{X'} & \xrightarrow{E_{f'}} & E_{Y'} \end{array} \qquad \begin{array}{ccc} V_X & \xrightarrow{V_f} & V_Y \\ V_g \downarrow & & \downarrow V_h \\ V_{X'} & \xrightarrow{V_{f'}} & V_{Y'} \end{array}$$

For instance, we can use Construction 36 to achieve this.

Because of the universal property of the pushout $E_{Y'}$, the map $t_{Y'}$ is uniquely existent with respect to $t_{X'} V_{f'} = E_{f'} t_{Y'}$ and $t_Y V_h = E_h t_{Y'}$.

Because of the universal property of the pushout $E_{Y'}$, the map $s_{Y'}$ is uniquely existent with respect to $s_{X'} V_{f'} = E_{f'} s_{Y'}$ and $s_Y V_h = E_h s_{Y'}$.

$$\begin{array}{ccccc} & & E_X & \xrightarrow{E_f} & E_Y \\ & E_g \swarrow & \parallel & & \swarrow E_h \\ E_{X'} & \xrightarrow{E_{f'}} & E_{Y'} & & \\ \parallel & \searrow t_X & \parallel & & \parallel t_Y \\ t_{X'} \downarrow & & t_{Y'} \downarrow & & \downarrow t_Y \\ & & V_X & \xrightarrow{V_f} & V_Y \\ & & \parallel & & \parallel \\ & & V_{X'} & \xrightarrow{V_{f'}} & V_{Y'} \\ & & \parallel & & \parallel \\ & & s_{X'} \downarrow & & \downarrow s_Y \\ & & s_{Y'} \downarrow & & \\ & & V_{X'} & \xrightarrow{V_{f'}} & V_{Y'} \end{array}$$

And so we have the pushout

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Gph.

Proof. We have $s_{X'} V_{f'} = E_{f'} s_{Y'}$ and $t_{X'} V_{f'} = E_{f'} t_{Y'}$.

So $f' = (V_{f'}, E_{f'})$ is a graph morphism from $X' = (V_{X'}, E_{X'}, s_{X'}, t_{X'})$ to $Y' = (V_{Y'}, E_{Y'}, s_{Y'}, t_{Y'})$.

We have $s_Y V_h = E_h s_{Y'}$ and $t_Y V_h = E_h t_{Y'}$.

So $h = (V_h, E_h)$ is a graph morphism from $Y = (V_Y, E_Y, s_Y, t_Y)$ to $Y' = (V_{Y'}, E_{Y'}, s_{Y'}, t_{Y'})$.

We have $gf' = (V_g, E_g) \cdot (V_{f'}, E_{f'}) = (V_g \cdot V_{f'}, E_g \cdot E_{f'}) = (V_{gf'}, E_{gf'}) = (V_{fh}, E_{fh}) = (V_f \cdot V_h, E_f \cdot E_h) = (V_f, E_f) \cdot (V_h, E_h) = fh$. \square

Universal property.

Suppose given a graph Z together with graph morphisms $v : X' \rightarrow Z$ and $u : Y \rightarrow Z$ such that $fu = gv$.

Since $E_{Y'}$ is constructed as a pushout, we obtain the unique map $E_w : E_{Y'} \rightarrow E_Z$ such that $E_h E_w = E_u$ and $E_{f'} E_w = E_v$.

Since $V_{Y'}$ is constructed as a pushout, we obtain the unique map $V_w : V_{Y'} \rightarrow V_Z$ such that $V_h V_w = V_u$ and $V_{f'} V_w = V_v$.

It remains to show that the pair of maps $w := (V_w, E_w)$ is in fact a graph morphism.

We have to show that the map w is a graph morphism.

So we have to show that $s_{Y'} V_w \stackrel{!}{=} E_w s_Z$ and that $t_{Y'} V_w \stackrel{!}{=} E_w t_Z$.

We have $E_{f'}(s_{Y'} V_w) = s_{X'} V_{f'} V_w = s_{X'} V_v = E_v s_Z = E_{f'}(E_w s_Z)$.

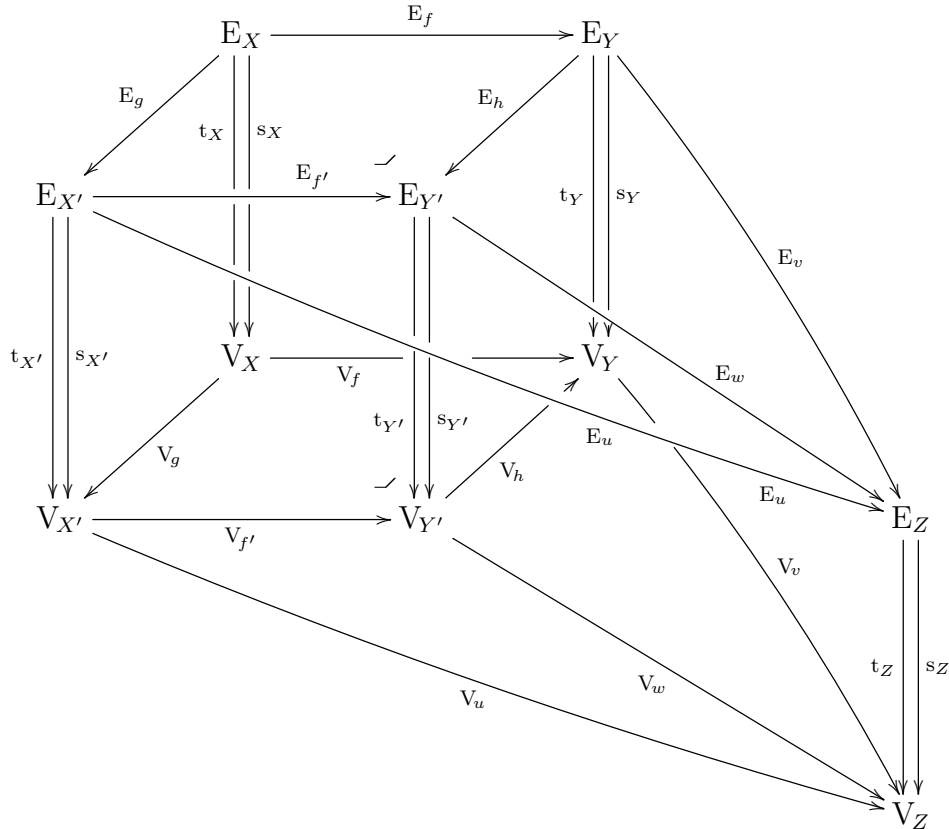
We have $E_h(s_{Y'} V_w) = s_Y V_h V_w = s_Y V_u = E_u s_Z = E_h(E_w s_Z)$.

This shows that $s_{Y'} V_w = E_w s_Z$ by Remark 6.

We have $E_{f'}(t_{Y'} V_w) = t_{X'} V_{f'} V_w = t_{X'} V_v = E_v t_Z = E_{f'}(E_w t_Z)$.

We have $E_h(t_{Y'} V_w) = t_Y V_h V_w = t_Y V_u = E_u t_Z = E_h(E_w t_Z)$.

This shows that $t_{Y'} V_w = E_w t_Z$ by Remark 6.



So we have obtained the following diagram in Gph.

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 \downarrow g & & \downarrow h \\
 X' & \xrightarrow{f'} & Y' \\
 & \searrow u & \downarrow w \\
 & & Z
 \end{array}$$

Remark 84 Suppose given

$$\begin{array}{ccc}
 X & \xrightarrow{a} & Y \\
 \downarrow f & & \\
 X' & &
 \end{array}$$

in Gph with an injective graph morphism $f : X \rightarrow X'$.

We let $V_{Y'} := (V_{X'} \setminus V_{Xf}) \sqcup V_Y$ and $E_{Y'} := (E_{X'} \setminus E_{Xf}) \sqcup E_Y$.

We let

$$\begin{array}{l}
 V_{a'} : V_{X'} \rightarrow V_{Y'} = (V_{X'} \setminus V_{Xf}) \sqcup V_Y \\
 v_{X'} \mapsto \begin{cases} (2, v_X V_a) & \text{if } v_{X'} = v_X V_f \in V_{Xf} \text{ for a unique } v_X \in V_X \\ (1, v_{X'}) & \text{if } v_{X'} \in V_{X'} \setminus V_{Xf} \end{cases}
 \end{array}$$

and

$$\begin{array}{l}
 E_{a'} : E_{X'} \rightarrow E_{Y'} = (E_{X'} \setminus E_{Xf}) \sqcup E_Y \\
 e_{X'} \mapsto \begin{cases} (2, e_X E_a) & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ (1, e_{X'}) & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} . \end{cases}
 \end{array}$$

We let

$$\begin{array}{l}
 V_g : V_Y \rightarrow V_{Y'} = (V_{X'} \setminus V_{Xf}) \sqcup V_Y \\
 v_Y \mapsto (2, v_Y)
 \end{array}$$

and

$$\begin{array}{l}
 E_g : E_Y \rightarrow E_{Y'} = (E_{X'} \setminus E_{Xf}) \sqcup E_Y \\
 e_Y \mapsto (2, e_Y)
 \end{array}$$

We have the pushouts

$$\begin{array}{ccc}
 V_X & \xrightarrow{V_a} & V_Y \\
 \downarrow V_f & & \downarrow V_g \\
 V_{X'} & \xrightarrow{V_{a'}} & V_{Y'}
 \end{array}$$

and

$$\begin{array}{ccc} E_X & \xrightarrow{E_a} & E_Y \\ E_f \downarrow & & \downarrow E_g \\ E_{X'} & \xrightarrow{E_{a'}} & E_{Y'} \end{array}$$

in Set; cf. Remark 38.

We will construct the maps $s_{Y'}$ and $t_{Y'}$.

Let

$$\begin{aligned} s_{Y'} : E_{Y'} = (E_{X'} \setminus E_{X'f}) \sqcup E_Y &\rightarrow (V_{X'} \setminus V_{X'f}) \sqcup V_Y = V_{Y'} \\ (1, e_{X'}) &\mapsto e_{X'} s_{X'} V_{a'} = \begin{cases} (2, v_X V_a) & \text{if } e_{X'} s_{X'} = v_X V_f \in V_{X'f} \\ & \text{for a unique } v_X \in V_X \\ (1, e_{X'} s_{X'}) & \text{if } e_{X'} s_{X'} \in V_{X'} \setminus V_{X'f} . \end{cases} \\ (2, e_Y) &\mapsto (2, e_Y s_Y) = e_Y s_Y V_g \end{aligned}$$

Let

$$\begin{aligned} t_{Y'} : E_{Y'} = (E_{X'} \setminus E_{X'f}) \sqcup E_Y &\rightarrow (V_{X'} \setminus V_{X'f}) \sqcup V_Y = V_{Y'} \\ (1, e_{X'}) &\mapsto e_{X'} t_{X'} V_{a'} = \begin{cases} (2, v_X V_a) & \text{if } e_{X'} t_{X'} = v_X V_f \in V_{X'f} \\ & \text{for a unique } v_X \in V_X \\ (1, e_{X'} t_{X'}) & \text{if } e_{X'} t_{X'} \in V_{X'} \setminus V_{X'f} . \end{cases} \\ (2, e_Y) &\mapsto (2, e_Y t_Y) = e_Y t_Y V_g \end{aligned}$$

In order to show that we have the pushout

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

it suffices to show that $E_{a'} s_{Y'} \stackrel{!}{=} s_{X'} V_{a'}$ and $E_g s_{Y'} \stackrel{!}{=} s_Y V_g$, and that $E_{a'} t_{Y'} \stackrel{!}{=} t_{X'} V_{a'}$ and $E_g t_{Y'} \stackrel{!}{=} t_Y V_g$; cf. Construction 83.

We show that $E_{a'} s_{Y'} \stackrel{!}{=} s_{X'} V_{a'}$ and $E_g s_{Y'} \stackrel{!}{=} s_Y V_g$.

For $e_{X'} \in E_{X'}$ we have $e_{X'} E_{a'} = \begin{cases} (2, e_X E_a) & \text{if } e_{X'} = e_X E_f \in E_{X'f} \text{ for a unique } e_X \in E_X \\ (1, e_{X'}) & \text{if } e_{X'} \in E_{X'} \setminus E_{X'f} . \end{cases}$

So we have

$$\begin{aligned} e_{X'} E_{a'} s_{Y'} &= \begin{cases} (2, e_X E_a) s_{Y'} & \text{if } e_{X'} = e_X E_f \in E_{X'f} \text{ for a unique } e_X \in E_X \\ (1, e_{X'}) s_{Y'} & \text{if } e_{X'} \in E_{X'} \setminus E_{X'f} \end{cases} \\ &= \begin{cases} (2, e_X E_a s_Y) & \text{if } e_{X'} = e_X E_f \in E_{X'f} \text{ for a unique } e_X \in E_X \\ e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{X'f} \end{cases} \end{aligned}$$

$$\begin{aligned}
&= \begin{cases} (2, e_X s_X V_a) & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_X s_X V_f V_{a'} & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_X E_f s_{X'} V_{a'} & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{Xf} \\ e_{X'} s_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= e_{X'} s_{X'} V_{a'}.
\end{aligned}$$

For $e_Y \in E_Y$ we have $e_Y E_g s_{Y'} = (2, e_Y) s_{Y'} = (2, e_Y s_Y) = e_Y s_Y V_g$.

We show that $E_{a'} t_{Y'} \stackrel{!}{=} t_{X'} V_{a'}$ and $E_g t_{Y'} \stackrel{!}{=} t_Y V_g$.

We have

$$\begin{aligned}
e_{X'} E_{a'} t_{Y'} &= \begin{cases} (2, e_X E_a) t_{Y'} & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ (1, e_{X'}) t_{Y'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} (2, e_X E_a t_Y) & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} (2, e_X t_X V_a) & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_X t_X V_f V_{a'} & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_X E_f t_{X'} V_{a'} & \text{if } e_{X'} = e_X E_f \in E_{Xf} \text{ for a unique } e_X \in E_X \\ e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= \begin{cases} e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{Xf} \\ e_{X'} t_{X'} V_{a'} & \text{if } e_{X'} \in E_{X'} \setminus E_{Xf} \end{cases} \\
&= e_{X'} t_{X'} V_{a'}.
\end{aligned}$$

For $e_Y \in E_Y$ we have $e_Y E_g t_{Y'} = (2, e_Y) t_{Y'} = (2, e_Y t_Y) = e_Y t_Y V_g$.

So we have the pushout

$$\begin{array}{ccc}
X & \xrightarrow{a} & Y \\
f \downarrow & & \downarrow g \\
X' & \xrightarrow{a'} & Y'
\end{array}$$

□

2.3.2 Coproducts

Definition 85 Suppose given graphs X and Y .

We will define the *coproduct* $X \sqcup Y$ of X and Y .

Note that the graph morphism $\iota_X : \emptyset \rightarrow X$ is injective.

By Construction 83 and Remark 84, we may form the pushout

$$\begin{array}{ccc} \emptyset & \xrightarrow{\iota_Y} & Y \\ \iota_X \downarrow & \lrcorner & \downarrow \iota_2 \\ X & \xrightarrow{\iota_1} & X \sqcup Y \end{array}$$

with $V_{X \sqcup Y} = (V_X \setminus V_{\emptyset \iota_X}) \sqcup V_Y = V_X \sqcup V_Y$ and $E_{X \sqcup Y} = (E_X \setminus E_{\emptyset \iota_X}) \sqcup E_Y = E_X \sqcup E_Y$ and with

$$\begin{aligned} s_{X \sqcup Y} : E_{X \sqcup Y} &\rightarrow V_{X \sqcup Y} \\ (1, e_X) &\mapsto (1, e_X s_X) \\ (2, e_Y) &\mapsto (2, e_Y s_Y) \end{aligned}$$

and

$$\begin{aligned} t_{X \sqcup Y} : E_{X \sqcup Y} &\rightarrow V_{X \sqcup Y} \\ (1, e_X) &\mapsto (1, e_X t_X) \\ (2, e_Y) &\mapsto (2, e_Y t_Y) . \end{aligned}$$

The graph $X \sqcup Y$ is called the *coproduct* of X and Y .

And we have

$$\begin{aligned} \iota_1 : X &\rightarrow X \sqcup Y \\ V_{\iota_1} : V_X &\rightarrow V_{X \sqcup Y} = V_X \sqcup V_Y \\ v_X &\mapsto (1, v_X) \\ E_{\iota_1} : E_X &\rightarrow E_{X \sqcup Y} = E_X \sqcup E_Y \\ e_X &\mapsto (1, e_X) \end{aligned}$$

and

$$\begin{aligned} \iota_2 : Y &\rightarrow X \sqcup Y \\ V_{\iota_2} : V_Y &\rightarrow V_{X \sqcup Y} = V_X \sqcup V_Y \\ v_Y &\mapsto (2, v_Y) \\ E_{\iota_2} : E_Y &\rightarrow E_{X \sqcup Y} = E_X \sqcup E_Y \\ e_Y &\mapsto (2, e_Y) \end{aligned}$$

Given a graph morphism $g : X \sqcup Y \rightarrow G$, we also write $g|_X := \iota_1 g$ and $g|_Y := \iota_2 g$.

The following definition is a way to express the universal property of the coproduct.

Definition 86 Suppose given graph morphisms $a : X \rightarrow G$ and $b : Y \rightarrow G$.

We have the following commutative diagram in Gph ; cf. Remminder 82.

$$\begin{array}{ccc} \emptyset & \xrightarrow{\iota_X} & X \\ \iota_Y \downarrow & \lrcorner & \downarrow \iota_1 \\ Y & \xrightarrow{\iota_2} & X \sqcup Y \end{array} \begin{array}{c} \searrow a \\ \downarrow \exists! \begin{pmatrix} a \\ b \end{pmatrix} \\ \searrow b \end{array} \rightarrow G$$

Since $\iota_1 \begin{pmatrix} a \\ b \end{pmatrix} = a$ and since $\iota_2 \begin{pmatrix} a \\ b \end{pmatrix} = b$, the graph morphism $\begin{pmatrix} a \\ b \end{pmatrix} : X \sqcup Y \rightarrow G$ maps as follows.

$$\begin{aligned} V_{\begin{pmatrix} a \\ b \end{pmatrix}} : \quad & V_{X \sqcup Y} \rightarrow V_G \\ (1, v_X) = v_X V_{\iota_1} & \mapsto v_X V_a \\ (2, v_Y) = v_Y V_{\iota_2} & \mapsto v_Y V_b \end{aligned}$$

$$\begin{aligned} E_{\begin{pmatrix} a \\ b \end{pmatrix}} : \quad & E_{X \sqcup Y} \rightarrow E_G \\ (1, e_X) = e_X E_{\iota_1} & \mapsto e_X E_a \\ (2, e_Y) = e_Y E_{\iota_2} & \mapsto e_Y E_b \end{aligned}$$

□

Definition 87 Suppose given graph morphisms $f : X \rightarrow X'$ and $g : Y \rightarrow Y'$.

We have the graph morphisms $\iota_1 : X \rightarrow X \sqcup Y$, $\iota_2 : Y \rightarrow X \sqcup Y$, $\iota'_1 : X' \rightarrow X' \sqcup Y'$ and $\iota'_2 : Y' \rightarrow X' \sqcup Y'$; cf. Definition 85.

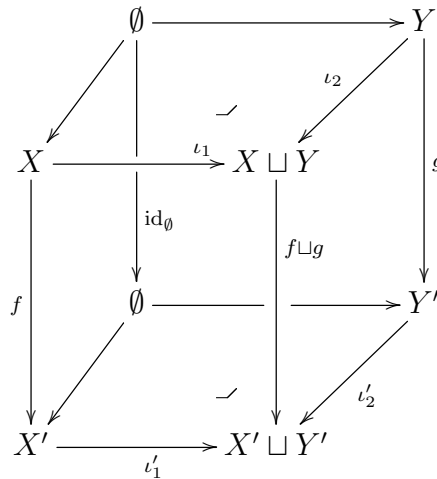
Then we have the graph morphism $f \sqcup g := \begin{pmatrix} f \iota'_1 \\ g \iota'_2 \end{pmatrix} : X \sqcup Y \rightarrow X' \sqcup Y'$; cf. Definition 86.

So we have $\iota_1(f \sqcup g) = f \iota'_1$ and $\iota_2(f \sqcup g) = g \iota'_2$.

We have

$$\begin{aligned} V_{f \sqcup g} : \quad & V_{X \sqcup Y} \rightarrow V_{X' \sqcup Y'} \\ (1, v_X) & \mapsto v_X V_{\iota_1} V_{f \sqcup g} = v_X V_f V_{\iota'_1} = (1, v_X V_f) \\ (2, v_Y) & \mapsto v_Y V_{\iota_2} V_{f \sqcup g} = v_Y V_g V_{\iota'_2} = (2, v_Y V_g) \end{aligned}$$

$$\begin{aligned} E_{f \sqcup g} : \quad & E_{X \sqcup Y} \rightarrow E_{X' \sqcup Y'} \\ (1, e_X) & \mapsto e_X E_{\iota_1} E_{f \sqcup g} = e_X E_f E_{\iota'_1} = (1, e_X E_f) \\ (2, e_Y) & \mapsto e_Y E_{\iota_2} E_{f \sqcup g} = e_Y E_g E_{\iota'_2} = (2, e_Y E_g) \end{aligned}$$



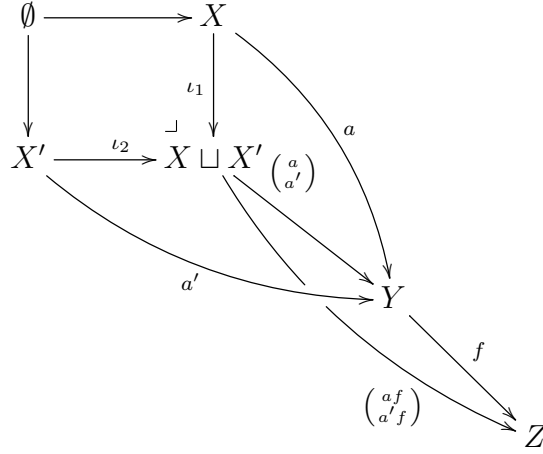
Remark 88 Suppose graph morphisms $a : X \rightarrow Y$, $a' : X' \rightarrow Y$ and $f : Y \rightarrow Z$.

We have the graph morphism $\binom{a}{a'} : X \sqcup X' \rightarrow Y$; cf. Definition 86.

We have $\binom{a}{a'} f = \binom{af}{a'f}$.

Proof. Recall the graph morphisms $\iota_1 : X \rightarrow X \sqcup X'$ and $\iota_2 : X' \rightarrow X \sqcup X'$ from Definition 86.

Since we have $\iota_1 \binom{a}{a'} f = af = \iota_1 \binom{af}{a'f}$ and since we have $\iota_2 \binom{a}{a'} f = a'f = \iota_2 \binom{af}{a'f}$, we have $\binom{a}{a'} f = \binom{af}{a'f}$. □



Remark 89 Suppose given graph morphisms $X \xrightarrow{f} X' \xrightarrow{a'} G$ and $Y \xrightarrow{g} Y' \xrightarrow{b'} G$.

We have $(f \sqcup g) \binom{a'}{b'} = \binom{fa'}{gb'}$.

Proof. Recall the graph morphisms $\iota_1 : X \rightarrow X \sqcup Y$, $\iota'_1 : X' \rightarrow X' \sqcup Y'$, $\iota_2 : Y \rightarrow X \sqcup Y$ and $\iota'_2 : Y' \rightarrow X' \sqcup Y'$ from Definition 86.

We have $f \sqcup g = \binom{f\iota'_1}{g\iota'_2}$; cf. Definition 87.

So we have to show that $(f \sqcup g) \binom{a'}{b'} = \binom{f\iota'_1}{g\iota'_2} \binom{a'}{b'} \stackrel{!}{=} \binom{fa'}{gb'}$.

It suffices to show that $\iota_1(f \sqcup g) \binom{a'}{b'} \stackrel{!}{=} \iota_1 \binom{fa'}{gb'}$ and that $\iota_2(f \sqcup g) \binom{a'}{b'} \stackrel{!}{=} \iota_2 \binom{fa'}{gb'}$.

We have $\iota_1(f \sqcup g) \binom{a'}{b'} = \iota_1 \binom{f\iota'_1}{g\iota'_2} \binom{a'}{b'} = f\iota'_1 \binom{a'}{b'} = fa' = \iota_1 \binom{fa'}{gb'}$.

We have $\iota_2(f \sqcup g) \binom{a'}{b'} = \iota_2 \binom{f\iota'_1}{g\iota'_2} \binom{a'}{b'} = g\iota'_2 \binom{a'}{b'} = gb' = \iota_2 \binom{fa'}{gb'}$. □

Now we generalize from the coproduct of two graphs to coproducts indexed with a set.

Definition 90 Suppose given a set I .

Suppose given graphs A_i for $i \in I$.

We define the *coproduct* $\coprod_{i \in I} A_i$ of the graphs A_i as follows.

We let $V_{\coprod_{i \in I} A_i} := \prod_{i \in I} V_{A_i}$ and $E_{\coprod_{i \in I} A_i} := \prod_{i \in I} E_{A_i}$; cf. Definition 32.

Additionally we let

$$\begin{aligned} s_{\coprod_{i \in I} A_i} : E_{\coprod_{i \in I} A_i} &\rightarrow V_{\coprod_{i \in I} A_i} \\ (i, e) &\mapsto (i, e s_{A_i}) \text{ for } i \in I \end{aligned}$$

and

$$\begin{aligned} t_{\coprod_{i \in I} A_i} : E_{\coprod_{i \in I} A_i} &\rightarrow V_{\coprod_{i \in I} A_i} \\ (i, e) &\mapsto (i, e t_{A_i}) \text{ for } i \in I. \end{aligned}$$

We have the inclusion graph morphism $\iota_k : A_k \rightarrow \coprod_{i \in I} A_i$ for $k \in I$ as follows.

$$\begin{aligned} \iota_k : A_k &\rightarrow \coprod_{i \in I} A_i \\ V_{\iota_k} : V_{A_k} &\rightarrow V_{\coprod_{i \in I} A_i} \\ v_k &\rightarrow (k, v_k) \\ E_{\iota_k} : E_{A_k} &\rightarrow E_{\coprod_{i \in I} A_i} \\ e_k &\rightarrow (k, e_k) \end{aligned}$$

Suppose given an edge $e_k \in A_k$.

We have $e_k E_{\iota_k} s_{\coprod_{i \in I} A_i} = (k, e_k) s_{\coprod_{i \in I} A_i} = (k, e_k s_{A_k}) = e_k s_{A_k} V_{\iota_k}$.

We have $e_k E_{\iota_k} t_{\coprod_{i \in I} A_i} = (k, e_k) t_{\coprod_{i \in I} A_i} = (k, e_k t_{A_k}) = e_k t_{A_k} V_{\iota_k}$.

So the pair of maps $\iota_k = (V_{\iota_k}, E_{\iota_k})$ is in fact a graph morphism since we have $E_{\iota_k} s_{\coprod_{i \in I} A_i} = s_{A_k} V_{\iota_k}$ and $E_{\iota_k} t_{\coprod_{i \in I} A_i} = t_{A_k} V_{\iota_k}$.

Definition 91 Let I be a set.

Suppose given graph morphisms $f_i : X_i \rightarrow Y$ for $i \in I$.

We have the graph morphism $f =: (f_i)_{i \in I} : \prod_{i \in I} X_i \rightarrow Y$ with

$$\begin{aligned} (i, v_{X_i}) V_f &:= v_{X_i} V_{f_i} \text{ for } i \in I \text{ and } v_{X_i} \in V_{X_i} \\ (i, e_{X_i}) E_f &:= e_{X_i} E_{f_i} \text{ for } i \in I \text{ and } e_{X_i} \in E_{X_i}; \end{aligned}$$

cf. also Definition 86.

We often abbreviate $(f_i)_i := (f_i)_{i \in I}$.

The pair of maps (V_f, E_f) is in fact a graph morphism, since $(i, e_{X_i}) E_f s_Y = e_{X_i} E_{f_i} s_Y = e_{X_i} s_{X_i} V_{f_i} = (i, e_{X_i} s_{X_i}) V_f = (i, e_{X_i}) s_{\prod_{i \in I} X_i} V_f$ and $(i, e_{X_i}) E_f t_Y = e_{X_i} E_{f_i} t_Y = e_{X_i} t_{X_i} V_{f_i} = (i, e_{X_i} t_{X_i}) V_f = (i, e_{X_i}) t_{\prod_{i \in I} X_i} V_f$.

Note that $\iota_j f = \iota_j (f_i)_{i \in I} = f_j$ for $j \in I$ and that f is unique with this property; cf. Definition 90.

Remark 92 Suppose given a set I .

Suppose graph morphisms $a_i : X_i \rightarrow Y$ for $i \in I$ and $f : Y \rightarrow Z$.

We have the graph morphism $(a_i)_i : \prod_{i \in I} X_i \rightarrow Y$; cf. Definition 91.

We have $(a_i)_i \cdot f = (a_i \cdot f)_i$.

Proof. Recall the graph morphism $\iota_k : X_k \rightarrow \coprod_{i \in I} X_i$ for $k \in I$ from Definition 91.

Since we have $\iota_k \cdot (a_i)_i \cdot f = a_k \cdot f = \iota_k \cdot (a_i \cdot f)_i$ for $k \in I$, we have $(a_i)_i \cdot f = (a_i \cdot f)_i$.

Definition 93 Suppose given a set I .

Suppose given graph morphisms $f_i : X_i \rightarrow Y_i$.

Recall the graph morphisms $\iota_k : X_k \rightarrow \coprod_{i \in I} X_i$ and $\iota_k : Y_k \rightarrow \coprod_{i \in I} Y_i$ for $k \in I$, from Definition 91.

We define the graph morphism $\coprod_{i \in I} f_i := (f_i \cdot \iota_i)_{i \in I} : \coprod_{i \in I} X_i \rightarrow \coprod_{i \in I} Y_i$.

Then we have $\iota_k \cdot \coprod_{i \in I} f_i = \iota_k \cdot (f_i \cdot \iota_i)_{i \in I} = f_k \cdot \iota_k$ for $k \in I$.

Remark 94 Suppose given a set I .

Suppose given graph morphisms $X_i \xrightarrow{f_i} Y_i \xrightarrow{a_i} G$ for $i \in I$.

We have $(\coprod_{i \in I} f_i) \cdot (a_i)_i = (f_i \cdot a_i)_i$.

Proof. Recall the graph morphisms $\iota_k : Y_k \rightarrow \coprod_{i \in I} Y_i$ for $k \in I$ from Definition 91.

We have $(\coprod_j f_j) \cdot (a_i)_i \stackrel{\text{Def. 93}}{=} (f_j \cdot \iota_j)_j \cdot (a_i)_i \stackrel{\text{Rem. 92}}{=} (f_j \cdot \iota_j \cdot (a_i)_i)_j \stackrel{\text{Def. 91}}{=} (f_j \cdot a_j)_j$. \square

Remark 95 Suppose given a set I .

Suppose given a graph morphism $f : G \rightarrow H$ and graphs X_i for $i \in I$.

We have the following commutative diagram

$$\begin{array}{ccc} (\coprod_{i \in I} X_i, G)_{\text{Gph}} & \xleftarrow{\sim a} & \prod_{i \in I} (X_i, G)_{\text{Gph}} \\ (\coprod_{i \in I} X_i, f)_{\text{Gph}} \downarrow & & \downarrow \prod_{i \in I} (X_i, f)_{\text{Gph}} \\ (\coprod_{i \in I} X_i, H)_{\text{Gph}} & \xleftarrow{\sim b} & \prod_{i \in I} (X_i, H)_{\text{Gph}} \end{array}$$

in Set with

$$\begin{aligned} a : \prod_{i \in I} (X_i, G)_{\text{Gph}} &\xrightarrow{\sim} (\coprod_{i \in I} X_i, G)_{\text{Gph}} \\ (g_i)_{i \in I} &\mapsto (g_i)_{i \in I} \end{aligned}$$

and

$$\begin{aligned} b : \prod_{i \in I} (X_i, H)_{\text{Gph}} &\xrightarrow{\sim} (\coprod_{i \in I} X_i, H)_{\text{Gph}} \\ (h_i)_{i \in I} &\mapsto (h_i)_{i \in I} . \end{aligned}$$

In particular, $(\coprod_{i \in I} X_i, f)_{\text{Gph}}$ is bijective if and only if $\prod_{i \in I} (X_i, f)_{\text{Gph}}$ is bijective.

Proof. Because of the universal property of the coproduct, the maps a and b are bijective and we have

$$\begin{aligned} a^{-1} : (\coprod_{i \in I} X_i, G)_{\text{Gph}} &\xrightarrow{\sim} \prod_{i \in I} (X_i, G)_{\text{Gph}} \\ g &\mapsto (\iota_i g)_i \end{aligned}$$

and

$$b^{-1} : \left(\coprod_{i \in I} X_i, H \right)_{\text{Gph}} \xrightarrow{\sim} \prod_{i \in I} (X_i, H)_{\text{Gph}}$$

$$h \mapsto (\iota_i h)_i .$$

So we have

$$\left(\coprod_{i \in I} X_i, G \right)_{\text{Gph}} \xrightarrow{a^{-1}} \prod_{i \in I} (X_i, G)_{\text{Gph}} \xrightarrow{\prod_{i \in I} (X_i, f)_{\text{Gph}}} \prod_{i \in I} (X_i, H)_{\text{Gph}} \xrightarrow{b} \left(\coprod_{i \in I} X_i, H \right)_{\text{Gph}}$$

$$g \quad \mapsto \quad (\iota_i g)_i \quad \mapsto \quad (\iota_i g f)_i \quad \mapsto \quad g f .$$

So we have $\left(\coprod_{i \in I} X_i, f \right)_{\text{Gph}} = a^{-1} \cdot \prod_{i \in I} (X_i, f)_{\text{Gph}} \cdot b$. □

2.3.3 Pullback of graphs

Reminder 96 Suppose given a quadrangle

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Gph . It is called a pullback if (Pullback 1–2) hold; cf. Definition 9.

(Pullback 1) We have $f \cdot h = g \cdot f'$.

(Pullback 2) Suppose given a graph G and graph morphisms $u : G \rightarrow Y$ and $v : G \rightarrow X'$ such that $u \cdot h = v \cdot f'$. Then there exists a unique graph morphism $w : G \rightarrow X$ such that $w \cdot f = u$ and $w \cdot g = v$.

$$\begin{array}{ccccc} & & G & & \\ & & \curvearrowright u & & \\ & & & & \\ & & \exists! w & & \\ & & \searrow & & \\ & & X & \xrightarrow{f} & Y \\ & & \downarrow g & & \downarrow h \\ & & X' & \xrightarrow{f'} & Y' \\ & & \swarrow v & & \end{array}$$

To indicate that this quadrangle is a pullback, we write

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & \lrcorner & \downarrow h \\ X' & \xrightarrow{f'} & Y' . \end{array}$$

Then we also say that f is a pullback of f' .

Construction 97 Suppose given a diagram

$$\begin{array}{ccc} & & Y \\ & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Gph.

We aim to construct a pullback

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & \lrcorner & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

In particular, we have to construct maps $E_g : E_X \rightarrow E_{X'}$ and $E_f : E_X \rightarrow E_Y$ and $V_g : V_X \rightarrow V_{X'}$ and $V_f : V_X \rightarrow V_Y$.

We construct the pullbacks

$$\begin{array}{ccc} E_X & \xrightarrow{E_f} & E_Y \\ E_g \downarrow & \lrcorner & \downarrow E_h \\ E_{X'} & \xrightarrow{E_{f'}} & E_{Y'} \end{array}$$

and

$$\begin{array}{ccc} V_X & \xrightarrow{V_f} & V_Y \\ V_g \downarrow & \lrcorner & \downarrow V_h \\ V_{X'} & \xrightarrow{V_{f'}} & V_{Y'} \end{array}$$

using Construction 41.

Because of the universal property of the pullback E_X , the map t_X is uniquely existent with respect to $t_X V_g = E_g t_{X'}$ and $t_X V_f = E_f t_Y$.

Because of the universal property of the pullback E_X , the map s_X is uniquely existent with respect to $s_X V_g = E_g s_{X'}$ and $s_X V_f = E_f s_Y$.

$$\begin{array}{ccccc} & & E_X & \xrightarrow{E_f} & E_Y \\ & & \parallel & & \parallel \\ E_{X'} & \xrightarrow{E_{f'}} & E_{Y'} & & E_Y \\ & & \parallel & & \parallel \\ & & E_{X'} & \xrightarrow{E_{f'}} & E_{Y'} \\ & & \parallel & & \parallel \\ & & V_X & \xrightarrow{V_f} & V_Y \\ & & \parallel & & \parallel \\ & & V_{X'} & \xrightarrow{V_{f'}} & V_{Y'} \end{array}$$

Additional arrows in the diagram include: E_g from E_X to $E_{X'}$, E_h from E_Y to $E_{Y'}$, t_X from E_X to V_X , s_X from E_X to V_X , $t_{X'}$ from $E_{X'}$ to $V_{X'}$, $s_{X'}$ from $E_{X'}$ to $V_{X'}$, t_Y from E_Y to V_Y , s_Y from E_Y to V_Y , $t_{Y'}$ from $E_{Y'}$ to $V_{Y'}$, $s_{Y'}$ from $E_{Y'}$ to $V_{Y'}$, V_g from V_X to $V_{X'}$, V_h from V_Y to $V_{Y'}$, and V_f from V_X to V_Y .

And so we have the pullback

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 \downarrow g & \lrcorner & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

in Gph .

Proof. We have $s_X V_g = E_g s_{X'}$ and $t_X V_g = E_g t_{X'}$.

So $g = (V_g, E_g)$ is a graph morphism from $X = (V_X, E_X, s_X, t_X)$ to $X' = (V_{X'}, E_{X'}, s_{X'}, t_{X'})$.

We have $s_X V_f = E_f s_Y$ and $t_X V_f = E_f t_Y$.

So $f = (V_f, E_f)$ is a graph morphism from $X = (V_X, E_X, s_X, t_X)$ to $Y = (V_Y, E_Y, s_Y, t_Y)$.

We have $gf' = (V_g, E_g) \cdot (V_{f'}, E_{f'}) = (V_g \cdot V_{f'}, E_g \cdot E_{f'}) = (V_{gf'}, E_{gf'}) = (V_{fh}, E_{fh}) = (V_f \cdot V_h, E_f \cdot E_h) = (V_f, E_f) \cdot (V_h, E_h) = fh$. \square

Universal property.

Suppose given a graph Z together with graph morphisms $u : Z \rightarrow Y$ and $v : Z \rightarrow X'$ such that $uh = vf'$.

Since E_X is constructed as a pullback, we obtain the unique map $E_w : E_Z \rightarrow E_X$ such that $E_w E_f = E_u$ and $E_w E_g = E_v$.

Since V_X is constructed as a pullback, we obtain the unique map $V_w : V_Z \rightarrow V_X$ such that $V_w V_f = V_u$ and $V_w V_g = V_v$.

It remains to show that the pair of maps $w := (V_w, E_w)$ is in fact a graph morphism.

We have to show that the map w is a graph morphism.

So we have to show that $s_Z V_w \stackrel{!}{=} E_w s_X$ and that $t_Z V_w \stackrel{!}{=} E_w t_X$.

We have $(s_Z V_w) V_g = s_Z V_v = E_v s_{X'} = E_w E_g s_{X'} = (E_w s_X) V_g$.

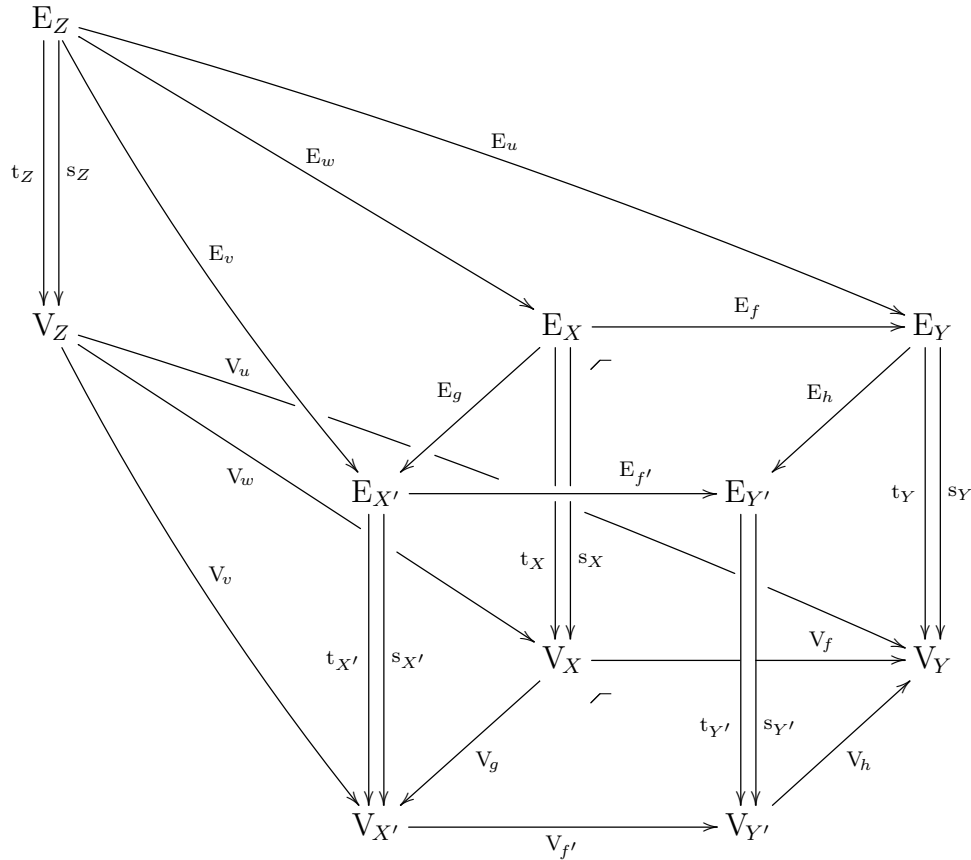
We have $(s_Z V_w) V_f = s_Z V_u = E_u s_Y = E_w E_f s_Y = (E_w s_X) V_f$.

This shows that $s_Z V_w = E_w s_X$ by Remark 10.

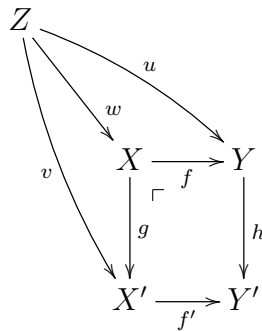
We have $(t_Z V_w) V_g = t_Z V_v = E_v t_{X'} = E_w E_g t_{X'} = (E_w t_X) V_g$.

We have $(t_Z V_w) V_f = t_Z V_u = E_u t_Y = E_w E_f t_Y = (E_w t_X) V_f$.

This shows that $t_Z V_w = E_w t_X$ by Remark 10.



So we have the following commutative diagram in Gph.



2.4 Colimit of a countable chain in Gph

Definition 98 Suppose given

$$X_0 \xrightarrow{f_{0,1}} X_1 \xrightarrow{f_{1,2}} X_2 \xrightarrow{f_{2,3}} \dots$$

in Gph.

We define

$$V_{X_\infty} := \varinjlim_{i \geq 0} V_{X_i} = \{[j, v_j] : j \geq 0, v_j \in V_{X_j}\}$$

and

$$\mathbf{E}_{X_\infty} := \varinjlim_{i \geq 0} \mathbf{E}_{X_i} = \{[j, e_j] : j \geq 0, e_j \in \mathbf{E}_{X_j}\};$$

cf. Definition 42.

Note that $[j, v_j] = [k, v_j V_{f_{j,k}}]$ for $0 \leq j \leq k$ and $v_j \in V_{X_j}$.

Note that $[j, e_j] = [k, e_j E_{f_{j,k}}]$ for $0 \leq j \leq k$ and $e_j \in \mathbf{E}_{X_j}$.

For $k \geq 0$, Definition 42.(2) gives the maps

$$\begin{aligned} V_{f_{k,\infty}} : V_{X_k} &\mapsto V_{X_\infty} \\ &v_k \mapsto [k, v_k] \\ E_{f_{k,\infty}} : \mathbf{E}_{X_k} &\mapsto \mathbf{E}_{X_\infty} \\ &e_k \mapsto [k, e_k]. \end{aligned}$$

We define $s_{X_\infty} := \varinjlim_{i \geq 0} s_{X_i}$ and $t_{X_\infty} := \varinjlim_{i \geq 0} t_{X_i}$, which is possible since $s_{X_i} V_{f_{i,i+1}} = E_{f_{i,i+1}} s_{X_{i+1}}$ and $t_{X_i} V_{f_{i,i+1}} = E_{f_{i,i+1}} t_{X_{i+1}}$ for $i \geq 0$; cf. Definition 44.

We have

$$\begin{aligned} s_{X_\infty} : \mathbf{E}_{X_\infty} &\rightarrow V_{X_\infty} \\ [k, e_k] &\mapsto [k, e_k s_{X_k}] \end{aligned}$$

and

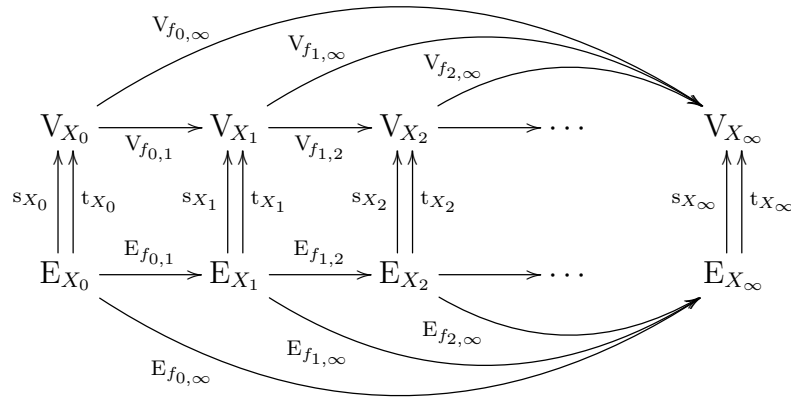
$$\begin{aligned} t_{X_\infty} : \mathbf{E}_{X_\infty} &\rightarrow V_{X_\infty} \\ [k, e_k] &\mapsto [k, e_k t_{X_k}]. \end{aligned}$$

In fact, we have $[k, e_k] s_{X_\infty} = e_k E_{f_{k,\infty}} s_{X_\infty} \stackrel{\text{Def. 44}}{=} e_k s_{X_k} V_{f_{k,\infty}} = [k, e_k s_{X_k}]$ and $[k, e_k] t_{X_\infty} = e_k E_{f_{k,\infty}} t_{X_\infty} \stackrel{\text{Def. 44}}{=} e_k t_{X_k} V_{f_{k,\infty}} = [k, e_k t_{X_k}]$.

This defines the graph $X_\infty := (V_{X_\infty}, \mathbf{E}_{X_\infty}, s_{X_\infty}, t_{X_\infty})$.

For $j \geq 0$ we have $E_{f_{j,\infty}} s_{X_\infty} = s_{X_j} V_{f_{j,\infty}}$ and $E_{f_{j,\infty}} t_{X_\infty} = t_{X_j} V_{f_{j,\infty}}$; cf. Definition 44.

So the pair of maps $f_{j,\infty} = (V_{f_{j,\infty}}, E_{f_{j,\infty}}) : X_j \rightarrow X_\infty$ is a graph morphism.



We have $f_{i,i+1} f_{i+1,\infty} = f_{i,\infty}$; cf. Definition 42.(2), applied to vertices and edges.

We have the following commutative diagram.

$$\begin{array}{ccccccc}
 E_{X_0} & \xrightarrow{E_{f_{0,1}}} & E_{X_1} & \xrightarrow{E_{f_{1,2}}} & E_{X_2} & \xrightarrow{E_{f_{2,3}}} & \dots \\
 & \searrow E_{t_0} & \downarrow E_{t_1} & \swarrow E_{t_2} & \dots & & \\
 & & E_T & & & &
 \end{array}$$

So because of the universal property of E_{X_∞} there exists a unique map $E_{t_\infty} : E_{X_\infty} \rightarrow E_T$ such that the following diagram is commutative.

$$\begin{array}{ccccccc}
 & & & & & & E_{X_\infty} \\
 & & \text{E}_{f_{0,\infty}} & & \text{E}_{f_{1,\infty}} & & \text{E}_{f_{2,\infty}} \\
 & & \text{---} & & \text{---} & & \text{---} \\
 E_{X_0} & \xrightarrow{E_{f_{0,1}}} & E_{X_1} & \xrightarrow{E_{f_{1,2}}} & E_{X_2} & \xrightarrow{E_{f_{2,3}}} & \dots \\
 & \searrow E_{t_0} & \downarrow E_{t_1} & \swarrow E_{t_2} & \dots & & \\
 & & E_T & & & & \\
 & & & & & & \text{E}_{t_\infty} \\
 & & & & & & \text{---}
 \end{array}$$

We let

$$\begin{aligned}
 t_\infty : X_\infty &\rightarrow T \\
 V_{t_\infty} : V_{X_\infty} &\mapsto V_T \\
 [j, v_j] &\mapsto v_j V_{t_j} \\
 E_{t_\infty} : E_{X_\infty} &\mapsto E_T \\
 [j, e_j] &\mapsto e_j E_{t_j}
 \end{aligned}$$

We show that these maps are well-defined.

Suppose given $(j, v_j), (k, v_k) \in \bigsqcup_{i \geq 0} V_{X_i}$ such that $[j, v_j] = [k, v_k]$.

Then $v_j V_{f_{j,m}} = v_k V_{f_{k,m}}$ for some $m \geq \max\{j, k\}$.

We have to show that $v_j V_{t_j} \stackrel{!}{=} v_k V_{t_k}$.

In fact, we have $v_j V_{t_j} = v_j V_{f_{j,m} t_m} = v_k V_{f_{k,m} t_m} = v_k V_{t_k}$.

Suppose given $v_j \in V_{X_j}$. For $j \geq 0$ we have $v_j V_{f_{j,\infty}} V_{t_\infty} = [j, v_j] V_{t_\infty} \stackrel{\text{Def. } V_{t_\infty}}{=} v_j V_{t_j}$.

So we have $V_{f_{j,\infty}} V_{t_\infty} = V_{t_j}$.

Suppose given $(j, e_j), (k, e_k) \in \bigsqcup_{i \geq 0} E_{X_i}$ such that $[j, e_j] = [k, e_k]$.

Then $e_j E_{f_{j,m}} = e_k E_{f_{k,m}}$ for some $m \geq \max\{j, k\}$.

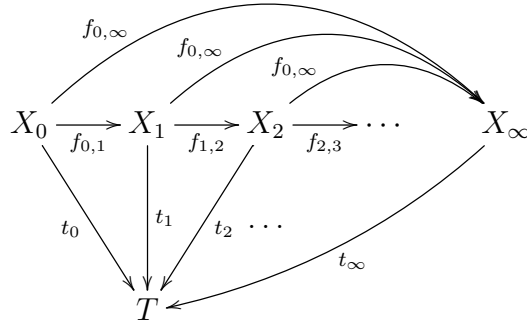
We have to show that $e_j E_{t_j} \stackrel{!}{=} e_k E_{t_k}$.

In fact, we have $e_j E_{t_j} = e_j E_{f_{j,m} t_m} = e_k E_{f_{k,m} t_m} = e_k E_{t_k}$.

Suppose given $e_j \in E_{X_j}$. For $j \geq 0$ we have $e_j E_{f_{j,\infty}} E_{t_\infty} = [j, e_j] E_{t_\infty} \stackrel{\text{Def. } E_{t_\infty}}{=} e_j E_{t_j}$.

So we have $E_{f_{j,\infty}} E_{t_\infty} = E_{t_j}$.

To show that $t_\infty := (V_{t_\infty}, E_{t_\infty})$ is the unique graph morphism such that



is commutative, it remains to show that t_∞ actually is a graph morphism.

We have to show that $E_{t_\infty} s_T \stackrel{!}{=} s_{X_\infty} V_{t_\infty}$ and that $E_{t_\infty} t_T \stackrel{!}{=} t_{X_\infty} V_{t_\infty}$.

Suppose given an edge $e_\infty = [j, e_j] \in E_{X_\infty}$ with $j \geq 0$ and $e_j \in E_{X_j}$.

We have $e_\infty E_{t_\infty} s_T = [j, e_j] E_{t_\infty} s_T = e_j E_{t_j} s_T = e_j s_{X_j} V_{t_j} = [j, e_j s_{X_j}] V_{t_\infty} = [j, e_j] s_{X_\infty} V_{t_\infty} = e_\infty s_{X_\infty} V_{t_\infty}$.

And we have $e_\infty E_{t_\infty} t_T = [j, e_j] E_{t_\infty} t_T = e_j E_{t_j} t_T = e_j t_{X_j} V_{t_j} = [j, e_j t_{X_j}] V_{t_\infty} = [j, e_j] t_{X_\infty} V_{t_\infty} = e_\infty t_{X_\infty} V_{t_\infty}$. \square

Remark 100 Suppose given graph morphisms $w', w'' : X_\infty \rightarrow T$ such that $f_{k,\infty} w' = f_{k,\infty} w''$ for $k \geq 0$.

Then $w' = w''$.

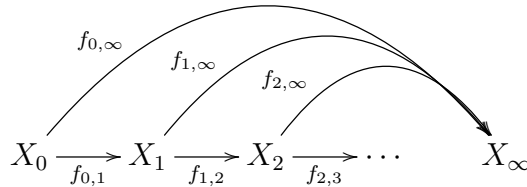
We say that for $k \geq 0$ the graph morphisms $f_{k,\infty}$ are *collectively epimorphic*.

Proof. Both graph morphisms w' and w'' are induced by $(f_{k,\infty} w')_{k \geq 0} = (f_{k,\infty} w'')_{k \geq 0}$.

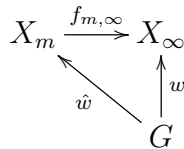
By the uniqueness in the universal property, it follows that $w' = w''$; cf. Lemma 99. \square

Lemma 101 Suppose given a finite graph G , i.e. the sets V_G and E_G are finite.

Suppose given



- (1) Suppose given a graph morphism $w : G \rightarrow X_\infty$. Then there exist $m \geq 0$ and a graph morphism $\hat{w} : G \rightarrow X_m$ such that $\hat{w} f_{m,\infty} = w$.



- (2) Suppose given graph morphisms $w' : G \rightarrow X_\infty$ and $w'' : G \rightarrow X_\infty$. Then there exist $m \geq 0$ and $\hat{w}' : G \rightarrow X_m$ and $\hat{w}'' : G \rightarrow X_m$ such that we have $\hat{w}' f_{m,\infty} = w'$ and $\hat{w}'' f_{m,\infty} = w''$.

$$\begin{array}{ccc}
 X_m & \xrightarrow{f_{m,\infty}} & X_\infty \\
 \hat{w}'' \swarrow & & \uparrow w' \\
 & & G \\
 \hat{w}' \swarrow & & \uparrow w''
 \end{array}$$

Proof.

Ad (1). We will construct \hat{w} .

Step 1. We choose $j \geq 0$ such that $e E_w \in E_{X_j f_{j,\infty}}$ and $v V_w \in V_{X_j f_{j,\infty}}$ for $e \in E_G$, $v \in V_G$.

This is possible since the graph G is finite; cf. Definition 98.

Step 2. For $e \in E_G$ we choose $e' \in E_{X_j}$ such that $[j, e'] = e E_w$.

Note that $[j, e'] = e' E_{f_{j,\infty}}$.

For $v \in V_G$ we choose $v' \in V_{X_j}$ such that $[j, v'] = v V_w$.

Note that $[j, v'] = v' V_{f_{j,\infty}}$.

Step 3. Let $e \in E_G$.

We have $[j, e' s_{X_j}] \stackrel{\text{Def. 98}}{=} [j, e'] s_{X_\infty} = e E_w s_{X_\infty} = e s_G V_w = [j, (e s_G)']$.

So there exists $m_{e,1} \geq j$ such that $e' s_{X_j} V_{f_{j,m_{e,1}}} = (e s_G)' V_{f_{j,m_{e,1}}}$.

We have $[j, e' t_{X_j}] \stackrel{\text{Def. 98}}{=} [j, e'] t_{X_\infty} = e E_w t_{X_\infty} = e t_G V_w = [j, (e t_G)']$.

So there exists $m_{e,2} \geq j$ such that $e' t_{X_j} V_{f_{j,m_{e,2}}} = (e t_G)' V_{f_{j,m_{e,2}}}$.

Step 4. Let $m := \max(\{m_{e,1} : e \in E_G\} \cup \{m_{e,2} : e \in E_G\})$. This is existent since the set E_G is finite.

Then $e' s_{X_j} V_{f_{j,m}} = (e s_G)' V_{f_{j,m}}$ and $e' t_{X_j} V_{f_{j,m}} = (e t_G)' V_{f_{j,m}}$ for $e \in E_G$.

This follows by application of $V_{f_{m_{e,1},m}}$ respectively $V_{f_{m_{e,2},m}}$ to the equations in Step 3.

Step 5. We have the pair of maps

$$\begin{aligned}
 \hat{w} : G &\rightarrow X_m \\
 E_{\hat{w}} : e &\mapsto e' E_{f_{j,m}} \\
 V_{\hat{w}} : v &\mapsto v' V_{f_{j,m}} .
 \end{aligned}$$

Step 6. The pair of maps $\hat{w} = (V_{\hat{w}}, E_{\hat{w}}) : G \rightarrow X_m$ is a graph morphism.

Suppose given $e \in E_G$.

We have to show that $e E_{\hat{w}} s_{X_m} \stackrel{!}{=} e s_G V_{\hat{w}}$ and that $e E_{\hat{w}} t_{X_m} \stackrel{!}{=} e t_G V_{\hat{w}}$.

We have $e E_{\hat{w}} s_{X_m} \stackrel{\text{Step 5}}{=} e' E_{f_{j,m}} s_{X_m} = e' s_{X_j} V_{f_{j,m}} \stackrel{\text{Step 4}}{=} (e s_G)' V_{f_{j,m}} \stackrel{\text{Step 5}}{=} e s_G V_{\hat{w}}$.

We have $e E_{\hat{w}} t_{X_m} \stackrel{\text{Step 5}}{=} e' E_{f_{j,m}} t_{X_m} = e' t_{X_j} V_{f_{j,m}} \stackrel{\text{Step 4}}{=} (e t_G)' V_{f_{j,m}} \stackrel{\text{Step 5}}{=} e t_G V_{\hat{w}}$.

Step 7. We have to show that $\hat{w} \cdot f_{m,\infty} \stackrel{!}{=} w$.

Suppose given $e \in E_G$.

We have $e E_{\hat{w}f_{m,\infty}} = e E_{\hat{w}} E_{f_{m,\infty}} \stackrel{\text{Step 5}}{=} e' E_{f_{j,m}} E_{f_{m,\infty}} = e' E_{f_{j,\infty}} = [j, e'] \stackrel{\text{Step 2}}{=} e E_w$.

Suppose given $v \in V_G$.

We have $v V_{\hat{w}f_{m,\infty}} = v V_{\hat{w}} V_{f_{m,\infty}} \stackrel{\text{Step 5}}{=} v' V_{f_{j,m}} V_{f_{m,\infty}} = v' V_{f_{j,\infty}} = [j, v'] \stackrel{\text{Step 2}}{=} v V_w$. \square

Ad (2). By (1) there exist $m', m'' \geq 0$ and graph morphisms $\tilde{w}' : G \rightarrow X_{m'}$ and $\tilde{w}'' : G \rightarrow X_{m''}$ such that $\tilde{w}' f_{m',\infty} = w'$ and $\tilde{w}'' f_{m'',\infty} = w''$.

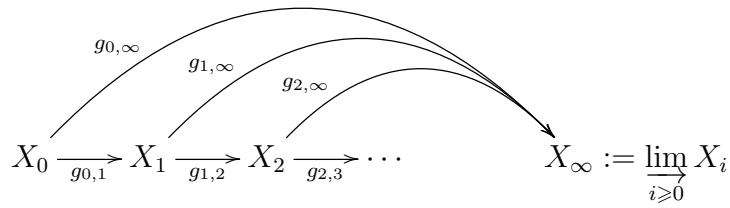
Let $m := \max\{m', m''\}$.

Let $\hat{w}' := \tilde{w}' f_{m',m} : G \rightarrow X_m$ and let $\hat{w}'' := \tilde{w}'' f_{m'',m} : G \rightarrow X_m$.

Then we have $\hat{w}' f_{m,\infty} = \tilde{w}' f_{m',m} f_{m,\infty} = \tilde{w}' f_{m',\infty} = w'$.

And we have $\hat{w}'' f_{m,\infty} = \tilde{w}'' f_{m'',m} f_{m,\infty} = \tilde{w}'' f_{m'',\infty} = w''$.

Lemma 102 Suppose given $A \subseteq \text{Mor}(\text{Gph})$ and



in Gph such that $g_{i,i+1} \in \square A$ for $i \geq 0$.

Then the graph morphism $g_{0,\infty} : X_0 \rightarrow X_\infty$ is in $\square A$.

Proof. Suppose given

$$\begin{array}{ccc}
 X_0 & \xrightarrow{a} & Y \\
 g_{0,\infty} \downarrow & & \downarrow h \\
 X_\infty & \xrightarrow{b} & Z
 \end{array} ,$$

where $h \in A$.

We have to show that $g_{0,\infty} \square h$, i.e. that there exists a graph morphism $c : X_\infty \rightarrow Y$ such that $g_{0,\infty} c = a$ and $ch = b$.

Let $c_0 := a$.

Since the graph morphism $g_{0,1} : X_0 \rightarrow X_1$ is in $\square A$ and since we have $g_{0,1} \cdot g_{1,\infty} b = g_{0,\infty} b = ah = c_0 h$, we may choose a graph morphism $c_1 : X_1 \rightarrow Y$ such that $g_{0,1} c_1 = c_0$ and $c_1 h = g_{1,\infty} b$.

Since the graph morphism $g_{1,2} : X_1 \rightarrow X_2$ is in $\square A$ and since we have $g_{1,2} \cdot g_{2,\infty} b = g_{1,\infty} b = c_1 h$, we may choose a graph morphism $c_2 : X_2 \rightarrow Y$ such that $g_{1,2} c_2 = c_1$ and $c_2 h = g_{2,\infty} b$.

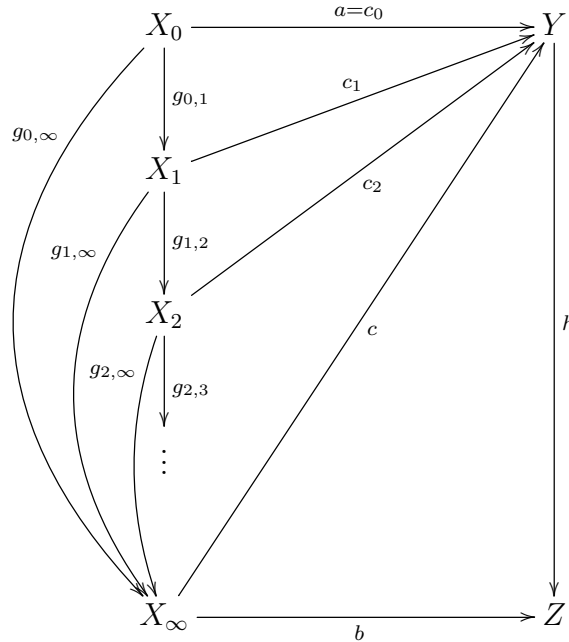
Since the graph morphism $g_{2,3} : X_2 \rightarrow X_3$ is in $\square A$ and since we have $g_{2,3} \cdot g_{3,\infty} b = g_{2,\infty} b = c_2 h$, we may choose a graph morphism $c_3 : X_3 \rightarrow Y$ such that $g_{2,3} c_3 = c_2$ and $c_3 h = g_{3,\infty} b$.

Etc.

In a recursive way we may choose a graph morphism $c_k : X_k \rightarrow Y$ such that $g_{k-1,k}c_k = c_{k-1}$ and $c_k h = g_{k,\infty}b$ for $k \geq 0$.

We obtain $g_{0,k}c_k = g_{0,1} \cdot g_{1,2} \cdot \dots \cdot g_{k-1,k}c_k = c_0$ for $k \geq 0$.

Because of the universal property of the colimit in Gph there exists a unique graph morphism $c : X_\infty \rightarrow Y$ such that $g_{k,\infty}c = c_k$ for $k \geq 0$; cf. Lemma 99.



So we have

$$\begin{array}{ccc} X_0 & \xrightarrow{a=c_0} & Y \\ g_{0,\infty} \downarrow & \nearrow c & \downarrow h \\ X_\infty & \xrightarrow{b} & Z \end{array}$$

such that $g_{0,\infty}c = c_0 = a$.

We have to show that $ch \stackrel{!}{=} b$.

So we have $g_{k,\infty}ch = c_k h = g_{k,\infty}b$ for $k \geq 0$.

Now we deduce $ch = b$ by Remark 100. □

Definition 103 Suppose given a countable chain of subgraphs $Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y$ of Y .

Let $V_{Y_\infty} := \bigcup_{i \geq 0} V_{Y_i} \subseteq V_Y$ and $E_{Y_\infty} := \bigcup_{i \geq 0} E_{Y_i} \subseteq E_Y$.

Let $s_{Y_\infty} := s_Y|_{E_{Y_\infty}}$ and $t_{Y_\infty} = t_Y|_{E_{Y_\infty}}$.

Then the graph $Y_\infty := \bigcup_{i \geq 0} Y_i := (V_{Y_\infty}, E_{Y_\infty}, s_{Y_\infty}, t_{Y_\infty})$ is a subgraph of Y ; cf. Definition 47.(1).

In fact, suppose given an edge $e \in E_{Y_\infty} = \bigcup_{i \geq 0} E_{Y_i}$. There exists $i \geq 0$ such that $e \in E_{Y_i}$.

We have $e s_Y = e s_{Y_i} \in V_{Y_i} \subseteq V_{Y_\infty}$ and $e t_Y = e t_{Y_i} \in V_{Y_i} \subseteq V_{Y_\infty}$. □

Remark 104 Suppose given a graph Y .

Suppose given a countable chain of subgraphs $Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y$ such that $Y_i \subseteq Y$ is a full subgraph for $i \in \mathbb{Z}_{\geq 0}$.

Then $\bigcup_{i \in \mathbb{Z}_{\geq 0}} Y_i \subseteq Y$ is a full subgraph; cf. Definition 103.

Proof. Suppose given vertices $v, w \in V_{\bigcup_{i \in \mathbb{Z}_{\geq 0}} Y_i} = \bigcup_{i \in \mathbb{Z}_{\geq 0}} V_{Y_i}$.

Suppose given an edge $e \in Y(v, w)$.

We have to show that $e \in E_{\bigcup_{i \in \mathbb{Z}_{\geq 0}} Y_i} = \bigcup_{i \in \mathbb{Z}_{\geq 0}} E_{Y_i}$.

We can choose $j \in \mathbb{Z}_{\geq 0}$ such that $v \in Y_j$ and $w \in Y_j$.

The subgraph $Y_j \subseteq Y$ is a full subgraph.

So we have $e \in E_{Y_j} \subseteq \bigcup_{i \in \mathbb{Z}_{\geq 0}} E_{Y_i}$. □

Remark 105 Suppose given a countable chain of subgraphs $X = Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y$ of Y .

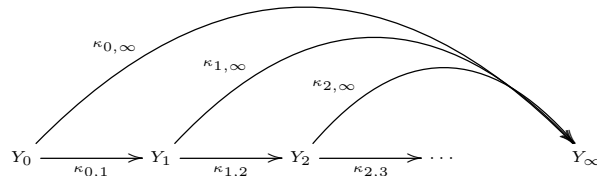
The graph $Y_\infty = \bigcup_{i \geq 0} Y_i \subseteq Y$ is a subgraph of Y ; cf. Definition 103.

Then the graph Y_∞ is a colimit of the countable chain of subgraphs $Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y$; cf. Definition 42.

Proof. First, we abbreviate $\kappa_{i,j} := \text{id}_{Y_j} |_{Y_i} : Y_i \rightarrow Y_j$ for $0 \leq i \leq j$.

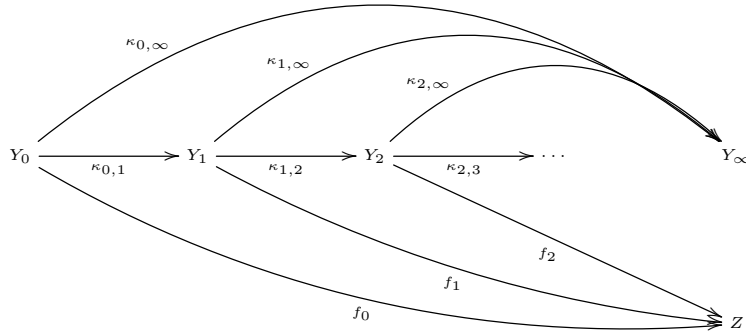
Suppose given $i \geq 0$. Since the graph Y_i is a subgraph of $Y_\infty = \bigcup_{i \geq 0} Y_i$, we have the graph morphisms $\kappa_{i,\infty} := \text{id}_{Y_\infty} |_{Y_i} : Y_i \rightarrow Y_\infty$.

We have

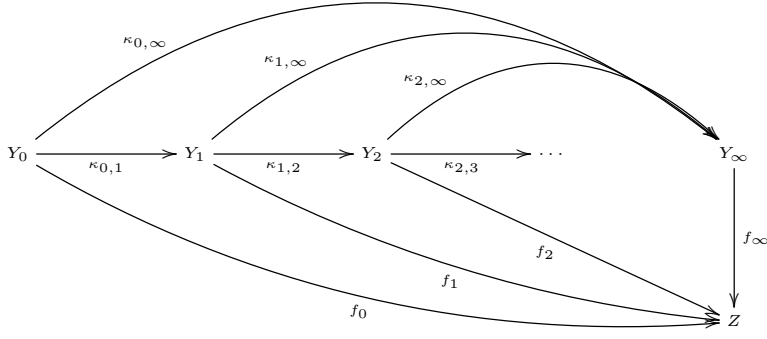


We show that the graph Y_∞ is a colimit of the chain of subgraphs $Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y$ with respect to $\kappa_{i,\infty} : Y_i \rightarrow Y_\infty$ for $i \geq 0$.

Suppose given a commutative diagram in Gph as follows.



We have to show that there exists a unique graph morphism $f_\infty : Y_\infty \rightarrow Z$ such that the following diagram is commutative.



Uniqueness. This follows by $V_{Y_\infty} = \bigcup_{i \geq 0} V_{Y_i}$ and $E_{Y_\infty} = \bigcup_{i \geq 0} E_{Y_i}$.

Existence. We let

$$\begin{aligned} V_{f_\infty} : V_{Y_\infty} = \bigcup_{i \geq 0} V_{Y_i} &\rightarrow V_Z \\ v &\mapsto v V_{f_i}, \text{ if } v \in V_{Y_i}. \end{aligned}$$

We have to show that this map is well-defined.

Suppose given $v \in V_{Y_\infty}$ and $v \in V_{Y_i}$ and $v \in V_{Y_j}$, where $i, j \geq 0$.

We show that we have $v V_{f_i} \stackrel{!}{=} v V_{f_j}$.

Without loss of generality, we have $i \leq j$ and thus $v V_{f_i} = v V_{\kappa_{i,j} f_j} = v V_{\kappa_{i,j}} V_{f_j} = v V_{f_j}$.

We let

$$\begin{aligned} E_{f_\infty} : E_{Y_\infty} = \bigcup_{i \geq 0} E_{Y_i} &\rightarrow E_Z \\ v &\mapsto v E_{f_i}, \text{ if } v \in E_{Y_i}. \end{aligned}$$

We have to show that this map is well-defined.

Suppose given $e \in E_{Y_\infty}$ and $e \in E_{Y_i}$ and $e \in E_{Y_j}$, where $i, j \geq 0$.

We show that we have $e E_{f_i} \stackrel{!}{=} e E_{f_j}$.

Without loss of generality, we have $i \leq j$ and thus $e E_{f_i} = e E_{\kappa_{i,j} f_j} = e E_{\kappa_{i,j}} E_{f_j} = e E_{f_j}$.

We show that the pair of maps $f_\infty := (V_{f_\infty}, E_{f_\infty})$ is a graph morphism.

Suppose given an edge $e \in E_{Y_\infty}$.

We may choose $i \geq 0$ such that $e \in E_{Y_i}$.

We have to show that $e s_{Y_\infty} V_{f_\infty} \stackrel{!}{=} e E_{f_\infty} s_Z$.

We have $e s_{Y_\infty} V_{f_\infty} \stackrel{\text{Def. 103}}{=} e s_{Y_i} V_{f_\infty} \stackrel{\text{Def. } V_{f_\infty}}{=} e s_{Y_i} V_{f_i} = e E_{f_i} s_Z \stackrel{\text{Def. } E_{f_\infty}}{=} e E_{f_\infty} s_Z$.

We have to show that $e t_{Y_\infty} V_{f_\infty} \stackrel{!}{=} e E_{f_\infty} t_Z$.

We have $e t_{Y_\infty} V_{f_\infty} \stackrel{\text{Def. 103}}{=} e t_{Y_i} V_{f_\infty} \stackrel{\text{Def. } V_{f_\infty}}{=} e t_{Y_i} V_{f_i} = e E_{f_i} t_Z \stackrel{\text{Def. } E_{f_\infty}}{=} e E_{f_\infty} t_Z$.

We now show that we have $\kappa_{i,\infty} f_\infty \stackrel{!}{=} f_i$ for $i \geq 0$.

Suppose given $v \in V_{Y_i}$. We have $v V_{\kappa_{i,\infty}} V_{f_\infty} = v V_{f_\infty} \stackrel{\text{Def. } f_\infty}{=} v V_{f_i}$.

Suppose given $e \in E_{Y_i}$. We have $e E_{\kappa_{i,\infty}} E_{f_\infty} = e E_{f_\infty} \stackrel{\text{Def. } f_\infty}{=} e E_{f_i}$. □

Lemma 106 Suppose given $A \subseteq \text{Mor}(\text{Gph})$.

Suppose given a countable chain of subgraphs $X = Y_0 \subseteq Y_1 \subseteq \dots \subseteq Y$ such that $Y = \bigcup_{i \geq 0} Y_i$.
Suppose that the inclusion morphism $\text{id}_{Y_{i+1}}|_{Y_i} : Y_i \rightarrow Y_{i+1}$ is in $\square A$ for $i \in \mathbb{Z}_{\geq 0}$.

Then the inclusion morphism $\text{id}_Y|_X : X \rightarrow Y$ is a morphism in $\square A$.

Proof. The graph Y is the colimit of countable chain of subgraphs $X = Y_0 \subseteq Y_1 \subseteq \dots \subseteq Y$ with respect to the inclusion morphisms; cf. Remark 105.

Then by Lemma 102 the inclusion morphism $\text{id}_Y|_X : X \rightarrow Y$ is a morphism in $\square A$. \square

2.5 Tree graphs

Definition 107 A graph morphism $p : D_n \rightarrow X$ for some $n \geq 0$ is called *path* in X .

Given a path $p : D_n \rightarrow X$, we let $ps := \hat{v}_0 V_p$ and $pt := \hat{v}_n V_p$.

We say p is a path from ps to pt with *length* n . We write $\text{length}(p) := n$.

We often write $p = (\hat{v}_0 V_p; \hat{e}_0 E_p, \dots, \hat{e}_{n-1} E_p)$.

Conversely, given $v \in V_X$, $m \geq 0$ and $e_0, \dots, e_{m-1} \in E_X$, then $(v; e_0, \dots, e_{m-1})$ is a path in X if $e_0 s_X = v$ and $e_i t_X = e_{i+1} s_X$ for $i \in [0, m-2]$.

A graph morphism $p : D_0 \rightarrow X$ with $\hat{v}_0 V_p = x \in V_X$ is called the *empty path at* x , also written $p = (x;)$.

Suppose given a path $p = (v; e_0, \dots, e_{n-1})$ in X . We have $ps = v$. If $n \geq 1$ then we have $pt = e_{n-1} t_X$, if $n = 0$ then we have $pt = v$.

For a path $p = (v; e_0, \dots, e_{n-1})$ in X and an edge $e \in E_X$ with $e s_X = pt$, we let

$$p \cdot e := (v; e_0, \dots, e_{n-1}, e).$$

Definition 108 A graph G is called a *tree* if there exists $r \in V_G$ such that the following properties (Tree 1–3) hold.

(Tree 1) We have $|\{e \in E_G : (e) t_G = v\}| = 1$ for $v \in V_G \setminus \{r\}$.

(Tree 2) We have $(e) t_G \neq r$ for $e \in E_G$.

(Tree 3) There exists a path from r to v for $v \in V_G$.

The vertex $r \in V_G$ is unique with these properties. We call this vertex the *root* r of G .

Proof. Suppose given two vertices $r, \tilde{r} \in V_G$ with $r \neq \tilde{r}$ satisfying (Tree 1–3).

Then by (Tree 1) we have $|\{e \in E_G : (e) t_G = \tilde{r}\}| = 1$ since $\tilde{r} \in V_G \setminus \{r\}$, in *contradiction* to (Tree 2). \square

Definition 109 Suppose given a graph X and a vertex $x \in V_X$.

We will define the graph $\text{Tree}(x, X)$.

We let

$$V_{\text{Tree}(x, X)} := \{\alpha : \alpha \text{ is a path in } X \text{ with } \alpha s = x\}.$$

We let

$$E_{\text{Tree}(x, X)} := \{(\alpha, e, \beta) : \alpha, \beta \in V_{\text{Tree}(x, X)}, e \in E_X \text{ such that } \alpha t = e s_X \text{ and } \alpha \cdot e = \beta\}.$$

Note that for $(\alpha, e, \beta) \in E_{\text{Tree}(x, X)}$ we have $e t_X = (\alpha \cdot e) t = \beta t$.

For $(\alpha, e, \beta) \in E_{\text{Tree}(x, X)}$, we let $(\alpha, e, \beta) s_{\text{Tree}(x, X)} := \alpha$ and $(\alpha, e, \beta) t_{\text{Tree}(x, X)} := \beta$.

Remark 110 Suppose given a graph X . Suppose given a vertex $x \in V_X$.

We have

$$V_{\text{Tree}(x, X)} = \{(x; e_0, \dots, e_{n-1}) : n \geq 0, (x; e_0, \dots, e_{n-1}) \text{ path in } X\}$$

and

$$E_{\text{Tree}(x, X)} = \{((x; e_0, \dots, e_{n-1}), e_n, (x; e_0, \dots, e_n)) : n \geq 0, (x; e_0, \dots, e_n) \text{ path in } X\}.$$

Proof.

Ad $V_{\text{Tree}(x, X)}$.

Suppose given $n \geq 1$. Suppose given a path $\alpha = (x; e_0, \dots, e_{n-1})$ in X . Then we have $\alpha s = x$; cf. Definition 109.

Conversely, suppose given a path α in X with $\alpha s = x$. Then we have $\text{length}(\alpha) \geq 0$ and we write $\alpha = (\alpha s; \hat{e}_0 E_\alpha, \dots, \hat{e}_{\text{length}(\alpha)-1} E_\alpha) =: (x; e_0, \dots, e_{\text{length}(\alpha)-1})$, which is a path in X .

Ad $E_{\text{Tree}(x, X)}$.

Suppose given $(\alpha, e, \beta) \in E_{\text{Tree}(x, X)}$. So α and β are paths in X and we have an edge $e \in E_X$ such that $\alpha s = x$ and $\alpha t = e s_X$ and $\alpha \cdot e = \beta$.

We write $n := \text{length}(\alpha) \geq 0$. Then $\text{length}(\beta) = \text{length}(\alpha \cdot e) = \text{length}(\alpha) + 1 = n + 1$. We write $\alpha = (\alpha s; \hat{e}_0 E_\alpha, \dots, \hat{e}_{n-1} E_\alpha) =: (x; e_0, \dots, e_{n-1})$. We write $e_n := e$. So we have $\beta = \alpha \cdot e = \alpha \cdot e_n = (x; e_0, \dots, e_{n-1}, e_n)$.

So we have $(\alpha, e, \beta) = ((x; e_0, \dots, e_{n-1}), e, (x; e_0, \dots, e_{n-1}, e_n))$.

Conversely, suppose given $n \geq 0$ and $((x; e_0, \dots, e_{n-1}), e_n, (x; e_0, \dots, e_n))$ such that $(x; e_0, \dots, e_n) =: \beta$ is a path in X .

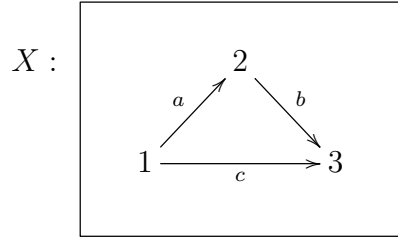
Then we have the graph morphism $\beta : D_{n+1} \rightarrow X$ with $(\hat{v}_0 V_\beta; \hat{e}_0 E_\beta, \dots, \hat{e}_n E_\beta) = (x; e_0, \dots, e_n)$. We let $\alpha : D_n \rightarrow X$ be the graph morphism with $\hat{v}_i V_\alpha := \hat{v}_i V_\beta$ for $i \in [0, n]$ and with $\hat{e}_i E_\alpha := \hat{e}_i E_\beta$ for $i \in [0, n-1]$. So we have $\alpha = \beta|_{D_n} : D_n \rightarrow X$.

We obtain $\alpha = (\hat{v}_0 V_\alpha; \hat{e}_0 E_\alpha, \dots, \hat{e}_{n-1} E_\alpha) = (\hat{v}_0 V_\beta; \hat{e}_0 E_\beta, \dots, \hat{e}_{n-1} E_\beta) = (x; e_0, \dots, e_{n-1})$.

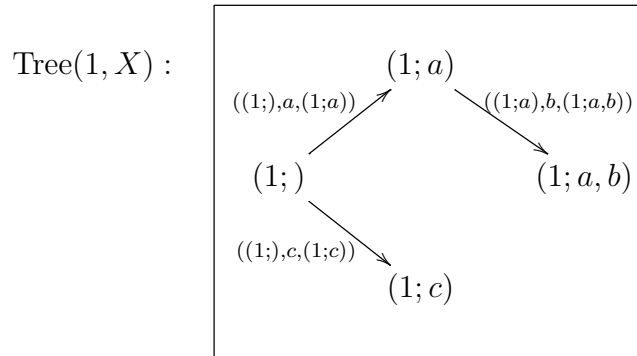
So we have $((x; e_0, \dots, e_{n-1}), e_n, (x; e_0, \dots, e_n)) = (\alpha, e_n, \beta)$, where α and β are paths in X and the edge $e_n = \hat{e}_n E_\beta$ is in E_X . Furthermore, we have $\alpha s = x$ and $\alpha t = \hat{v}_n V_\alpha = \hat{v}_n V_\beta = \hat{e}_n s_{D_{n+1}} V_\beta = \hat{e}_n E_\beta s_X = e_n s_X$.

And we have $\alpha \cdot e_n = (x; e_0, \dots, e_{n-1}) \cdot e_n = (x; e_0, \dots, e_{n-1}, e_n) = \beta$. □

Example 111 We consider the following graph.



We have the graph



In the following Remark 112 we will show that for every graph G and every vertex $x \in V_G$, the graph $\text{Tree}(x, G)$ actually is a tree.

Remark 112 Suppose given a graph G and a vertex $x \in V_G$.

Then the graph $\text{Tree}(x, G)$ is a tree with root $r := (x;) \in V_{\text{Tree}(x, G)}$.

Proof.

We have to show that the properties (Tree 1–3) hold for $\text{Tree}(x, G)$; cf. Definitions 108 and 109.

Ad (Tree 1).

Suppose given a vertex $v \in V_{\text{Tree}(x, G)} \setminus \{r\}$.

We have to show that $|\{\epsilon \in E_{\text{Tree}(x, G)} : (\epsilon) t_{\text{Tree}(x, G)} = v\}| \stackrel{!}{=} 1$.

The vertex $v \in V_{\text{Tree}(x, G)}$ is a path $v : D_n \rightarrow G$ in G with $v s = x$ and $n \geq 1$ since $v \neq r$.

We write $v = (x; e_0, \dots, e_n)$ with $e_i \in E_G$ for $i \in [1, n]$.

For an edge $\epsilon = (\alpha, e, \beta) \in E_{\text{Tree}(x, G)}$, we have $(\alpha, e, \beta) t_{\text{Tree}(x, G)} = \beta$.

So $(\epsilon) t_{\text{Tree}(x, G)} = v$ if and only if $\epsilon = (\alpha, e, v)$, where $\alpha : D_{n-1} \rightarrow G$ is a path with $\alpha t = e s_G$ and $\alpha \cdot e = v$. So it has to be $e = e_n$ and $\alpha = (x; e_1, \dots, e_{n-1})$; cf. Remark 110.

We obtain

$$\{e \in E_{\text{Tree}(x, G)} : e t_{\text{Tree}(x, G)} = v\} = \{((x; e_0, \dots, e_{n-1}), e_n, (x; e_0, \dots, e_n))\},$$

which contains a single element.

Ad (Tree 2).

We have to show that $(\epsilon) t_{\text{Tree}(x, G)} \stackrel{!}{\neq} r$ for $\epsilon \in E_{\text{Tree}(x, G)}$.

Suppose given an edge $(\alpha, e, \beta) \in E_{\text{Tree}(x, G)}$.

We have $(\alpha, e, \beta) \text{t}_{\text{Tree}(x, G)} = \beta$. Since $\alpha \cdot e = \beta$ we have $\text{length}(\beta) = \text{length}(\alpha \cdot e) = \text{length}(\alpha) + 1 \geq 1$, whereas $\text{length}(r) = 0$. So we conclude that $(\alpha, e, \beta) \text{t}_{\text{Tree}(x, G)} = \beta \neq r$.

Ad (Tree 3).

We have to show that there exists a path from $(x;)$ to v for $v \in V_{\text{Tree}(x, G)}$.

The vertex $v \in V_{\text{Tree}(x, G)}$ is a path $v : D_n \rightarrow G$ in G from $v \text{s} = x$, where $n := \text{length}(v) \geq 0$.

We write $v = (x; e_0, \dots, e_{n-1})$ with $e_i \in E_G$ for $i \in [1, n-1]$.

The following graph morphism $p : D_n \rightarrow \text{Tree}(x, G)$ will be a path in $\text{Tree}(x, G)$ from $p \text{s} = \hat{v}_0 V_p = (x;)$ to $p \text{t} = \hat{v}_n V_p = v = (x; e_0, \dots, e_{n-1})$.

Using Remark 110, we let

$$\begin{aligned} p : D_n &\rightarrow \text{Tree}(x, G) \\ V_p : V_{D_n} &\rightarrow V_{\text{Tree}(x, G)} \\ &\hat{v}_i \mapsto (x; e_0, \dots, e_{i-1}) \text{ for } i \in [0, n] \\ E_p : E_{D_n} &\rightarrow E_{\text{Tree}(x, G)} \\ &\hat{e}_i \mapsto ((x; e_0, \dots, e_{i-1}), e_i, (x; e_0, \dots, e_i)) \text{ for } i \in [0, n-1]. \end{aligned}$$

We have to show that $E_p \text{s}_{\text{Tree}(x, G)} \stackrel{!}{=} \text{s}_{D_n} V_p$ and that $E_p \text{t}_{\text{Tree}(x, G)} \stackrel{!}{=} \text{t}_{D_n} V_p$.

Suppose given an edge $\hat{e}_i \in E_{D_n}$, where $i \in [0, n-1]$.

We have $\hat{e}_i E_p \text{s}_{\text{Tree}(x, G)} = ((x; e_0, \dots, e_{i-1}), e_i, (x; e_0, \dots, e_i)) \text{s}_{\text{Tree}(x, G)} = (x; e_0, \dots, e_{i-1}) = \hat{v}_i V_p = \hat{e}_i \text{s}_{D_n} V_p$.

And we have $\hat{e}_i E_p \text{t}_{\text{Tree}(x, G)} = ((x; e_0, \dots, e_{i-1}), e_i, (x; e_0, \dots, e_i)) \text{t}_{\text{Tree}(x, G)} = (x; e_0, \dots, e_i) = \hat{v}_{i+1} V_p = \hat{e}_i \text{t}_{D_n} V_p$.

Finally, $p \text{s} = \hat{v}_0 V_p = (x;) = r$ and $p \text{t} = \hat{v}_n V_p = (x; e_0, \dots, e_{n-1}) = v$.

$$(x;) \xrightarrow{((x;), e_0, (x; e_0))} (x; e_0) \xrightarrow{((x; e_0), e_1, (x; e_0, e_1))} (x; e_0, e_1) \xrightarrow{((x; e_0, e_1), e_2, (x; e_0, e_1, e_2))} \dots \quad (x; e_1, \dots, e_n)$$

□

Definition 113 Suppose given a graph G and a vertex $x \in V_G$.

We define the *projection morphism* $p_x : \text{Tree}(x, G) \rightarrow G$ at x .

For a path $\alpha : D_n \rightarrow G$ in $V_{\text{Tree}(x, G)}$ we let $\alpha V_{p_x} := \hat{v}_{\text{length}(\alpha)} V_\alpha$.

For an edge $(\alpha, e, \beta) \in E_{\text{Tree}(x, G)}$ we let $(\alpha, e, \beta) E_{p_x} := e \in E_G$.

To verify that

$$\begin{aligned} p_x : \text{Tree}(x, G) &\rightarrow G \\ V_{p_x} : V_{\text{Tree}(x, G)} &\rightarrow V_G \\ &\alpha \mapsto \hat{v}_{\text{length}(\alpha)} V_\alpha \\ E_{p_x} : E_{\text{Tree}(x, G)} &\rightarrow E_G \\ &(\alpha, e, \beta) \mapsto e \end{aligned}$$

is a graph morphism, we have to show that $E_{p_x} s_G \stackrel{!}{=} s_{\text{Tree}(x,G)} V_{p_x}$ and that $E_{p_x} t_G \stackrel{!}{=} t_{\text{Tree}(x,G)} V_{p_x}$.

So suppose given an edge $(\alpha, e, \beta) \in E_{\text{Tree}(x,G)}$.

Note that $e \in E_G$ with $e s_G = \alpha t = \hat{v}_{\text{length}(\alpha)} V_\alpha$ and with $e t_G = \beta t = \hat{v}_{\text{length}(\beta)} V_\beta$.

We have

$$(\alpha, e, \beta) E_{p_x} s_G = e s_G = \alpha t = \hat{v}_{\text{length}(\alpha)} V_\alpha = \alpha V_{p_x} = (\alpha, e, \beta) s_{\text{Tree}(x,G)} V_{p_x} .$$

And we have

$$(\alpha, e, \beta) E_{p_x} t_G = e t_G = \beta t = \hat{v}_{\text{length}(\beta)} V_\beta = \beta V_{p_x} = (\alpha, e, \beta) t_{\text{Tree}(x,G)} V_{p_x} .$$

So p_x is in fact a graph morphism.

Given $n \geq 1$ and $\alpha = (x; e_0, \dots, e_{n-1}) \in V_{\text{Tree}(x,G)}$, we have $\alpha V_{p_x} = e_{n-1} t_G$.

Moreover, we have $(x;) V_{p_x} = x$.

Given $(\alpha, e, \beta) = (x; (e_0, \dots, e_{n-1}), e_n, (e_0, \dots, e_n)) \in E_{\text{Tree}(x,G)}$, we have

$$(\alpha, e, \beta) E_{p_x} = (x; (e_0, \dots, e_{n-1}), e_n, (e_0, \dots, e_n)) E_{p_x} = e_n .$$

Remark 114 Suppose given a graph G and a vertex $x \in V_G$.

We recall the projection morphism $p_x : \text{Tree}(x, G) \rightarrow G$; cf. Definition 113.

We have the bijection

$$E_{p_x} \Big|_{\text{Tree}(x,G)((x;),*)}^{G((x;), V_{p_x}, *)} : \text{Tree}(x, G)((x;), *) \rightarrow G((x;), V_{p_x}, *) = G(x, *) .$$

Proof. Note that $(x;)$ is the empty path in $x \in V_G$ and thus a vertex in $\text{Tree}(x, G)$. So the set $\text{Tree}(x, G)((x;), *)$ consists of all edges in $\text{Tree}(x, G)$ which have the empty path as source. So

$$\text{Tree}(x; G)((x;), *) = \{((x;), e, (x; e)) : e \in G(x, *) = G((x;), V_{p_x}, *)\} .$$

The claimed bijection ensues. □

Chapter 3

Properties of graph morphisms

We shall show in §6, Proposition 204, that Gph , equipped with a set of fibrations, a set of cofibrations and a set of quasiisomorphisms is a model category in the sense of Definition 198. To this end, we shall introduce these sets, already employing the language and symbolism of model categories before §6.

3.1 Quasiisomorphisms

Definition 115 Suppose given graphs G and H .

A graph morphism $f : G \rightarrow H$ is called a *quasiisomorphism* if the map $(C_n, f)_{\text{Gph}}$ is bijective for $n \geq 1$.

To indicate that f is a quasiisomorphism, we often write $G \xrightarrow{\approx} H$.

By $\text{Qis}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$ we denote the set of quasiisomorphisms in the category Gph .

We often write $\text{Qis} := \text{Qis}(\text{Gph})$.

So $f : G \rightarrow H$ is a quasiisomorphism if and only if for $n \geq 1$ and for each graph morphism $h : C_n \rightarrow H$ there exists a unique graph morphism $g : C_n \rightarrow G$ such that $gf = h$.

$$\begin{array}{ccc} & & G \\ & \nearrow \exists!g & \downarrow f \\ C_n & \xrightarrow{h} & H \end{array}$$

Remark 116 We have $\text{Iso}(\text{Gph}) \subseteq \text{Qis}(\text{Gph})$.

Proof. Suppose given a graph isomorphism $f : G \rightarrow H$.

We have to show that the graph isomorphism $f : G \rightarrow H$ is a quasiisomorphism, i.e. that the map $(C_n, f)_{\text{Gph}} : (C_n, G)_{\text{Gph}} \rightarrow (C_n, H)_{\text{Gph}}$ is bijective for $n \geq 1$; cf. Definition 115.

Suppose given $n \geq 1$. Then $(C_n, f)_{\text{Gph}} \cdot (C_n, f^{-1})_{\text{Gph}} \stackrel{\text{Rem. 69.(1)}}{=} (C_n, f \cdot f^{-1})_{\text{Gph}} = (C_n, \text{id}_H)_{\text{Gph}} \stackrel{\text{Rem. 69.(1)}}{=} \text{id}_{(C_n, H)_{\text{Gph}}}$.

Moreover $(C_n, f^{-1})_{\text{Gph}} \cdot (C_n, f)_{\text{Gph}} \stackrel{\text{Rem. 69}}{=} (C_n, f^{-1} \cdot f)_{\text{Gph}} = (C_n, \text{id}_G)_{\text{Gph}} \stackrel{\text{textRem. 69}}{=} \text{id}_{(C_n, G)_{\text{Gph}}}$.

So $(C_n, f)_{\text{Gph}}$ is bijective.

So the map $(C_n, f)_{\text{Gph}} : (C_n, G)_{\text{Gph}} \rightarrow (C_n, H)_{\text{Gph}}$ is bijective for $n \geq 1$ and thus the graph isomorphism $f : G \rightarrow H$ is a quasiisomorphism. \square

Remark 117 In Gph , the subset of quasiisomorphisms $\text{Qis} \subseteq \text{Mor}$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc} G' & \xrightarrow{f'} & H' \\ \uparrow p & & \uparrow q \\ \text{id}_{G'} \left(G \xrightarrow{\approx} H \right) \text{id}_{H'} & & \\ \uparrow i & & \uparrow j \\ G' & \xrightarrow{f'} & H' \end{array}$$

We have to show that the graph morphism $f' : G' \rightarrow H'$ is a quasiisomorphism; cf. Definition 115.

Suppose given

$$\begin{array}{ccc} G' & \xrightarrow{f'} & H' \\ \uparrow p & & \uparrow q \\ \text{id}_{G'} \left(G \xrightarrow{\approx} H \right) \text{id}_{H'} & & \\ \uparrow i & & \uparrow j \\ G' & \xrightarrow{f'} & H' \\ & & \uparrow u \\ & & C_n \end{array}$$

We have to show that there exists a unique graph morphism $\tilde{v} : C_n \rightarrow G'$ such that $\tilde{v}f' = u$; cf. Definition 115.

Existence. Because the graph morphism $f : G \rightarrow H$ is a quasiisomorphism there exists a unique graph morphism $v : C_n \rightarrow G$ such that $vf = u$.

So we have the graph morphism $\tilde{v} := vp : C_n \rightarrow G'$ satisfying $\tilde{v}f' = vp f' = v f q = u$.

Uniqueness. Suppose given

$$\begin{array}{ccc} G' & \xrightarrow{f'} & H' \\ \uparrow p & & \uparrow q \\ \text{id}_{G'} \left(G \xrightarrow{\approx} H \right) \text{id}_{H'} & & \\ \uparrow i & & \uparrow j \\ G' & \xrightarrow{f'} & H' \\ & \swarrow \tilde{v} & \uparrow u \\ & v & C_n \end{array}$$

such that $vf' = \tilde{v}f' = u$.

We have to show that $v \stackrel{!}{=} \tilde{v}$.

We have $vif = vf'j = uj = \tilde{v}f'j = \tilde{v}if$

Since the graph morphism $f : G \rightarrow H$ is a quasiisomorphism, we have $vi = \tilde{v}i$; cf. Definition 115.

So we have $v = vip = \tilde{v}ip = \tilde{v}$ and this shows uniqueness. \square

Example 118 The unique graph morphism $\iota_{D_0} : \emptyset \rightarrow D_0$ is a quasiisomorphism; cf. Definitions 70 and 115:

Suppose given $n \in \mathbb{N}$. There does not exist a graph morphism from C_n to \emptyset and there does not exist a graph morphism from C_n to D_0 , because $E_{C_n} \neq \emptyset$, but $E_\emptyset = \emptyset$ and $E_{D_0} = \emptyset$. So

$$(C_n, \iota_{D_0}) : (C_n, \emptyset)_{\text{Gph}} \rightarrow (C_n, D_0)_{\text{Gph}}$$

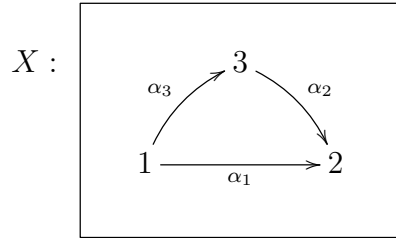
is bijective.

Lemma 119 Suppose given a graph X such that $(C_n, X) = \emptyset$ for $n \in \mathbb{X}$.

Then the graph morphism $\iota_X : \emptyset \rightarrow X$ is a quasiisomorphism.

Proof. For $n \in \mathbb{N}$ we have $(C_n, X) = \emptyset = (C_n, \emptyset)$. \square

Example 120 We consider the following graph.



Then $(C_n, X) = \emptyset$ for $n \in \mathbb{N}$. So the graph morphism $\iota_X : \emptyset \rightarrow X$ is a quasiisomorphism; cf. Lemma 119.

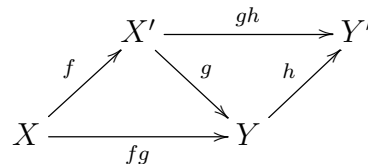
Example 121 Suppose given a graph morphism $f : X \rightarrow Y$ with $(C_n, X) = \emptyset = (C_n, Y)$ for $n \in \mathbb{N}$. Then the graph morphism $f : X \rightarrow Y$ is a quasiisomorphism; cf. Definition 115.

E.g. for $k, l \in \mathbb{N}$ with $k \leq l$, we may consider the graph morphism $\iota_{k,l} : D_k \rightarrow D_l$; cf. Definition 56. Then $\iota_{k,l}$ is a quasiisomorphism, since $(C_n, D_m) = \emptyset$ for $n, m \in \mathbb{N}$.

The following remark is called (2 of 6).

Remark 122 The subset $\text{Qis} \subseteq \text{Mor}(\text{Gph})$ satisfies (2 of 6).

Proof. Suppose given a commutative diagram in Gph as follows.



We have to show that the composites $fg : X \rightarrow Y$ and $gh : X' \rightarrow Y'$ are quasiisomorphisms if and only if f , g and h are quasiisomorphisms.

Applying $(C_n, -)_{\text{Gph}}$ to the given commutative diagram yields the following commutative diagram in Set ; cf. Remark 69.

$$\begin{array}{ccccc}
 & & (C_n, X')_{\text{Gph}} & \xrightarrow{(C_n, gh)_{\text{Gph}}} & (C_n, Y')_{\text{Gph}} \\
 & \nearrow (C_n, f)_{\text{Gph}} & & \searrow (C_n, g)_{\text{Gph}} & \\
 (C_n, X)_{\text{Gph}} & \xrightarrow{(C_n, fg)_{\text{Gph}}} & (C_n, Y)_{\text{Gph}} & \xleftarrow{(C_n, h)_{\text{Gph}}} &
 \end{array}$$

In this situation we have (2 of 6); cf. Remark 35.

The morphisms f, g and h are quasiisomorphisms if and only if the maps $(C_n, f)_{\text{Gph}}$, $(C_n, g)_{\text{Gph}}$ and $(C_n, h)_{\text{Gph}}$ are bijective for $n \in \mathbb{N}$. This holds if and only if the maps $(C_n, f)_{\text{Gph}} \cdot (C_n, g)_{\text{Gph}} = (C_n, fg)_{\text{Gph}}$ and $(C_n, g)_{\text{Gph}} \cdot (C_n, h)_{\text{Gph}} = (C_n, gh)_{\text{Gph}}$ are bijective for $n \in \mathbb{N}$; cf. Remark 35. This holds if and only if fg and gh are quasiisomorphisms; cf. Definition 115. \square

Remark 123 The subset $\text{Qis} \subseteq \text{Mor}(\text{Gph})$ satisfies (2 of 3).

Explicitly, this means the following.

Suppose given graph morphisms $f : X \rightarrow Y$ and $g : Y \rightarrow Z$.

Note the graph morphism $fg : X \rightarrow Z$.

We have the following commutative triangle.

$$\begin{array}{ccc}
 & Y & \\
 f \nearrow & & \searrow g \\
 X & \xrightarrow{fg} & Z
 \end{array}$$

The composite $fg : X \rightarrow Z$ is a quasiisomorphism if f and g are quasiisomorphisms.

The graph morphism g is a quasiisomorphism if f and fg are quasiisomorphisms.

The graph morphism f is a quasiisomorphism if g and fg are quasiisomorphisms.

Proof. Since Qis satisfies (2 of 6) and contains all identities in Gph , we conclude that Qis satisfies (2 of 3) by Lemma 2; cf. Remark 116. \square

Remark 124 Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc}
 & Y & \\
 a \nearrow & & \searrow b \\
 X & \xrightarrow{f} & Z
 \end{array}$$

Suppose given $k \geq 1$.

Suppose that the map $(C_k, f)_{\text{Gph}}$ is surjective.

Then the map $(C_k, b)_{\text{Gph}}$ is surjective.

Proof. Since $(C_k, a)_{\text{Gph}} \cdot (C_k, b)_{\text{Gph}} = (C_k, f)_{\text{Gph}}$ is surjective, so is $(C_k, b)_{\text{Gph}}$. \square

Remark 125 In Gph , a pullback of a quasiisomorphism is a quasiisomorphism; cf. Definition 115.

Proof. Suppose given a pullback in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & \ulcorner & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

We have to show that the graph morphism $g : X \rightarrow X'$ is a quasiisomorphism.

Suppose given $n \geq 1$ and a graph morphism $c' : C_n \rightarrow X'$.

We have to show that there exists a unique graph morphism $c : C_n \rightarrow X$ with $cg = c'$.

First, we remark that there exists a unique graph morphism $w : C_n \rightarrow Y$ such that $wh = c'f'$ because h is a quasiisomorphism.

Existence.

Because X is a pullback, we may choose a graph morphism $c : C_n \rightarrow X$ such that $cf = w$ and $cg = c'$.

Uniqueness.

Now we have to show that c is unique with respect to $cg = c'$.

Suppose given a graph morphism $\tilde{c} : C_n \rightarrow X$ such that $\tilde{c}g = c'$.

We have to show that $c \stackrel{!}{=} \tilde{c}$.

We have $\tilde{c}fh = \tilde{c}g f' = c'f'$ and thus $\tilde{c}f = w$.

So we have $cf = \tilde{c}f$.

And we have $cg = \tilde{c}g$.

So we have $c = \tilde{c}$; cf. Remark 10.

$$\begin{array}{ccccc} & & X & \xrightarrow{f} & Y \\ & \nearrow \exists! c & \downarrow g & \ulcorner & \downarrow h \\ C_n & \xrightarrow{c'} & X' & \xrightarrow{f'} & Y' \\ & & & & \downarrow h \end{array}$$

\square

Question 126 Suppose given a graph morphism $f : G \rightarrow H$ with G and H finite.

- (1) Is it possible to give an algorithm to calculate a number $\zeta_f \in \mathbb{Z}_{\geq 1}$ such that

$$\min\{n \in \mathbb{N} : (C_n, f) \text{ is not bijective}\} \leq \zeta_f$$

if f is not a quasiisomorphism?

An affirmative answer would allow to algorithmically decide whether f is a quasiisomorphism; for then f would be a quasiisomorphism if and only if $(C_n, f)_{\text{Gph}}$ is bijective for $n \in [1, \zeta_f]$.

If f is not a quasiisomorphism, then the left hand side

$$\min\{n \in \mathbb{N} : (C_n, f) \text{ is not bijective}\}$$

seems to be difficult to calculate.

We can only give an algorithm that verifies a sufficient condition for f to be an acyclic fibration, in particular a quasiisomorphism; cf. Proposition 210. The function `SuffCond` is given in §10.6.

- (2) Experiments indicate that $\zeta_f = \max\{|E_G|, |E_H|\}$ could be a possible choice in (1).

3.2 Fibrations and fibrant graphs

Definition 127 Suppose given graphs G and H .

- (1) A graph morphism $f = (V_f, E_f) : G \rightarrow H$ is called a *fibration* if the map

$$E_{f,v} := E_f|_{G(v,*)}^{H(vV_f,*)} : G(v,*) \rightarrow H(vV_f,*)$$

is surjective for $v \in V_G$.

To indicate that f is a fibration, we often write $G \xrightarrow{f} H$.

By $\text{Fib}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$ we denote the set of fibrations in the category Gph .

We often write $\text{Fib} := \text{Fib}(\text{Gph})$.

- (2) A graph morphism $f = (V_f, E_f) : G \rightarrow H$ is called an *etale fibration* if the map

$$E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$$

is bijective for $v \in V_G$.

By $\text{EtaleFib}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$ we denote the set of etale fibrations in the category Gph .

The Assertion 249 below shows that being surjective is not a sufficient condition for a graph morphism to be a fibration.

Example 128 Suppose given a graph H .

The morphism $\iota_H : \emptyset \rightarrow H$ is a fibration; cf. Definitions 70 and 127.(1).

We have to show that the map

$$E_{h,v} = E_h |_{\emptyset(v,*)}^{H(vV_h,*)} : \emptyset(v,*) \rightarrow H(vV_h,*)$$

is surjective for $v \in V_\emptyset$.

The set $V_\emptyset = \emptyset$ does not contain any element, so there is nothing to show and the condition for h to be a fibration is satisfied. \square

Remark 129 We have $\text{Iso}(\text{Gph}) \subseteq \text{EtaleFib}(\text{Gph}) \subseteq \text{Fib}(\text{Gph})$.

Proof.

Ad $\text{EtaleFib}(\text{Gph}) \stackrel{!}{\subseteq} \text{Fib}(\text{Gph})$. Suppose given an etale fibration $f : G \rightarrow H$ in $\text{EtaleFib}(\text{Gph})$.

Then the map

$$E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$$

is bijective and thus surjective for $v \in V_G$; cf. Definition 127. So the graph morphism $f : G \rightarrow H$ is a fibration and thus in Fib .

Ad $\text{Iso}(\text{Gph}) \stackrel{!}{\subseteq} \text{EtaleFib}(\text{Gph})$. Suppose given an isomorphism $f : G \rightarrow H$ in $\text{Iso}(\text{Gph})$.

We have to show that the isomorphism $f : G \rightarrow H$ is an etale fibration.

Therefor we have to show that the map

$$E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$$

is bijective for $v \in V_G$.

Because the graph morphism $f : G \rightarrow H$ is an isomorphism, the map $E_f : E_G \rightarrow E_H$ is bijective.

Suppose given a vertex $v \in V_G$.

Since we have $E_{f,v} = E_f |_{G(v,*)}^{H(vV_f,*)}$, the map $E_{f,v}$ is injective because the map E_f is injective.

We have to show that the map $E_{f,v}$ is surjective.

Suppose given an edge $e \in H(vV_f,*) \subseteq E_H$.

We have $e s_H = v V_f$.

We have to find an edge $e_G \in G(v,*) \subseteq E_G$ such that $e_G E_{f,v} = e$.

Let $e_G := e E_{f^{-1}}$. Then $e_G E_f = e E_{f^{-1}} E_f = e$.

Moreover, $e_G \in G(v,*)$ since $e_G s_G = e E_{f^{-1}} s_G = e s_H V_{f^{-1}} = v V_f V_{f^{-1}} = v$.

So the map $E_{f,v} : G(v,*) \rightarrow H(vV_f,*)$ is bijective. \square

Remark 130 In Gph , the subset of fibrations $\text{Fib} \subseteq \text{Mor}$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc} & G' & \xrightarrow{f'} & H' \\ & \uparrow p & & \uparrow q \\ \text{id}_{G'} & \left(\begin{array}{ccc} G & \xrightarrow{f} & H \\ \uparrow i & & \uparrow j \\ G' & \xrightarrow{f'} & H' \end{array} \right) & & \text{id}_{H'} \end{array}$$

We have to show that the graph morphism $f' : G' \rightarrow H'$ is a fibration; cf. Definition 127.(1).

Since the graph morphism $f : G \rightarrow H$ is a fibration, given a vertex $v \in V_G$, the map $E_{f,v} : G(v, *) \rightarrow H(v V_f, *)$ is surjective; cf. Definition 127.(1).

Suppose given a vertex $v' \in V_{G'}$.

We have to show that the map $E_{f',v'} : G'(v', *) \rightarrow H'(v' V_{f'}, *)$ is surjective.

Suppose given an edge $e' \in H'(v' V_{f'}, *)$.

We have to show that there exists an edge $\hat{e}' \in G'(v', *)$ such that $\hat{e}' E_{f',v'} \stackrel{!}{=} e'$.

Since $e' E_j s_H = e s_{H'} V_j = v V_{f'} V_j = v V_{f'j}$ we have $e' E_j \in H(v' V_{f'j}, *)$.

Let $v := v' V_i$. We have $v V_f = v' V_i V_f = v' V_{if} = v' V_{f'j}$. So we have $e' E_j \in H(v V_f, *)$.

Since the graph morphism $f : G \rightarrow H$ is a fibration, there exists an edge $\hat{e} \in G(v, *)$ such that $\hat{e} E_f = e' E_j$.

Let $\hat{e}' := \hat{e} E_p$. Then we have $\hat{e}' s_{G'} = \hat{e} E_p s_{G'} = \hat{e} s_G V_p = v V_p = v' V_i V_p = v' V_{ip} = v' V_{\text{id}_{G'}} = v'$. So we have $\hat{e}' \in G'(v', *)$.

And we have $\hat{e}' E_{f'} = \hat{e} E_p E_{f'} = \hat{e} E_{pf'} = \hat{e} E_{fq} = \hat{e} E_f E_q = e' E_j E_q = e' E_{jq} = e' E_{\text{id}_{H'}} = e'$.

So the graph morphism $f' : G' \rightarrow H'$ is a fibration. \square

Once Lemma 192 below is known, which states that we have $\text{Fib} = \text{AcCofib}^{\square}$, we can also argue by Remark 26 to obtain the statement of Remark 130.

Remark 131 Suppose given fibrations $f : X \twoheadrightarrow Y$ and $g : Y \twoheadrightarrow Z$.

Then the composite $fg : X \rightarrow Z$ is also a fibration.

Proof.

The map $E_{f,v} : X(v, *) \rightarrow Y(v V_f, *)$ is surjective for $v \in V_X$.

The map $E_{g,v V_f} : Y(v V_f, *) \rightarrow Z(v V_f V_g, *)$ is surjective for $v \in V_X$.

We have to show that the map $E_{fg,v} : X(v, *) \rightarrow Z(v V_{fg}, *)$ is surjective for $v \in V_X$.

We *claim* that $E_{f,v} \cdot E_{g,v V_f} \stackrel{!}{=} E_{fg,v}$.

Suppose given $e \in X(v, *)$, i.e. $e \in E_X$ with $e s_X = v$.

Then $e E_{f,v} \cdot E_{g,v V_f} = e E_f E_g = e E_{fg} = e E_{fg,v}$.

This proves the *claim*.

Now since $E_{f,v}$ and $E_{g,v V_f}$ are surjective, so is $E_{fg,v}$. \square

Once Lemma 192 below is known, which states that we have $\text{Fib} = \text{AcCofib}^{\square}$, we can also argue by Remark 20 to obtain the statement of Remark 131.

Remark 132 Suppose given etale fibrations $f : X \rightarrow Y$ and $g : Y \rightarrow Z$.

Then the composite $fg : X \rightarrow Z$ is also an etale fibration.

Proof. The map $E_{f,v} : X(v, *) \rightarrow Y(v V_f, *)$ is bijective for $v \in V_X$.

The map $E_{g,v V_f} : Y(v V_f, *) \rightarrow Z(v V_f V_g, *)$ is bijective for $v \in V_X$.

We have to show that the map $E_{fg,v} : X(v, *) \rightarrow Z(v V_{fg}, *)$ is bijective for $v \in V_X$.

We have $E_{f,v} \cdot E_{g,v V_f} = E_{fg,v}$; cf. proof of Remark 131.

Now since $E_{f,v}$ and $E_{g,v V_f}$ are bijective, so is $E_{fg,v}$. \square

Remark 133 In Gph , a pullback of a fibration is a fibration; cf. Definition 127.(1).

Proof. Suppose given a pullback in Gph as follows.

$$\begin{array}{ccc} X' & \xrightarrow{g} & X \\ f' \downarrow & \ulcorner & \downarrow f \\ Y' & \xrightarrow{h} & Y \end{array}$$

We have to show that the graph morphism $f' : X' \rightarrow Y'$ is a fibration.

Suppose given $v_{X'} \in V_{X'}$.

We write $v_{Y'} := v_{X'} V_{f'} \in V_{Y'}$ and $v_X := v_{X'} V_g \in V_X$ and

$$v_Y := v_{Y'} V_h = v_{X'} V_{f'} V_h = v_{X'} V_{f'h} \stackrel{f'h=gf}{=} v_{X'} V_{gf} = v_{X'} V_g V_f = v_X V_f \in V_Y .$$

We have to show that the map

$$E_{f',v_{X'}} := E_{f'} \Big|_{X'(v_{X'}, *)}^{Y'(v_{Y'}, *)} : X'(v_{X'}, *) \rightarrow Y'(v_{Y'}, *)$$

is surjective.

Suppose given an edge $e_{Y'} \in Y'(v_{Y'}, *) \subseteq E_{Y'}$.

We have to show that there exists an edge $e_{X'} \in X'(v_{X'}, *) \subseteq E_{X'}$ such that $e_{X'} E_{f'} = e_{Y'}$.

We have $v_{Y'} = e_{Y'} s_{Y'}$. We write $w_{Y'} := e_{Y'} t_{Y'}$.

We write $e_Y := e_{Y'} E_h \in Y(v_{Y'} V_h, *) = Y(v_Y, *)$.

We write $w_Y := w_{Y'} V_h = e_{Y'} t_{Y'} V_h = e_{Y'} E_h t_Y = e_Y t_Y$.

We have $v_Y = v_{Y'} V_h = e_{Y'} s_{Y'} V_h = e_{Y'} E_h s_Y = e_Y s_Y$.

Because the graph morphism $f : X \dashrightarrow Y$ is a fibration, the map $E_{f, v_X} = E_f|_{X(v_X, *)}^{Y(v_Y, *)}$ is surjective.

So there exists an edge $e_X \in X(v_X, *) \subseteq E_X$ such that $e_X E_f = e_Y$.

Note that $e_X s_X = v_X$. We write $w_X := e_X t_X$.

Note that $w_X V_f = e_X t_X V_f = e_X E_f t_Y = e_Y t_Y = w_Y$.

Recall that we have to show that there exists an edge $e_{X'} \in X'(v_{X'}, *) \subseteq E_{X'}$, i.e. $e_{X'} s_{X'} = v_{X'}$, such that $e_{X'} E_{f'} = e_{Y'}$.

We consider the direct graph D_1 with $V_{D_1} = \{\hat{v}_0, \hat{v}_1\}$ and $E_{D_1} = \{\hat{e}_0\}$ and $\hat{e}_0 s_{D_1} = \hat{v}_0$ and $\hat{e}_0 t_{D_1} = \hat{v}_1$; cf. Definition 56.

We define the graph morphism $q : D_1 \rightarrow X$ by $\hat{v}_0 V_q := v_X$, $\hat{v}_1 V_q := w_X$ and $\hat{e}_0 E_q := e_X$.

Note that $\hat{e}_0 E_q s_X = e_X s_X = v_X = \hat{v}_0 V_q = \hat{e}_0 s_{D_1} V_q$ and that $\hat{e}_0 E_q t_X = e_X t_X = w_X = \hat{v}_1 V_q = \hat{e}_0 t_{D_1} V_q$. So $q : D_1 \rightarrow X$ is in fact a graph morphism.

We define the graph morphism $p : D_1 \rightarrow Y'$ with $\hat{v}_0 V_p := v_{Y'}$, $\hat{v}_1 V_p := w_{Y'}$ and $\hat{e}_0 E_p := e_{Y'}$.

Note that $\hat{e}_0 E_p s_{Y'} = e_{Y'} s_{Y'} = v_{Y'} = \hat{v}_0 V_p = \hat{e}_0 s_{D_1} V_p$ and that $\hat{e}_0 E_p t_{Y'} = e_{Y'} t_{Y'} = w_{Y'} = \hat{v}_1 V_p = \hat{e}_0 t_{D_1} V_p$. So $p : D_1 \rightarrow Y'$ is in fact a graph morphism.

We have $\hat{v}_0 V_{qf} = v_X V_f = v_Y = v_{Y'} V_h = \hat{v}_0 V_{ph}$.

We have $\hat{v}_1 V_{qf} = w_X V_f = w_Y = w_{Y'} V_h = \hat{v}_1 V_{ph}$.

We have $\hat{e}_0 E_{qf} = e_X E_f = e_Y = e_{Y'} E_h = \hat{e}_0 E_{ph}$.

So we have $qf = ph$.

Because X' is a pullback there exists a unique graph morphism $r : D_1 \rightarrow X'$ such that $rg = q$ and $rf' = p$.

We let $e_{X'} := \hat{e}_0 E_r \in E_{X'}$. We have $e_{X'} E_{f'} = \hat{e}_0 E_r E_{f'} = \hat{e}_0 E_{rf'} = \hat{e}_0 E_p = e_{Y'}$.

We will show that $e_{X'} \stackrel{\dagger}{\in} X'(v_{X'}, *)$, i.e. that $e_{X'} s_{X'} = \hat{e}_0 E_r s_{X'} = \hat{e}_0 s_{D_1} V_r = \hat{v}_0 V_r \stackrel{\dagger}{=} v_{X'}$.

We consider the subgraph $D_0 \subseteq D_1$ with $V_{D_0} = \{\hat{v}_0\}$ and $E_{D_0} = \emptyset$; cf. Definition 56.(1).

We consider the graph morphism $\iota_1 : D_0 \rightarrow D_1$ with $\hat{v}_0 V_{\iota_1} = \hat{v}_0$; cf. Definition 56.(3).

We write $p' := \iota_1 p = \iota_1 r f'$ and $q' := \iota_1 q = \iota_1 r g$.

We have $p'h = \iota_1 r f' h = \iota_1 r g f = q' f$.

So there exists a unique graph morphism $r' : D_0 \rightarrow X'$ such that $r' f' = p'$ and $r' g = q'$.

We define the graph morphism $\tilde{r} : D_0 \rightarrow X$ by $\hat{v}_0 V_{\tilde{r}} := v_{X'}$.

We show that we have $\tilde{r} f' \stackrel{\dagger}{=} p'$ and $\tilde{r} g \stackrel{\dagger}{=} q'$.

We have $\hat{v}_0 V_{\tilde{r} f'} = v_{X'} V_{f'} = v_{Y'} = \hat{v}_0 V_p = \hat{v}_0 V_{\iota_1 p} = \hat{v}_0 V_{p'}$. So we have $\tilde{r} f' = p'$.

We have $\hat{v}_0 V_{\tilde{r} g} = v_{X'} V_g = v_X = \hat{v}_0 V_q = \hat{v}_0 V_{\iota_1 q} = \hat{v}_0 V_{q'}$. So we have $\tilde{r} g = q'$.

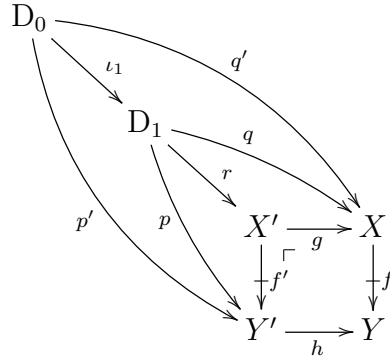
Because of the uniqueness of r' , we obtain $r' = \tilde{r}$.

We consider the graph morphism $\iota_1 r : D_0 \rightarrow X$. We have $(\iota_1 r) f' = p'$ and $(\iota_1 r) g = q'$.

Because of the uniqueness of r' , we obtain $r' = \iota_1 r$.

Altogether, we have $\iota_1 r = \tilde{r}$. In particular $\hat{v}_0 V_r = \hat{v}_0 V_{\iota_1 r} = \hat{v}_0 V_{\tilde{r}} = v_{X'}$.

So the graph morphism $f' : X' \rightarrow Y'$ is a fibration.



□

Once Lemma 192 below is known, which states that we have $\text{Fib} = \text{AcCofib}^{\square}$, we can also argue by Remark 22 to obtain the statement of Remark 133.

Remark 134 Suppose given a thin graph Y and an etale fibration $X \xrightarrow{f} Y$; cf. Definitions 73 and 127.(2).

Then the graph X is thin.

Proof. We assume that the graph X is not thin.

Then we may choose two edges $e_1 \neq e_2$ in E_X such that $e_1 s_X = e_2 s_X$ and $e_1 t_X = e_2 t_X$.

Let $v_1 := e_1 s_X$. Let $v_2 := e_1 t_X$.

Since the graph morphism $X \xrightarrow{f} Y$ is an etale fibration, the map

$$E_{f,v_1} : X(v_1, *) \rightarrow Y(v_1 V_f, *)$$

is bijective.

Since $v_1 = e_1 s_X = e_2 s_X$ we have $e_1, e_2 \in X(v_1, *) = X(e_1 s_X, *) = X(e_2 s_X, *)$.

We have $e_1 E_{f,v_1}, e_2 E_{f,v_1} \in Y(v_1 V_f, *)$. Since the map E_{f,v_1} is bijective, we have $e_1 E_f = e_1 E_{f,v_1} \neq e_2 E_{f,v_1} = e_2 E_f$.

Because $f : X \rightarrow Y$ is a graph morphism we have $e_1 E_f t_Y = e_1 t_X V_f = v_2 V_f = e_2 t_X V_f = e_2 E_f t_Y$.

So we have $e_1 E_f \neq e_2 E_f$ in $Y(v_1 V_f, v_2 V_f)$, which is a *contradiction* to the fact that the graph Y is thin. □

Definition 135 Suppose given a graph X . The graph X is called *fibrant* if the graph morphism $\tau_X : X \rightarrow C_1$ is a fibration; cf. Remark 70 and Definition 127.(1).

Remark 136 Suppose given a graph X .

The graph X is fibrant if and only if $X(v, *) \neq \emptyset$ for $v \in V_X$.

Proof. The graph X is fibrant if and only if the graph morphism $\tau_X : X \rightarrow C_1$ is a fibration, i.e. if and only if the map

$$E_{\tau_X, v} := E_{\tau_X} \Big|_{X(v, *)}^{C_1(v V_{\tau_X}, *)} : X(v, *) \rightarrow C_1(v V_{\tau_X}, *)$$

is surjective for $v \in V_X$; cf. Definitions 127.(1) and 135.

For $v \in V_X$ we have $v V_{\tau_X} = v_1$ and so $C_1(v V_{\tau_X}, *) = C_1(v_1, *) = \{e_1\}$.

So $E_{\tau_X, v}$ is surjective for $v \in V_X$ if and only if $X(v, *) \neq \emptyset$ for $v \in V_X$. \square

Note that the empty graph \emptyset is fibrant; cf. Remark 70.(2).

Example 137 Suppose given $n \in \mathbb{N}$.

The cyclic graph C_n is fibrant; cf. Definitions 52 and 135 and Remark 136.

The direct graph D_n is **not** fibrant, since $e t_{D_n} \neq \hat{v}_n$ for $e \in E_{D_n}$.

3.3 Acyclic fibrations

Recall the notion of quasiisomorphisms from Definition 115 and of fibrations from Definition 127.(1).

Definition 138 A fibration that is a quasiisomorphism is called an *acyclic fibration*; cf. Definitions 115 and 127.(1).

To indicate that a graph morphism $G \xrightarrow{f} H$ is an acyclic fibration, we often write $G \xrightarrow[\dashv]{f} H$.

By $\text{AcFib} := \text{AcFib}(\text{Gph}) := \text{Fib} \cap \text{Qis} \subseteq \text{Mor}(\text{Gph})$ we denote the set of acyclic fibrations in the category Gph .

Remark 139 We have $\text{Iso}(\text{Gph}) \subseteq \text{AcFib}(\text{Gph})$.

Proof. In Gph , we have $\text{Iso} \subseteq \text{Fib} \cap \text{Qis} \stackrel{\text{Def. 138}}{=} \text{AcFib}$ since $\text{Iso} \subseteq \text{Fib}$ by Remark 129 and $\text{Iso} \subseteq \text{Qis}$ by Remark 116. \square

Example 140 We consider the graph morphism $\iota_{D_0} : \emptyset \rightarrow D_0$.

Recall from Example 128 that ι_{D_0} is a fibration.

Recall from Example 118 that ι_{D_0} is a quasiisomorphism.

So $\iota_{D_0} : \emptyset \rightarrow D_0$ is an acyclic fibration.

Remark 141 In Gph , the subset of acyclic fibrations $\text{AcFib} \subseteq \text{Mor}$ is closed under retracts; cf. Definition 23.

Proof. Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc}
 G' & \xrightarrow{f'} & H' \\
 \uparrow p & & \uparrow q \\
 G & \xrightarrow{f} & H \\
 \uparrow i & & \uparrow j \\
 G' & \xrightarrow{f'} & H'
 \end{array}
 \begin{array}{l}
 \text{id}_{G'} \\
 \text{id}_{H'}
 \end{array}$$

We have to show that the graph morphism $f' : G' \rightarrow H'$ is an acyclic fibration; cf. Definition 138.

Since the subset of quasiisomorphisms $\text{Qis} \subset \text{Mor}(\text{Gph})$ is closed under retracts the graph morphism $f' : G' \rightarrow H'$ is a quasiisomorphism; cf. Remark 117.

Since the subset of fibrations $\text{Fib} \subset \text{Mor}(\text{Gph})$ is closed under retracts the graph morphism $f' : G' \rightarrow H'$ is a fibration; cf. Remark 130.

So the subset of acyclic fibrations $\text{AcFib} \subseteq \text{Mor}(\text{Gph})$ is closed under retracts, since the graph morphism $f' : G' \rightarrow H'$ is an acyclic fibration; cf. Definition 23. \square

Once Lemma 193 below is known, which states that we have $\text{AcFib} = \text{Cofib}^{\square}$, we can also argue by Remark 26 to obtain the statement of Remark 141.

Remark 142 Suppose given acyclic fibrations $X \xrightarrow{f} Y$ and $Y \xrightarrow{g} Z$.

Then the composite $fg : X \rightarrow Z$ is also an acyclic fibration.

Proof. Since f and g are fibrations, so is fg ; cf. Definition 127.(1) and Remark 131.

Since f and g are quasiisomorphisms, so is fg ; cf. Definition 115 and Remark 122.

So the graph morphism $fg : X \rightarrow Z$ is a fibration and a quasiisomorphism. Hence fg is an acyclic fibration. \square

Once Lemma 193 below is known, which states that we have $\text{AcFib} = \text{Cofib}^{\square}$ we can also argue by Remark 20 to obtain the statement of Remark 142.

Remark 143 In Gph, a pullback of an acyclic fibration is an acyclic fibration; cf. Definition 138.

Proof. Recall that $\text{AcFib} = \text{Fib} \cap \text{Qis}$. I.e. a morphism is an acyclic fibration if and only if it is a fibration and a quasiisomorphism.

A pullback of a fibration is a fibration; cf. Remark 133.

A pullback of a quasiisomorphism is a quasiisomorphism; cf. Remark 125.

So a pullback of an acyclic fibration is an acyclic fibration. \square

Once Lemma 193 below is known, which states that we have $\text{AcFib} = \text{Cofib}^{\square}$, we can also argue by Remark 22 to obtain the statement of Remark 143.

3.4 Cofibrations and cofibrant graphs

Definition 144 A graph morphism $f : X \rightarrow Y$ is called a *cofibration* if it satisfies $(\text{LLP}_{\text{AcFib}})$; cf. Definitions 13 and 138.

To indicate that f is a cofibration, we often write $X \xrightarrow{\bullet} Y$.

By

$$\text{Cofib}(\text{Gph}) := \square \text{AcFib}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$$

we denote the set of cofibrations in the category Gph .

We often write $\text{Cofib} := \text{Cofib}(\text{Gph})$.

Remark 145 We have $\text{Iso}(\text{Gph}) \subseteq \text{Cofib}(\text{Gph})$.

Proof. By definition we have $\text{Cofib} = \square \text{AcFib}$.

We have $\text{Iso} \subseteq \square \text{AcFib} = \text{Cofib}$; cf. Remark 18. □

Remark 146 In Gph , the subset of cofibrations $\text{Cofib} \subseteq \text{Mor}$ is closed under retracts; cf. Definition 23.

Proof. Since, by definition, $\text{Cofib} = \square \text{AcFib}$ we can argue by Remark 25. □

For an example of a cofibration we refer to Remark 155, which makes use of Remark 151.

Remark 147 Suppose given cofibrations $f : X \xrightarrow{\bullet} Y$ and $g : Y \xrightarrow{\bullet} Z$.

Then the composite $fg : X \rightarrow Z$ is also a cofibration.

Proof. Since $\text{Cofib} = \square \text{AcFib}$, this follows by Remark 19.

Remark 148 In Gph , a pushout of a cofibration is a cofibration.

Proof. Suppose given a pushout in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array} \quad \lrcorner$$

We have to show that the graph morphism $h : Y \rightarrow Y'$ is a cofibration; cf. Definition 144.

We have to show that the graph morphism h satisfies $(\text{LLP}_{\text{AcFib}})$; cf. Definitions 144, 13 and 138.

Suppose given the following commutative diagram.

$$\begin{array}{ccccc} X & \xrightarrow{f} & Y & \xrightarrow{u} & Z \\ g \downarrow & & \downarrow h & & \downarrow k \\ X' & \xrightarrow{f'} & Y' & \xrightarrow{v} & Z' \end{array} \quad \lrcorner$$

We have to show that there exists a graph morphism $\tilde{w} : Y' \rightarrow Z$ such that $h\tilde{w} = u$ and $\tilde{w}k = v$.

Because the graph morphism $X \xrightarrow{g} X'$ is a cofibration there exists a graph morphism $w : X' \rightarrow Z$ such that $gw = fu$ and $wk = f'v$.

Because Y' is a pushout and $gw = fu$ there exists a graph morphism $\tilde{w} : Y' \rightarrow Z$ such that $f'\tilde{w} = w$ and $h\tilde{w} = u$.

It remains to show that $\tilde{w}k \stackrel{!}{=} v$.

We have $f'\tilde{w}k = wk = f'v$ and $h\tilde{w}k = uk = hv$.

Cancelling f' and h simultaneously using Remark 6, we obtain $\tilde{w}k = v$.

$$\begin{array}{ccccc}
 X & \xrightarrow{f} & Y & \xrightarrow{u} & Z \\
 \downarrow g & & \downarrow h & \nearrow \tilde{w} & \downarrow k \\
 X' & \xrightarrow{f'} & Y' & \xrightarrow{v} & Z'
 \end{array}$$

□

Using Definition 144 where we have $\text{Cofib}(\text{Gph}) := \square \text{AcFib}(\text{Gph})$ and Remark 21 the set $\text{Cofib}(\text{Gph})$ is stable under pushouts.

Lemma 149 Suppose given a set I and cofibrations $X_i \xrightarrow{g_i} Y_i$ for $i \in I$.

Then the graph morphism $\coprod_{i \in I} g_i : \coprod_{i \in I} X_i \rightarrow \coprod_{i \in I} Y_i$ is a cofibration; cf. Definition 93.

Proof. Suppose given a commutative quadrangle

$$\begin{array}{ccc}
 \coprod_{i \in I} X_i & \xrightarrow{(a_i)_i} & G \\
 \coprod_{i \in I} g_i \downarrow & & \downarrow f \\
 \coprod_{i \in I} Y_i & \xrightarrow{(b_i)_i} & H
 \end{array}$$

in Gph with $a_i : X_i \rightarrow G$ and $b_i : Y_i \rightarrow H$ for $i \in I$.

We have to show that there exists a graph morphism $(h_i)_i : \coprod_{i \in I} Y_i \rightarrow G$ such that $(\coprod_{i \in I} g_i) \cdot (h_i)_i = (a_i)_i$ and $(h_i)_i \cdot f = (b_i)_i$.

We have $(g_i \cdot b_i)_i \stackrel{\text{Rem. 94}}{=} (\coprod_{i \in I} g_i) \cdot (b_i)_i = (a_i)_i \cdot f \stackrel{\text{Rem. 92}}{=} (a_i \cdot f)_i : \coprod_{i \in I} X_i \rightarrow H$.

So we have $g_i \cdot b_i = a_i \cdot f$ for $i \in I$.

$$\begin{array}{ccc}
 X_i & \xrightarrow{a_i} & G \\
 g_i \downarrow & & \downarrow f \\
 Y_i & \xrightarrow{b_i} & H
 \end{array}$$

Since the graph morphism $g_i : X_i \rightarrow Y_i$ is a cofibration, there exists a graph morphism $h_i : Y_i \rightarrow G$ such that $g_i \cdot h_i = a_i$ and $h_i \cdot f = b_i$, for each $i \in I$.

$$\begin{array}{ccc}
 X_i & \xrightarrow{a_i} & G \\
 g_i \downarrow & \nearrow h_i & \downarrow f \\
 Y_i & \xrightarrow{b_i} & H
 \end{array}$$

So we have the graph morphism $(h_i)_i : \coprod_{i \in I} Y_i \rightarrow G$, where $h_i : Y_i \rightarrow G$.

We have $(\coprod_{i \in I} g_i) \cdot (h_i)_i \stackrel{\text{Rem. 94}}{=} (g_i \cdot h_i)_i = (a_i)_i$ and $(h_i)_i \cdot f \stackrel{\text{Rem. 92}}{=} (h_i \cdot f)_i = (b_i)_i$.

Therefore, the following diagram commutes.

$$\begin{array}{ccc}
 \coprod_{i \in I} X_i & \xrightarrow{(a_i)_i} & G \\
 \downarrow \coprod_{i \in I} g_i & \nearrow (h_i)_i & \downarrow f \\
 \coprod_{i \in I} Y_i & \xrightarrow{(b_i)_i} & H
 \end{array}$$

So the graph morphism $\coprod_{i \in I} g_i : \coprod_{i \in I} X_i \rightarrow \coprod_{i \in I} Y_i$ is a cofibration. \square

Definition 150 Suppose given a graph X . The graph X is called *cofibrant* if the graph morphism $\iota_X : \emptyset \rightarrow X$ is a cofibration; cf. Remark 70 and Definition 144.

Remark 151 Suppose given $n \in \mathbb{N}$.

The cyclic graph C_n is cofibrant; cf. Definitions 52 and 150.

Proof. Suppose given

$$\begin{array}{ccc}
 \emptyset & \xrightarrow{u} & X \\
 \downarrow \iota_{C_n} & & \downarrow f' \\
 C_n & \xrightarrow{v} & Y
 \end{array}$$

in Gph such that $uf' = \iota_{C_n}v$.

Since the graph morphism $X \xrightarrow{f'} Y$ is an acyclic fibration, f' is in particular a quasiisomorphism; cf. Definition 138. Since $v \in (C_n, Y) \simeq (C_n, X)$, there exists a graph morphism $g : C_n \rightarrow X$ such that $gf' = v$.

We have $\iota_{C_n}g = \iota_X = u$.

So the graph morphism $\iota_{C_n} : \emptyset \rightarrow C_n$ is a cofibration and so the cyclic graph C_n is cofibrant. \square

Remark 152 Suppose given a set I .

Suppose given cofibrant graphs X_i for $i \in I$.

Then the coproduct $\coprod_{i \in I} X_i$ is cofibrant.

Proof. Since the graphs X_i are cofibrant, the graph morphisms $\iota_{X_i} : \emptyset \rightarrow X_i$ are cofibrations for $i \in I$.

Since $\coprod_{i \in I} \emptyset = \emptyset$, the graph morphism $\iota_{\coprod_{i \in I} Y_i} : \emptyset \rightarrow \coprod_{i \in I} Y_i$ is a cofibration; cf. Lemma 149.

So the coproduct $\coprod_{i \in I} Y_i$ is cofibrant. \square

Example 153 The cyclic graph C_2 is cofibrant; cf. Remark 151.

Thus the coproduct $C_2 \sqcup C_2$ is cofibrant; cf. Remark 152.

Definition 154 Suppose given the graph X .

We have the coproduct $X \sqcup X$; cf. Definition 85.

We define the *diagonal* graph morphism $d_X := \begin{pmatrix} \text{id}_X \\ \text{id}_X \end{pmatrix} : X \sqcup X \rightarrow X$; cf. Definition 86.

In detail, we have

$$\begin{array}{ccc} X \sqcup X & \xrightarrow{d_X = \begin{pmatrix} \text{id}_X \\ \text{id}_X \end{pmatrix}} & X \\ V_{d_X} : (1, v) & \mapsto & v \quad \text{for } v \in V_X \\ & (2, v) & \mapsto & v \quad \text{for } v \in V_X \\ E_{d_X} : (1, e) & \mapsto & e \quad \text{for } e \in E_X \\ & (2, e) & \mapsto & e \quad \text{for } e \in E_X . \end{array}$$

Remark 155 Suppose given $n \in \mathbb{N}$.

The diagonal graph morphism $d_{C_n} : C_n \sqcup C_n \rightarrow C_n$ is a cofibration, i.e. d_{C_n} satisfies $(\text{LLP}_{\text{AcFib}})$, cf. Definitions 144, 138, 154.

Proof. In this proof we abbreviate $d := d_{C_n}$.

Suppose given the following commutative quadrangle in Gph .

$$\begin{array}{ccc} C_n \sqcup C_n & \xrightarrow{\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}} & X \\ \begin{pmatrix} \text{id}_{C_n} \\ \text{id}_{C_n} \end{pmatrix} = d \downarrow & & \downarrow f \\ C_n & \xrightarrow{a} & Y \end{array}$$

In particular, we have $b_1 f = a$ and $b_2 f = a$.

Since f is a quasiisomorphism, the map $(C_n, f)_{\text{Gph}}$ is bijective and so we have $b_1 = b_2 =: \hat{a}$.

Thus $d\hat{a} = \begin{pmatrix} \text{id}_{C_n} \\ \text{id}_{C_n} \end{pmatrix} \hat{a} = \begin{pmatrix} \hat{a} \\ \hat{a} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$.

Moreover, we have $\hat{a}f = b_1 f = a$.

$$\begin{array}{ccc} C_n \sqcup C_n & \xrightarrow{\begin{pmatrix} b_1 \\ b_2 \end{pmatrix}} & X \\ d \downarrow & \nearrow \hat{a} & \downarrow f \\ C_n & \xrightarrow{a} & Y \end{array}$$

So the diagonal graph morphism $d_{C_n} : C_n \sqcup C_n \rightarrow C_n$ is shown to be a cofibration.

The following lemma generalizes Remark 155.

Lemma 156 Suppose given $n \in \mathbb{N}$.

Suppose given sets M' and M , and a map $\mu : M' \rightarrow M$ in $\text{Mor}(\text{Set})$.

Suppose given a map $\nu : M \rightarrow \mathbb{N}$.

We have the coproducts $\coprod_{m' \in M'} C_{m'\mu}$ and $\coprod_{m \in M} C_{m\nu}$; cf. Definition 90.

We define the graph morphism $j : \coprod_{m' \in M'} C_{m'\mu\nu} \rightarrow \coprod_{m \in M} C_{m\nu}$ as follows.

$$\begin{aligned} j &: \coprod_{m' \in M'} C_{m'\mu\nu} \rightarrow \coprod_{m \in M} C_{m\nu} \\ V_j &: (m', v_i) \mapsto (m'\mu, v_i) \text{ for } i \in \mathbb{Z}/m'\mu\nu\mathbb{Z} \\ E_j &: (m', e_i) \mapsto (m'\mu, e_i) \text{ for } i \in \mathbb{Z}/m'\mu\nu\mathbb{Z} \end{aligned}$$

Then $j = (V_j, E_j) : \coprod_{m' \in M'} C_{m'\mu\nu} \rightarrow \coprod_{m \in M} C_{m\nu}$ is a cofibration; cf. Definition 144.

Proof. Since for an edge $(m', e_i) \in E_{\coprod_{m' \in M'} C_{m'\mu\nu}}$ we have

$$\begin{aligned} (m', e_i) E_j s_{\coprod_{m \in M} C_{m\nu}} &= (m'\mu, e_i) s_{\coprod_{m \in M} C_{m\nu}} = (m'\mu, e_i) s_{C_{m'\mu\nu}} \\ &= (m', e_i) s_{C_{m'\mu\nu}} V_j = (m', e_i) s_{\coprod_{m' \in M'} C_{m'\mu\nu}} V_j \end{aligned}$$

and

$$\begin{aligned} (m', e_i) E_j t_{\coprod_{m \in M} C_{m\nu}} &= (m'\mu, e_i) t_{\coprod_{m \in M} C_{m\nu}} = (m'\mu, e_i) t_{C_{m'\mu\nu}} \\ &= (m', e_i) t_{C_{m'\mu\nu}} V_j = (m', e_i) t_{\coprod_{m' \in M'} C_{m'\mu\nu}} V_j, \end{aligned}$$

the tuple $(V_j, E_j) = j : \coprod_{m' \in M'} C_{m'\mu\nu} \rightarrow \coprod_{m \in M} C_{m\nu}$ is in fact a graph morphism.

Suppose given

$$\begin{array}{ccc} \coprod_{m' \in M'} C_{m'\mu\nu} & \xrightarrow{b} & X \\ j \downarrow & & \downarrow f \\ \coprod_{m \in M} C_{m\nu} & \xrightarrow{a} & Y \end{array}$$

in Gph such that $ja = bf$.

Given $m' \in M'$, we have the graph morphism

$$\begin{aligned} \iota_{m'} &: C_{m'\mu\nu} \rightarrow \coprod_{m' \in M'} C_{m'\mu\nu} \\ V_{\iota_{m'}} &: v_i \mapsto (m', v_i) \\ E_{\iota_{m'}} &: e_i \mapsto (m', e_i). \end{aligned}$$

Since the graph morphism $f : X \dashrightarrow Y$ is an acyclic fibration, in particular a quasiisomorphism, the map $\prod_{m \in M} (C_{m\nu}, f)_{\text{Gph}} : \prod_{m \in M} (C_{m\nu}, X)_{\text{Gph}} \rightarrow \prod_{m \in M} (C_{m\nu}, Y)_{\text{Gph}}$ is bijective.

Hence the map $(\prod_{m \in M} C_{m\nu}, f)_{\text{Gph}} : (\prod_{m \in M} C_{m\nu}, X)_{\text{Gph}} \rightarrow (\prod_{m \in M} C_{m\nu}, Y)_{\text{Gph}}$ is bijective, cf. Remark 95.

So there exists a unique graph morphism $\hat{a} : \prod_{m \in M} C_{m\nu} \rightarrow X$ such that $\hat{a}f = a$.

We have

$$\begin{array}{ccc} & \coprod_{m' \in M'} C_{m'\mu\nu} & \xrightarrow{b} & X \\ & \uparrow \iota_{m'} & \nearrow \hat{a} & \downarrow f \\ C_{m'\mu\nu} & & \coprod_{m \in M} C_{m\nu} & \xrightarrow{a} & Y \end{array}$$

We have to show that $j\hat{a} \stackrel{!}{=} b$.

It suffices to show that $\iota_{m'} j\hat{a} \stackrel{!}{=} \iota_{m'} b$ for $m' \in M'$.

Since the graph morphism $f : X \dashrightarrow Y$ is a quasiisomorphism and thus $(C_{m'\mu\nu}, f)_{\text{Gph}}$ is bijective, it suffices to show that $\iota_{m'}j\hat{a}f \stackrel{!}{=} \iota_{m'}bf$ for $m' \in M'$.

In fact, we have $\iota_{m'}j\hat{a}f = \iota_{m'}ja = \iota_{m'}bf$. \square

Remark 157 Suppose given a graph X and a cofibrant graph Y , i.e. we have $\emptyset \xrightarrow{\iota_Y} Y$.

Then we have the cofibration $\iota_1 : X \rightarrow X \sqcup Y$; cf. Definition 85.

Proof. By Definition 85 and Remark 148 we have

$$\begin{array}{ccc} \emptyset & \xrightarrow{\iota_Y} & Y \\ \downarrow & & \downarrow \\ X & \xrightarrow{\iota_1} & X \sqcup Y \end{array} \quad \lrcorner$$

\square

Using a result obtained **below** in Proposition 210, we show that gluing cofibrant graphs via pushout does not yield a cofibrant graph in general; cf. Assertion 255.

The following Example 158 also shows that fibrations are not necessarily surjective.

Example 158 Suppose given $n \geq 0$.

We consider the graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$; cf. Remark 70.(2), Definition 56.

(1) The graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is an acyclic fibration; cf. Definition 138.

(2) The graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is not a cofibration; cf. Definition 144.

That is, D_n is not cofibrant.

Proof.

Ad (1). Suppose given $k \in \mathbb{N}$.

We have $(C_k, D_n)_{\text{Gph}} = \emptyset = (C_k, \emptyset)_{\text{Gph}}$. So the graph morphism ι_{D_n} is a quasiisomorphism; cf. Definition 115.

Since $V_\emptyset = \emptyset$, the graph morphism ι_{D_n} is a fibration; cf. Definition 127.(1).

So the graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is an acyclic fibration.

Ad (2). We have the commutative quadrangle

$$\begin{array}{ccc} \emptyset & \longrightarrow & \emptyset \\ \iota_{D_n} \downarrow & & \downarrow \iota_{D_n} \\ D_n & \xrightarrow{\text{id}_{D_n}} & D_n \end{array}$$

in Gph ; cf. (1).

But there does not exist a graph morphism $g : D_n \rightarrow \emptyset$.

So the acyclic fibration $\iota_{D_n} : \emptyset \rightarrow D_n$ is not a cofibration and thus the graph D_n is not cofibrant; cf. Definition 150. \square

Once Lemma 185 below is known, which states that we have $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$, we can also argue as follows.

The graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is a quasiisomorphism; cf. Definition 138.

The graph morphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is not an acyclic cofibration since (AcCofib 5) is not satisfied.

Since $\text{AcCofib} \stackrel{\text{Lemma 185}}{=} \text{Cofib} \cap \text{Qis}$, the quasiisomorphism $\iota_{D_n} : \emptyset \rightarrow D_n$ is not a cofibration.

Example 159 Suppose given $n \in \mathbb{N}$.

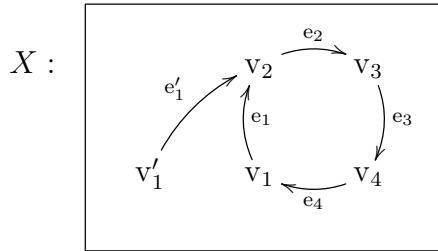
We consider the graph morphism $f : D_0 \rightarrow C_n$ with $\hat{v}_0 V_f := v_1$.

Then the graph morphism $f : D_0 \rightarrow C_n$ is not a cofibration; cf. Definition 144.

Proof. Consider the graph X with $V_X := V_{C_n} \cup \{v'_1\}$ and with $E_X := E_{C_n} \cup \{e'_1\}$ with $e'_1 s_X := v'_1$ and with $e'_1 t_X := v_2$.

We have the acyclic fibration $g : X \twoheadrightarrow C_n$ with $g|_{C_n} := \text{id}_{C_n}$ and with $e'_1 E_g := e_1$ and with $v'_1 V_g := v_1$; cf. Definition 138.

E.g. for $n = 4$ we have



So we have $v_1 V_g = v'_1 V_g = v_1$.

Since the graph morphism $g : X \twoheadrightarrow C_n$ is an acyclic fibration, there exists a unique graph morphism $k : C_n \rightarrow X$ such that $kg = \text{id}_{C_n}$; cf. Definition 138.

We have $k|^{C_n} = \text{id}_{C_n}$.

We have the commutative diagram

$$\begin{array}{ccc}
 D_0 & & X \\
 f \downarrow & \nearrow \exists! k & \downarrow g \\
 C_n & \xrightarrow{\text{id}_{C_n}} & C_n
 \end{array}$$

in Gph.

We will find a graph morphism $p : D_0 \rightarrow X$ such that $f \text{id}_{C_n} = pg$ and such that $fk \stackrel{!}{\neq} p$.

We have $v_1 V_g = v'_1 V_g$ and we have $v_1 V_k = v_1$.

So we let $p : D_0 \rightarrow X$ be the graph morphism with $\hat{v}_0 V_p := v'_1$.

Now we have $f \text{id}_{C_n} = pg$ and we have $kg = \text{id}_{C_n}$. But we have $fk \neq p$ since $\hat{v}_0 V_f V_k = v_1 V_k = v_1 \neq v'_1 = \hat{v}_0 V_p$.

So the graph morphism $f : D_0 \rightarrow C_n$ is not a cofibration. □

3.5 Bifibrant graphs

Definition 160 A graph X is called *bifibrant* if X is fibrant and cofibrant; cf. Definitions 135 and 150.

Example 161 Suppose given $n \in \mathbb{N}$.

The cyclic graph C_n is bifibrant; cf. Example 137 and Remark 151.

3.6 Acyclic cofibrations

Definition 162 An *acyclic cofibration* is a graph morphism $f = (V_f, E_f) : G \rightarrow H$ that satisfies (AcCofib 1–5).

(AcCofib 1) The map $V_f : V_G \rightarrow V_H$ is injective.

(AcCofib 2) The map $E_f : E_G \rightarrow E_H$ is injective.

(AcCofib 3) We have $|\{e \in E_H : (e) \mathfrak{t}_H = v_H\}| = 1$ for $v_H \in V_H \setminus V_{Gf}$.

(AcCofib 4) We have $(e) \mathfrak{t}_H \in V_H \setminus V_{Gf}$ for $e \in E_H \setminus E_{Gf}$.

(AcCofib 5) For $v_H \in V_H \setminus V_{Gf}$ there exist $n \geq 1$ and $e_i \in E_H$ for $i \in [1, n]$ such that $(e_1) \mathfrak{s}_H \in V_{Gf}$, such that $(e_i) \mathfrak{t}_H = (e_{i+1}) \mathfrak{s}_H$ for $i \in [1, n-1]$ and such that $(e_n) \mathfrak{t}_H = v_H$.

To indicate that f is an acyclic cofibration, we often write $G \xrightarrow{f} H$.

By $\text{AcCofib}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$ we denote the set of acyclic cofibrations in the category Gph .

We often write $\text{AcCofib} := \text{AcCofib}(\text{Gph})$.

Remark 163 BISSON and TSEMO [3, Def. 3.2] call the acyclic cofibrations *whiskerings*.

Remark 164 Condition (AcCofib 5) in Definition 162 is equivalent to the following condition (AcCofib 5').

(AcCofib 5') For $v_H \in V_H \setminus V_{Gf}$, there exist $n \geq 1$ and a graph morphism $p : D_n \rightarrow H$ with $\hat{v}_0 V_p \in V_{Gf}$ and $\hat{v}_n V_p = v_H$.

Here, p is a path from a vertex in V_{Gf} to v_H ; cf. Definition 107.

Remark 165 We have $\text{Iso}(\text{Gph}) \subseteq \text{AcCofib}(\text{Gph})$.

Proof. Suppose given a graph isomorphism $f : G \xrightarrow{\sim} H$.

We have to show that $f : G \xrightarrow{\sim} H$ is an acyclic cofibration; cf. Definition 162.

Ad (AcCofib 1–2). The maps $V_f : V_G \rightarrow V_H$ and $E_f : E_G \rightarrow E_H$ are injective; cf. Definition 55.

Ad (AcCofib 3). We have $V_H \setminus V_{Gf} = V_H \setminus V_H = \emptyset$.

Ad (AcCofib 4). We have $E_H \setminus E_{Gf} = E_H \setminus E_H = \emptyset$.

Ad (AcCofib 5). We have $V_H \setminus V_{Gf} = V_H \setminus V_H = \emptyset$. □

Remark 166 Suppose given $0 \leq i \leq k$ in $\mathbb{Z}_{\geq 0}$.

The graph morphism $\iota_{i,k} : D_i \rightarrow D_k$ is an acyclic cofibration; cf. Definition 56.(3).

Proof.

Ad (AcCofib 1–2). We have $\iota_{i,k} = \text{id}_{D_k}|_{D_i}$ and so the maps $V_{\iota_{i,k}} : V_{D_i} \rightarrow V_{D_k}$ and $E_{\iota_{i,k}} : E_{D_i} \rightarrow E_{D_k}$ are injective.

Ad (AcCofib 3). We have $V_{D_k} \setminus V_{D_i \iota_{i,k}} = V_{D_k} \setminus V_{D_i} = \{\hat{v}_j : j \in [i+1, k]\}$.

So suppose given $j \in [i+1, k]$.

We have $|\{e \in E_{D_k} : (e) t_{D_k} = \hat{v}_j\}| = |\{\hat{e}_{j-1}\}| = 1$.

Ad (AcCofib 4). We have $E_{D_k} \setminus E_{D_i \iota_{i,k}} = E_{D_k} \setminus E_{D_i} = \{\hat{e}_j : j \in [i, k-1]\}$.

So suppose given $j \in [i, k-1]$.

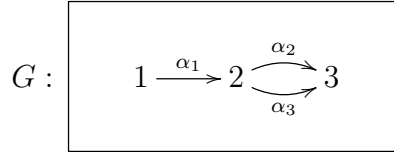
We have $(\hat{e}_j) t_{D_k} = \hat{v}_{j+1} \in V_{D_k} \setminus V_{D_i}$.

Ad (AcCofib 5). We have $V_{D_k} \setminus V_{D_i \iota_{i,k}} = V_{D_k} \setminus V_{D_i} = \{\hat{v}_j : j \in [i+1, k]\}$.

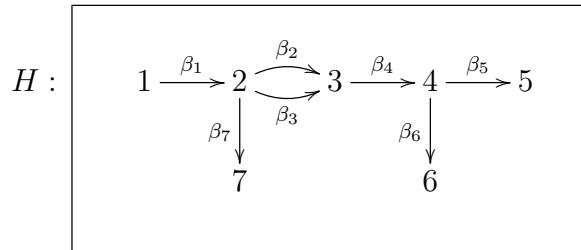
So suppose given $j \in [i+1, k]$.

We let $n := j - i$. Let $e_u := \hat{e}_{i+u-1}$ for $u \in [1, n]$. We consider the edges $\{\hat{e}_u\}_{u \in [i, j-1]}$. We have $e_1 s_{D_k} = \hat{e}_i s_{D_k} = \hat{v}_i \in V_{D_i}$. We have $e_u t_{D_k} = \hat{e}_{i+u-1} t_{D_k} = \hat{v}_{i+u} = \hat{e}_{i+u} s_{D_k} = e_{u+1} s_{D_k}$ for $u \in [1, n-1]$. And we have $e_n t_{D_k} = \hat{e}_{j-1} t_{D_k} = \hat{v}_j$. □

Example 167 We consider the graphs



and



Let $f = (V_f, E_f) : G \rightarrow H$ be the graph morphism with

$$1 V_f = 1, \quad 2 V_f = 2, \quad 3 V_f = 3,$$

and with

$$\alpha_1 E_f = \beta_1, \quad \alpha_2 E_f = \beta_2, \quad \alpha_3 E_f = \beta_3.$$

Then the graph morphism f is an acyclic cofibration; cf. Definition 162.

Proof. We use this opportunity to illustrate the properties (AcCofib 1–5) from Definition 162.

Ad (AcCofib 1, 2).

The maps V_f and E_f both are injective.

Ad (AcCofib 3).

We have to show that $|\{e \in E_H : (e) t_H = v_H\}| = 1$ for $v_H \in V_H \setminus V_{Gf}$.

We have $V_H \setminus V_{Gf} = \{1, 2, 3, 4, 5, 6, 7\} \setminus \{1, 2, 3\} = \{4, 5, 6, 7\}$.

We have $|\{e \in E_H : (e) t_H = 4\}| = |\{\beta_4\}| = 1$ and $|\{e \in E_H : (e) t_H = 5\}| = |\{\beta_5\}| = 1$ and $|\{e \in E_H : (e) t_H = 6\}| = |\{\beta_6\}| = 1$ and $|\{e \in E_H : (e) t_H = 7\}| = |\{\beta_7\}| = 1$.

So (AcCofib 3) holds for f .

Ad (AcCofib 4).

We have to show that $(e) t_H \in V_H \setminus V_{Gf}$ for $e \in E_H \setminus E_{Gf}$.

We have $E_H \setminus E_{Gf} = \{\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6, \beta_7\} \setminus \{\beta_1, \beta_2, \beta_3\} = \{\beta_4, \beta_5, \beta_6, \beta_7\}$.

We have $V_H \setminus V_{Gf} = \{4, 5, 6, 7\}$.

We have $\beta_4 t_H = 4$, $\beta_5 t_H = 5$, $\beta_6 t_H = 6$, $\beta_7 t_H = 7$, which are elements in $V_H \setminus V_{Gf} = \{4, 5, 6, 7\}$, so (AcCofib 4) holds for f .

Ad (AcCofib 5).

We have $V_H \setminus V_{Gf} = \{4, 5, 6, 7\}$.

We have $V_{Gf} = \{1, 2, 3\}$.

For the vertex $4 \in V_H \setminus V_{Gf}$ we may choose $n := 1$ and $e_1 := \beta_4$. Then $e_1 s_H = \beta_4 s_H = 3 \in V_{Gf}$ and $e_1 t_H = \beta_4 t_H = 4$.

For the vertex $5 \in V_H \setminus V_{Gf}$ we may choose $n := 2$ and $e_1 := \beta_4$ and $e_2 := \beta_5 \in E_H$. Then $e_1 s_H = \beta_4 s_H = 3 \in V_{Gf}$ and $e_1 t_H = \beta_4 t_H = 4 = \beta_5 s_H = e_2 s_H$ and $e_2 t_H = \beta_5 t_H = 5$.

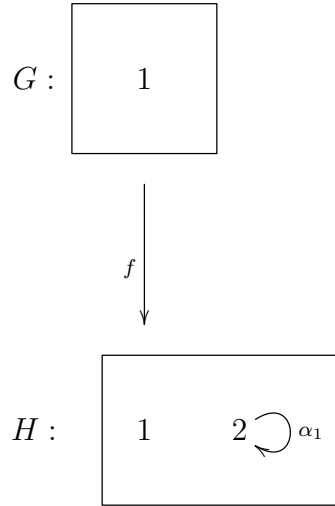
For the vertex $6 \in V_H \setminus V_{Gf}$ we may choose $n := 2$ and $e_1 := \beta_4$ and $e_2 := \beta_6 \in E_H$. Then $e_1 s_H = \beta_4 s_H = 3 \in V_{Gf}$ and $e_1 t_H = \beta_4 t_H = 4 = \beta_6 s_H = e_2 s_H$ and $e_2 t_H = \beta_6 t_H = 6$.

For the vertex $7 \in V_H \setminus V_{Gf}$ we may choose $n := 1$ and $e_1 := \beta_7$. Then $e_1 s_H = \beta_7 s_H = 2 \in V_{Gf}$ and $e_1 t_H = \beta_7 t_H = 7$. \square

Alternatively, via Magma [2] we may proceed as follows, using the functions given in §10 below.

```
G := <[1,2,3], [1,1,2], [2,2,3], [2,3,3]>;
H := <[1,2,3,4,5,6,7], [1,1,2], [2,2,3], [2,3,3], [3,4,4], [4,5,5], [4,6,6], [2,7,7]>;
f := <[1,1], [2,2], [3,3], [[1,1,2], [1,1,2]], [[2,2,3], [2,2,3]], [[2,3,3], [2,3,3]]>;
> AcCofib1to4(f,G,H);
true
> AcCofib5(f,G,H);
true
> IsAcCofib(f,G,H);
true
```

Example 168 We consider the following graph morphism.



We let $1 V_f := 1$.

Then the graph morphism f satisfies (AcCofib 1–4), but **not** (AcCofib 5).

Hence the graph morphism f is **not** an acyclic cofibration.

Proof. We show that (AcCofib 5) is not satisfied.

We *assume* that (AcCofib 5) is satisfied.

So for $v_H \in V_H \setminus V_{Gf}$ there exist $n \geq 1$ and $e_i \in E_H$ for $i \in [1, n]$ such that $e_1 s_H \in V_{Gf}$, such that $e_i t_H = e_{i+1} s_H$ for $i \in [1, n-1]$ and such that $e_n t_H = v_H$.

Then $e_1 s_H = 1$. But there is no edge in H with this property. *Contradiction.*

Alternatively, via Magma we may proceed as follows, using the functions given in §10 below.

```

G := <[1], []>;
H := <[1, 2], [<2, 1, 2>]>;
f := <[<1, 1>], []>;
> IsGraphMorphism(f, G, H);
true
> AcCofib1to4(f, G, H);
true
> AcCofib5(f, G, H);
false
> IsAcCofib(f, G, H);
false

```

□

Remark 169 Suppose given acyclic cofibrations $f : X \twoheadrightarrow Y$ and $g : Y \twoheadrightarrow Z$.

Then the composite fg is also an acyclic cofibration.

Proof.

We have to show that (AcCofib 1–5) hold for the graph morphism fg .

Ad (AcCofib 1).

The composite of injective maps is injective.

The composite of the injective maps V_f and V_g yields the injective map $V_{fg} = V_f \cdot V_g$ and so (AcCofib 1) holds for fg .

Ad (AcCofib 2).

The composite of injective maps is injective.

The composite of the injective maps E_f and E_g yields the injective map $E_{fg} = E_f \cdot E_g$ and so (AcCofib 2) holds for fg .

Ad (AcCofib 3).

We have $|\{e_Y \in E_Y : (e_Y) t_Y = v_Y\}| = 1$ for $v_Y \in V_Y \setminus V_{Xf}$.

We have $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| = 1$ for $v_Z \in V_Z \setminus V_{Yg}$.

We have to show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{=} 1$ for $v_Z \in V_Z \setminus (V_X) V_{fg}$.

We have $V_Z \setminus (V_X) V_{fg} = V_Z \setminus (V_X) V_f V_g \stackrel{\text{Rem. 30}}{=} (V_Z \setminus (V_Y) V_g) \dot{\cup} (V_Y \setminus (V_X) V_f) V_g$, since $V_{fg} = V_f V_g$.

Suppose given $v_Z \in V_Z \setminus (V_X) V_{fg}$. We have to show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{=} 1$.

We consider two cases.

Case 1: $v_Z \in V_Z \setminus (V_Y) V_g$. We have to show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{=} 1$. This follows by g being an acyclic cofibration.

Case 2: $v_Z \in (V_Y \setminus (V_X) V_f) V_g$. We have to show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{=} 1$. There exists $v_Y \in V_Y \setminus (V_X) V_f$ with $(v_Y) V_g = v_Z$. We obtain $|\{e_Y \in E_Y : (e_Y) t_Y = v_Y\}| = 1$ by f being an acyclic cofibration.

We obtain $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{=} 1$:

First, we show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{\geq} 1$.

Suppose given $e_Y \in E_Y$ with $(e_Y) t_Y = v_Y$. Then $(e_Y E_g) t_Z = (e_Y t_Y) V_g = v_Y V_g = v_Z$. So we obtain $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \geq 1$.

Second, we show that $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| \stackrel{!}{\leq} 1$.

Suppose given $e_Z, e'_Z \in E_Z$ with $(e_Z) t_Z = v_Z = (e'_Z) t_Z$. Since $v_Z \in (V_Y \setminus (V_X) V_f) V_g$, we may write $v_Z = v_Y V_g$ with $v_Y \notin (V_X) V_f$.

We have to show that $e_Z \stackrel{!}{=} e'_Z$.

We *claim* $e_Z \in (E_Y) E_g$.

We *assume* $e_Z \notin (E_Y) E_g$. With (AcCofib 4) for g we get $(V_Y \setminus (V_X) V_f) V_g \ni v_Z = (e_Z) t_Z \in V_Z \setminus (V_Y) V_g$, which is a *contradiction*.

This proves the *claim*.

So we have $e_Z \in (E_Y) E_g$. Likewise we have $e'_Z \in (E_Y) E_g$.

There exist $e_Y, e'_Y \in E_Y$ such that $e_Z = e_Y E_g$ and $e'_Z = e'_Y E_g$.

We have to show that $e_Y \stackrel{!}{=} e'_Y$.

But we have $(e_Y t_Y) V_g = (e_Y E_g) t_Z = (e_Z) t_Z = v_Z = (e'_Z) t_Z = (e'_Y E_g) t_Z = (e'_Y t_Y) V_g$.

Moreover, $v_Z = (v_Y) V_g$.

Because V_g is injective we get $e_Y t_Y = e'_Y t_Y = v_Y$.

And so $e_Y, e'_Y \in \{\tilde{e}_Y \in E_Y : (\tilde{e}_Y) t_Y = v_Y\}$ which contains at most one element by (AcCofib 3) for f . So $e_Y = e'_Y$.

This proves $|\{e_Z \in E_Z : (e_Z) t_Z = v_Z\}| = 1$.

So (AcCofib 3) holds for fg .

Ad (AcCofib 4).

We have $(e_Y) t_Y \in V_Y \setminus (V_X) V_f$ for $e_Y \in E_Y \setminus (E_X) E_f$ by (AcCofib 4) for f .

We have $(e_Z) t_Z \in V_Z \setminus (V_Y) V_g$ for $e_Z \in E_Z \setminus (E_Y) E_g$ by (AcCofib 4) for g .

We have $E_Z \setminus (E_X) E_{fg} = E_Z \setminus (E_X) E_f E_g \stackrel{\text{Rem. 30}}{=} (E_Z \setminus (E_Y) E_g) \dot{\cup} (E_Y \setminus (E_X) E_f) E_g$, since $E_{fg} = E_f E_g$.

Suppose given $e_Z \in E_Z \setminus (E_X) E_{fg}$. We have to show that $(e_Z) t_Z \in V_Z \setminus (V_X) V_{fg}$.

We consider two cases.

Case 1: $e_Z \in E_Z \setminus (E_Y) E_g$. We have to show that $(e_Z) t_Z \in V_Z \setminus (V_X) V_{fg}$. We obtain:

We have $(e_Z) t_Z \in V_Z \setminus (V_Y) V_g \subseteq V_Z \setminus (V_X) V_f V_g = V_Z \setminus (V_X) V_{fg}$.

Case 2: $e_Z \in (E_Y \setminus (E_X) E_f) E_g$. We have to show that $(e_Z) t_Z \in V_Z \setminus (V_X) V_{fg}$. There exists $e_Y \in E_Y \setminus (E_X) E_f$ such that $(e_Y) E_g = e_Z$. We have $(e_Y) t_Y \in V_Y \setminus (V_X) V_f$. So we get

$$(e_Z) t_Z = (e_Y) E_g t_Z = (e_Y) t_Y V_g \in (V_Y \setminus (V_X) V_f) V_g \stackrel{\text{Rem. 30}}{\subseteq} V_Z \setminus (V_X) V_{fg}.$$

So (AcCofib 4) holds for fg .

Ad (AcCofib 5).

Suppose given $v_Z \in V_Z \setminus (V_X) V_{fg}$. We have so show that there exist $n \geq 1$ and $e_i \in E_Z$ for $i \in [1, n]$ such that $e_1 s_Z \in (V_X) V_{fg}$, such that $e_i t_Z = e_{i+1} s_Z$ for $i \in [1, n-1]$ and such that $e_n t_Z = v_Z$.

We have $(V_X) V_{fg} \subseteq (V_Y) V_g \subseteq V_Z$.

We have $V_Z \setminus (V_X) V_{fg} = (V_Z \setminus (V_Y) V_g) \dot{\cup} ((V_Y) V_g \setminus (V_X) V_{fg}) = (V_Z \setminus (V_Y) V_g) \dot{\cup} (V_Y \setminus (V_X) V_f) V_g$; cf. Remark 30.

Case $v_Z \in (V_Y \setminus (V_X) V_f) V_g$.

There exists a unique vertex $v_Y \in V_Y \setminus (V_X) V_f$ such that $v_Y V_g = v_Z$.

The graph morphism $f : X \rightarrow Y$ is an acyclic cofibration. So we may choose $n \geq 1$ and $\tilde{e}_i \in E_Y$ for $i \in [1, n]$ such that $\tilde{e}_1 s_Y \in (V_X) V_f$, such that $\tilde{e}_i t_Y = \tilde{e}_{i+1} s_Y$ for $i \in [1, n-1]$ and such that $\tilde{e}_n t_Y = v_Y$.

So $(\tilde{e}_1 E_g) s_Z = (\tilde{e}_1 s_Y) V_g \in (V_X) V_{fg}$. Moreover, $(\tilde{e}_i E_g) t_Z = (\tilde{e}_i t_Y) V_g = (\tilde{e}_{i+1} s_Y) V_g = (\tilde{e}_{i+1} E_g) s_Z$ for $i \in [1, n-1]$. Finally, $(\tilde{e}_n E_g) t_Z = (\tilde{e}_n t_Y) V_g = v_Y V_g = v_Z$.

So we may take $e_i := \tilde{e}_i E_g$ for $i \in [1, n]$.

Case $v_Z \in V_Z \setminus (V_Y) V_g$.

The graph morphism $g : X \rightarrow Y$ is an acyclic cofibration. So we may choose $k \geq 1$ and $e'_i \in E_Z$ for $i \in [1, k]$ such that $e'_1 s_Z \in (V_Y) V_g$, such that $e'_i t_Z = e'_{i+1} s_Z$ for $i \in [1, k-1]$ and such that $e'_k t_Z = v_Z$.

There exists a unique vertex $v_Y \in V_Y$ such that $v_Y V_g = e'_1 s_Z$.

Subcase 1: $v_Y \in (V_X) V_f$.

We map $n := k$, and $e_i := e'_i$ for $i \in [1, n]$.

Then, in particular, $e_1 s_Z = e'_1 s_Z = v_Y V_g \in (V_X) V_{fg}$.

Subcase 2: $v_Y \in V_Y \setminus (V_X) V_{fg}$.

Because $f : X \rightarrow Y$ is an acyclic cofibration and because of $v_Y \in V_Y \setminus (V_X) V_f$, we may choose $m \geq 1$ and $e''_i \in E_Y$ for $i \in [1, m]$ such that $e''_1 s_Y \in (V_X) V_f$, such that $e''_i t_Y = e''_{i+1} s_Y$ for $i \in [1, m-1]$ and such that $e''_m t_Y = v_Y$.

Let $n := m + k$. For $i \in [1, n]$ let $e_i := \begin{cases} e''_i E_g & \text{if } i \in [1, m] \\ e'_{i-m} & \text{if } i \in [m+1, m+k]. \end{cases}$

Then $e_1 s_Z = e''_1 E_g s_Z = e''_1 s_Y V_g \in (V_X) V_f V_g = (V_X) V_{fg}$.

For $i \in [1, m-1]$ we have $e_i t_Z = e''_i E_g t_Z = e''_i t_Y V_g = e''_{i+1} s_Y V_g = e''_{i+1} E_g s_Z = e_{i+1} s_Z$.

For $i = m$, we have $e_m t_Z = e''_m E_g t_Z = e''_m t_Y V_g = v_Y V_g = e'_1 s_Z = e_{m+1} s_Z$.

For $i \in [m+1, n-1]$ we have $e_i t_Z = e'_{i-m} t_Z = e'_{i+1-m} s_Z = e_{i+1} s_Z$.

Finally, we have $e_n t_Z = e_{m+k} t_Z = e'_k t_Z = v_Z$.

So we may choose $m \geq 1$ and $\tilde{e}_i E_g \in E_Y E_g \subseteq E_Z$ for $i \in [1, m]$ such that $(\tilde{e}_1 E_g) s_Z = (\tilde{e}_1 s_Y) V_g \in (V_X) V_{fg}$, such that $(\tilde{e}_i E_g) t_Z = (\tilde{e}_i t_Y) V_g = (\tilde{e}_{i+1} s_Y) V_g = (\tilde{e}_{i+1} E_g) s_Z$ for $i \in [1, m-1]$ and such that $(\tilde{e}_m E_g) t_Z = (\tilde{e}_m t_Y) V_g = v_Y V_g = v_Z$.

So for $v_Z \in V_Z \setminus (V_Y) V_g$ we may choose $n := m + k \geq 1$ and $\tilde{e}_i E_g \in E_Y E_g \subseteq E_Z$ for $i \in [1, m]$ such that $(\tilde{e}_1 E_g) s_Z = (\tilde{e}_1 s_Y) V_g \in (V_X) V_{fg}$, such that $(\tilde{e}_i E_g) t_Z = (\tilde{e}_i t_Y) V_g = (\tilde{e}_{i+1} s_Y) V_g = (\tilde{e}_{i+1} E_g) s_Z$ for $i \in [1, m-1]$ and such that $(\tilde{e}_m E_g) t_Z = (\tilde{e}_m t_Y) V_g = v_Y V_g = v_Z$.

And we may choose $e_i \in E_Z$ for $i \in [m+1, m+k]$ such that $e_{m+1} s_Z \in V_Y V_g$, such that $e_i t_Z = e_{i+1} s_Z$ for $i \in [m+1, m+k-1]$ and such that $e_{m+k} t_Z = v_Z$.

So we may choose $n := k + m \geq 1$ and $e_i \in E_Z$ for $i \in [1, n]$ such that $e_1 s_Z \in (V_X) V_{fg}$, such that $e_i t_Z = e_{i+1} s_Z$ for $i \in [1, n-1]$ and such that $e_n t_Z = v_Z$.

So (AcCofib 5) holds for fg and so the graph morphism $fg : X \rightarrow Z$ is an acyclic cofibration. \square

Once Lemma 191 below is known, which states that we have $\text{AcCofib} = \square \text{Fib}$, we can also argue by Remark 19 to obtain the statement of Remark 169.

Remark 170 Suppose given an acyclic cofibration $f : X \twoheadrightarrow Y$.

Then the image $Xf \subseteq Y$ is a full subgraph of Y ; cf. Definition 47.

Proof. We have to show that we have $E_{Xf} = \{e_Y \in E_Y : e_Y s_Y \in V_{Xf} \text{ and } e_Y t_Y \in V_{Xf}\}$, i.e. we have to show that we have $e \in E_{Xf}$ for $e \in E_Y$, $v, w \in V_{Xf}$ with $e s_Y = v$, $e t_Y = w$.

Suppose given $e \in E_Y$, $v, w \in V_{Xf}$ with $e s_Y = v$, $e t_Y = w$.

We have to show that $e \in E_{Xf} \subseteq E_Y$.

We have $(e) t_Y \in V_Y \setminus (V_X) V_f$ for $e \in E_Y \setminus (E_X) E_f$; cf. Definition 162 (AcCofib 4).

So we have $e \in (E_X) E_f$ for $(e) t_Y \in (V_X) V_f$.

We have $e t_Y = w \in V_{Xf} = V_X V_f$.

So we have $e \in (E_X) E_f = E_{Xf}$. □

Remark 171 In Gph , a pushout of an acyclic cofibration is an acyclic cofibration; cf. Definition 162.

Proof. Suppose given a pushout in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \circlearrowleft \downarrow & & \downarrow g \\ X' & \xrightarrow{a'} & Y' \end{array}$$

We use the alternative construction for the pushout in Gph from Remark 84; cf. Remark 38.

We may use this construction of the pushout to prove that g is an acyclic cofibration, since pushouts are unique up to isomorphism; cf. Remarks 7, 165 and 169.

We have the pushouts

$$\begin{array}{ccc} V_X & \xrightarrow{V_a} & V_Y \\ V_f \downarrow & & \downarrow V_g \\ V_{X'} & \xrightarrow{V_{a'}} & V_{Y'} \end{array}$$

and

$$\begin{array}{ccc} E_X & \xrightarrow{E_a} & E_Y \\ E_f \downarrow & & \downarrow E_g \\ E_{X'} & \xrightarrow{E_{a'}} & E_{Y'} \end{array}$$

in Gph ; cf. Remarks 38 and 84.

Ad (AcCofib 1, 2).

Since the graph morphism $f : X \rightarrow X'$ is an acyclic cofibration, the maps $V_f : V_X \rightarrow V_{X'}$ and $E_f : E_X \rightarrow E_{X'}$ are injective; cf. (AcCofib 1, 2).

So the maps $V_g : V_Y \rightarrow V_{Y'}$ and $E_g : E_Y \rightarrow E_{Y'}$ are injective; cf. Remark 39.

So (AcCofib 1) and (AcCofib 2) hold for the graph morphism $g : Y \rightarrow Y'$.

Ad (AcCofib 3). Suppose given a vertex $v_{Y'} \in V_{Y'} \setminus V_{Y'g}$.

So $v_{Y'} = (1, v_{X'}) \in V_{Y'} = (V_{X'} \setminus V_{X'f}) \sqcup V_Y$ with $v_{X'} \in V_{X'} \setminus V_{X'f}$.

We have to show that $|\{e_{Y'} \in E_{Y'} : e_{Y'} t_{Y'} = v_{Y'}\}| \stackrel{!}{=} 1$.

First, we show that $|\{e_{Y'} \in E_{Y'} : e_{Y'} t_{Y'} = v_{Y'}\}| \stackrel{!}{\geq} 1$.

The graph morphism $f : X \rightarrow X'$ is an acyclic cofibration and we have $v_{X'} \in V_{X'} \setminus V_{X'f}$.

Because of (AcCofib 3) for f there exists a unique edge $e_{X'} \in E_{X'}$ such that $e_{X'} t_{X'} = v_{X'}$.

Assume that $e_{X'} \in E_{X'f}$. Then $e_{X'} = e_X E_f$ for some $e_X \in E_X$.

Hence $v_{X'} = e_{X'} t_{X'} = e_X E_f t_{X'} = (e_X t_X) V_f \in V_{X'f}$. *Contradiction*.

So $e_{X'} \in E_{X'} \setminus E_{X'f}$. Thus we have $(1, e_{X'}) \in E_{Y'} = (E_{X'} \setminus E_{X'f}) \sqcup E_Y$.

We obtain $(1, e_{X'}) t_{Y'} = e_{X'} t_{X'} V_{a'} = v_{X'} V_{a'} = (1, v_{X'}) = v_{Y'}$, because $v_{X'} \in V_{X'} \setminus V_{X'f}$.

Second, we show that $|\{e_{Y'} \in E_{Y'} : e_{Y'} t_{Y'} = v_{Y'}\}| \stackrel{!}{\leq} 1$.

Suppose given $e_{Y'}, \tilde{e}_{Y'} \in E_{Y'}$ such that $e_{Y'} t_{Y'} = v_{Y'} = \tilde{e}_{Y'} t_{Y'}$.

We have to show that $e_{Y'} \stackrel{!}{=} \tilde{e}_{Y'}$.

We *assume* that $e_{Y'} \in E_{Y'g}$. Then there exists an edge $e_Y \in E_Y$ such that $e_{Y'} = e_Y E_g$.

So we have $v_{Y'} = e_{Y'} t_{Y'} = e_Y E_g t_{Y'} = (e_Y t_Y) V_g \in V_{Y'g}$, which is a *contradiction*.

So we have $e_{Y'} \in E_{Y'} \setminus E_{Y'g}$. Thus there exists an edge $e_{X'} \in E_{X'} \setminus E_{X'f}$ such that $e_{Y'} = (1, e_{X'})$.

We *assume* that $\tilde{e}_{Y'} \in E_{Y'g}$. Then there exists an edge $\tilde{e}_Y \in E_Y$ such that $\tilde{e}_{Y'} = \tilde{e}_Y E_g$.

So we have $v_{Y'} = \tilde{e}_{Y'} t_{Y'} = \tilde{e}_Y E_g t_{Y'} = (\tilde{e}_Y t_Y) V_g \in V_{Y'g}$, which is a *contradiction*.

So we have $\tilde{e}_{Y'} \in E_{Y'} \setminus E_{Y'g}$. Thus there exists an edge $\tilde{e}_{X'} \in E_{X'} \setminus E_{X'f}$ such that $\tilde{e}_{Y'} = (1, \tilde{e}_{X'})$.

Now we have

$$\begin{aligned} v_{Y'} &= e_{Y'} t_{Y'} = (1, e_{X'}) t_{Y'} = e_{X'} t_{X'} V_{a'} \\ &= \begin{cases} (2, v_X V_a) & \text{if } e_{X'} t_{X'} = v_X V_f \in V_{X'f} \text{ for a unique } v_X \in V_X \\ (1, e_{X'} t_{X'}) & \text{if } e_{X'} t_{X'} \in V_{X'} \setminus V_{X'f} . \end{cases} \end{aligned}$$

We *assume* that there exists a vertex $v_X \in V_X$ such that $e_{X'} t_{X'} = v_X V_f$.

Then we have $v_{Y'} = (2, v_X V_a) \in V_{Y'g}$. But we have $v_{Y'} \in V_{Y'} \setminus V_{Y'g}$, which is a *contradiction*.

So we have $v_{Y'} = (1, e_{X'} t_{X'})$.

Now we have

$$\begin{aligned} v_{Y'} &= \tilde{e}_{Y'} t_{Y'} = (1, \tilde{e}_{X'}) t_{Y'} = \tilde{e}_{X'} t_{X'} V_{a'} \\ &= \begin{cases} (2, v_X V_a) & \text{if } \tilde{e}_{X'} t_{X'} = v_X V_f \in V_{X'f} \text{ for a unique } v_X \in V_X \\ (1, \tilde{e}_{X'} t_{X'}) & \text{if } \tilde{e}_{X'} t_{X'} \in V_{X'} \setminus V_{X'f} . \end{cases} \end{aligned}$$

We assume that there exists a vertex $v_X \in V_X$ such that $\tilde{e}_{X'} t_{X'} = v_X V_f$.

Then we have $v_{Y'} = (2, v_X V_a) \in V_{Y_g}$. But we have $v_{Y'} \in V_Y \setminus V_{Y_g}$, which is a *contradiction*.

So we have $v_{Y'} = (1, \tilde{e}_{X'} t_{X'})$.

Recall that $v_{X'} \in V_{X'} \setminus V_{X_f}$ and $v_{Y'} = (1, v_{X'})$.

Therefore we have $e_{X'} t_{X'} = \tilde{e}_{X'} t_{X'} = v_{X'}$.

Because of (AcCofib 3) for f we have $e_{X'} = \tilde{e}_{X'}$.

So we have $e_{Y'} = (1, e_{X'}) = (1, \tilde{e}_{X'}) = \tilde{e}_{Y'}$.

So (AcCofib 3) holds for the graph morphism $g : Y \rightarrow Y'$.

Ad (AcCofib 4). Suppose given an edge $e_{Y'} \in E_{Y'} \setminus E_{Y_g}$.

We have to show that $e_{Y'} t_{Y'} \notin V_{Y'} \setminus V_{Y_g}$.

We have $e_{Y'} \notin E_{Y_g}$. So there exists an edge $e_{X'} \in E_{X'} \setminus E_{X_f}$ such that $e_{Y'} = (1, e_{X'})$.

So we have

$$\begin{aligned} e_{Y'} t_{Y'} &= (1, e_{X'}) t_{Y'} = e_{X'} t_{X'} V_{a'} \\ &= \begin{cases} (2, v_X V_a) & \text{if } e_{X'} t_{X'} = v_X V_f \in V_{X_f} \text{ for a unique } v_X \in V_X \\ (1, e_{X'} t_{X'}) & \text{if } e_{X'} t_{X'} \in V_{X'} \setminus V_{X_f} . \end{cases} \end{aligned}$$

We have $e_{X'} \in E_{X'} \setminus E_{X_f}$. Because of (AcCofib 4) for f we conclude that $e_{X'} t_{X'} \notin V_{X_f}$.

So we have $e_{Y'} t_{Y'} = (1, e_{X'} t_{X'}) \notin V_{Y_g}$.

Hence (AcCofib 4) holds for the graph morphism $g : Y \rightarrow Y'$.

Ad (AcCofib 5). Suppose given $v_{Y'} \in V_{Y'} \setminus V_{Y_g}$, where $V_{Y_g} = \{(2, v_Y) : v_Y \in V_Y\}$.

We have to show that there exist $n \geq 1$ and $e_{Y',i} \in E_{Y'}$ for $i \in [1, n]$ such that $e_{Y',1} s_{Y'} \in V_{Y_g}$, such that $e_{Y',i} t_{Y'} = e_{Y',i+1} s_{Y'}$ for $i \in [1, n-1]$ and such that $e_{Y',n} t_{Y'} = v_{Y'}$.

Since $v_{Y'} \in V_{Y'} \setminus V_{Y_g}$, there exists a unique vertex $v_{X'} \in V_{X'} \setminus V_{X_f}$ such that $v_{Y'} = (1, v_{X'})$.

Now (AcCofib 5) holds for the graph morphism $f : X \rightarrow X'$.

So we may choose $n \geq 1$ and edges $e_{X',i} \in E_{X'}$ for $i \in [1, n]$ such that $e_{X',1} s_{X'} \in V_{X_f}$, such that $e_{X',i} t_{X'} = e_{X',i+1} s_{X'}$ for $i \in [1, n-1]$ and such that $e_{X',n} t_{X'} = v_{X'}$.

Without loss of generality, we have $e_{X',i} t_{X'} \notin V_{X_f}$ for $i \in [1, n]$. So we have $e_{X',i} \notin E_{X_f}$ for $i \in [1, n]$.

We let $e_{Y',i} := (1, e_{X',i})$ for $i \in [1, n]$.

We have $e_{Y',1} s_{Y'} = (1, e_{X',1}) s_{Y'} = e_{X',1} s_{X'} V_{a'}$.

Now there exists a unique vertex $v_X \in V_X$ such that $e_{X',1} s_{X'} = v_X V_f$.

So we have $e_{X',1} s_{X'} V_{a'} = (2, v_X V_a) = v_X V_a V_g \in V_{Y_g}$.

Suppose given $i \in [1, n-1]$. Then we have

$$e_{Y',i} t_{Y'} = (1, e_{X',i}) t_{Y'} = e_{X',i} t_{X'} V_{a'} = e_{X',i+1} s_{X'} V_{a'} = (1, e_{X',i+1}) s_{Y'} = e_{Y',i+1} s_{Y'} .$$

And we have

$$e_{Y',n} t_{Y'} = (1, e_{X',n}) t_{Y'} = e_{X',n} t_{X'} V_{a'} = v_{X'} V_{a'} = (1, v_{X'}) = v_{Y'} ,$$

because $v_{X'} \in V_{X'} \setminus V_{X'f}$.

So (AcCofib 5) holds for the graph morphism $g : Y \rightarrow Y'$.

So the graph morphism $g : Y \rightarrow Y'$ is an acyclic cofibration; cf. Definition 162.

$$\begin{array}{ccc} X & \xrightarrow{a} & Y \\ f \circlearrowleft \downarrow & & \downarrow \circlearrowright g \\ X' & \xrightarrow{a'} & Y' \end{array} \quad \lrcorner$$

□

Once Lemma 191 below is known, which states that we have $\text{AcCofib} = \square \text{Fib}$, we can also argue by Remark 21 to obtain the statement of Remark 171.

Remark 172 In Gph , the subset of acyclic cofibrations $\text{AcCofib} \subseteq \text{Mor}$ is closed under retracts; cf. Definition 23.

Proof. Suppose given the following commutative diagram in Gph .

$$\begin{array}{ccccc} & & \text{id}_{X'} & & \\ & & \curvearrowright & & \\ X' & \xrightarrow{a} & X & \xrightarrow{a'} & X' \\ \downarrow f' & & \downarrow f & & \downarrow f' \\ Y' & \xrightarrow{b} & Y & \xrightarrow{b'} & Y' \\ & & \curvearrowleft & & \\ & & \text{id}_{Y'} & & \end{array}$$

We have to show that the graph morphism $f' : X' \rightarrow Y'$ is an acyclic cofibration.

Ad (AcCofib 1).

We have to show that the map $V_{f'} : V_{X'} \rightarrow V_{Y'}$ is injective.

We have $V_{f'} \cdot V_b = V_a \cdot V_f$. So it suffices to show that the map $V_a \cdot V_f$ is injective.

The map V_f is injective since $f \in \text{AcCofib}$.

The map V_a is injective since $V_a \cdot V_{a'} = V_{\text{id}_{X'}}$.

Ad (AcCofib 2).

We have to show that the map $E_{f'} : E_{X'} \rightarrow E_{Y'}$ is injective.

We have $E_{f'} \cdot E_b = E_a \cdot E_f$. So it suffices to show that the map $E_a \cdot E_f$ is injective.

The map E_f is injective since $f \in \text{AcCofib}$.

The map E_a is injective since $E_a \cdot E_{a'} = E_{\text{id}_{X'}}$.

Ad (AcCofib 3).

Suppose given a vertex $v_{Y'} \in V_{Y'} \setminus V_{X'f'}$.

We have to show that we have $|\{e \in E_{Y'} : (e) t_{Y'} = v_{Y'}\}| \stackrel{!}{=} 1$.

We show that $v_{Y'} V_b \notin V_{Xf}$.

We assume that $v_{Y'} V_b \in V_{Xf}$. Then there exists a unique vertex $v_X \in V_X$ such that $v_{Y'} V_b = v_X V_f$.

Then we have $v_{Y'} = v_{Y'} V_b V_{b'} = v_X V_f V_{b'} = (v_X V_{a'}) V_{f'} \in V_{X'f'}$, which is a *contradiction*.

First, we show that $|\{e \in E_{Y'} : e t_{Y'} = v_{Y'}\}| \stackrel{!}{\geq} 1$.

Because the graph morphism $f : X \rightarrow Y$ is an acyclic cofibration, there exists a unique edge $e_Y \in E_Y$ such that $e_Y t_Y = v_{Y'} V_b$. We have $(e_Y E_{b'}) t_{Y'} = e_Y t_Y V_{b'} = v_{Y'} V_b V_{b'} = v_{Y'}$.

Second, we show that $|\{e \in E_{Y'} : e t_{Y'} = v_{Y'}\}| \stackrel{!}{\leq} 1$.

Suppose given $e_{Y'}, \tilde{e}_{Y'} \in E_{Y'}$ such that $e_{Y'} t_{Y'} = \tilde{e}_{Y'} t_{Y'} = v_{Y'}$. We have to show that $e_{Y'} \stackrel{!}{=} \tilde{e}_{Y'}$.

We have $(e_{Y'} E_b) t_Y = e_{Y'} t_Y V_b = v_{Y'} V_b = \tilde{e}_{Y'} t_Y V_b = (\tilde{e}_{Y'} E_b) t_Y$.

Because the graph morphism $f : X \rightarrow Y$ is an acyclic cofibration and because we have $v_{Y'} V_b \in V_Y \setminus V_{Xf}$ we conclude that $e_{Y'} E_b = \tilde{e}_{Y'} E_b$.

So we have $e_{Y'} = e_{Y'} E_b E_{b'} = \tilde{e}_{Y'} E_b E_{b'} = \tilde{e}_{Y'}$.

Ad (AcCofib 4).

Suppose given an edge $e_{Y'} \in E_{Y'} \setminus E_{X'f'}$. We have to show that $e_{Y'} t_{Y'} \notin V_{X'f'}$.

We show that $e_{Y'} E_b \in E_Y \setminus E_{Xf}$.

We assume that $e_{Y'} E_b \in E_{Xf}$. Then there exists a unique edge $e_X \in E_X$ such that $e_{Y'} E_b = e_X E_f$.

Then we have $e_{Y'} = e_{Y'} E_b E_{b'} = e_X E_f E_{b'} = (e_X E_{a'}) E_{f'} \in E_{X'f'}$, which is a *contradiction*.

We assume that $e_{Y'} t_{Y'} \in V_{X'f'}$. Then there exists a unique vertex $v_{X'} \in V_{X'}$ such that we have $e_{Y'} t_{Y'} = v_{X'} V_{f'}$. So we have $(e_{Y'} E_b) t_Y = e_{Y'} t_Y V_b = v_{X'} V_{f'} V_b = (v_{X'} V_a) V_f \in V_{Xf}$.

But we have $e_{Y'} E_b \in E_Y \setminus E_{Xf}$. Since the graph morphism $f : X \rightarrow Y$ is an acyclic cofibration, we conclude that $e_{Y'} E_b t_Y \in V_Y \setminus V_{Xf}$ which is a *contradiction*.

Ad (AcCofib 5).

Suppose given a vertex $v_{Y'} \in V_{Y'} \setminus V_{X'f'}$.

We have to show that there exist $n \geq 1$ and $e_i \in E_{Y'}$ for $i \in [1, n]$ such that $e_1 s_{Y'} \in V_{X'f'}$, such that $e_i t_{Y'} = e_{i+1} s_{Y'}$ for $i \in [1, n-1]$ and such that $e_n t_{Y'} = v_{Y'}$.

We show that $v_{Y'} V_b \notin V_{Xf}$.

We assume that $v_{Y'} V_b \in V_{Xf}$. Then there exists a unique vertex $v_X \in V_X$ such that $v_{Y'} V_b = v_X V_f$.

Then we have $v_{Y'} = v_{Y'} V_b V_{b'} = v_X V_f V_{b'} = (v_X V_{a'}) V_{f'} \in V_{X'f'}$, which is a *contradiction*.

Now (AcCofib 5) holds for the acyclic cofibration $f : X \rightarrow Y$. So we may choose $n \geq 1$ and edges $e_{Y,i} \in E_Y$ for $i \in [1, n]$ such that $e_{Y,1} s_Y \in V_{Xf}$ and $e_{Y,i} t_Y = e_{Y,i+1} s_Y$ for $i \in [1, n-1]$ and

such that $e_{Y,n} t_Y = v_{Y'} V_b$.

We let $e_i := e_{Y,i} E_{b'}$ for $i \in [1, n]$.

We will show that the following statements (i, ii, iii) hold.

$$(i) \quad e_1 s_{Y'} \stackrel{!}{\in} V_{X'f'}$$

$$(ii) \quad e_i t_{Y'} \stackrel{!}{=} e_{i+1} s_{Y'} \text{ for } i \in [1, n-1].$$

$$(iii) \quad e_n t_{Y'} \stackrel{!}{=} v_{Y'}$$

Ad (i). There exists a unique vertex $v_X \in V_X$ such that $e_{Y,1} s_Y = v_X V_f$.

Then we have $e_1 s_{Y'} = (e_{Y,1} E_{b'}) s_{Y'} = e_{Y,1} s_Y V_{b'} = v_X V_f V_{b'} = (v_X V_a) V_{f'} \in V_{X'f'}$.

Ad (ii). Suppose given $i \in [1, n-1]$.

We have $e_{Y,i} t_Y = e_{Y,i+1} s_Y$. So we have $e_i t_{Y'} = (e_{Y,i} E_{b'}) t_{Y'} = e_{Y,i} t_Y V_{b'} = e_{Y,i+1} s_Y V_{b'} = (e_{Y,i+1} E_{b'}) s_{Y'} = e_{i+1} s_{Y'}$.

Ad (iii). We have $e_{Y,n} t_Y = v_{Y'} V_b$.

So we have $e_n t_{Y'} = (e_{Y,n} E_{b'}) t_{Y'} = e_{Y,n} t_Y V_{b'} = v_{Y'} V_b V_{b'} = v_{Y'}$.

Altogether, the graph morphism $f' : X' \rightarrow Y'$ is an acyclic cofibration, i.e. f' is in AcCofib . \square

Once Lemma 191 below is known, which states that we have $\text{AcCofib} = \square \text{Fib}$, we can also argue by Remark 25 to obtain the statement of Remark 172.

However, we need Remark 172 to prove Lemma 185, which we need to prove Lemma 191.

Lemma 173 Suppose given a graph Y .

Suppose given a full subgraph $X \subseteq Y$ such that the following properties (1, 2) hold.

(1) We have $e_Y s_Y \in V_X$ for $e_Y \in E_Y \setminus E_X$.

(2) For $v_Y \in V_Y \setminus V_X$, there exists a unique $e_Y \in E_Y$ such that $e_Y t_Y = v_Y$.

Then the inclusion morphism $\iota := \text{id}_Y|_X : X \rightarrow Y$ is in $\square \text{Fib}$.

Proof. Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ \downarrow \iota & & \downarrow f' \\ Y & \xrightarrow{b} & Y' \end{array}$$

We have to show that there exists a graph morphism $h : Y \rightarrow X'$ such that $\iota h = a$ and $h f' = b$.

Let $v_X V_h := v_X V_a$ for $v_X \in V_X$ and $e_X E_h := e_X E_a$ for $e_X \in E_X$.

Now suppose given an edge $e_Y \in E_Y \setminus E_X$.

We now consider the edge $e_Y E_b \in E_{Y'}$.

We abbreviate $v_{X'} := e_Y s_Y V_a \in V_{X'}$; cf. property (1).

Since $ib = af'$, we have $e_Y E_b s_{Y'} = e_Y s_Y V_b \stackrel{e_Y s_Y \in V_X}{=} e_Y s_Y V_l V_b = e_Y s_Y V_a V_{f'} = v_{X'} V_{f'}$.

Since the graph morphism $f' : X' \rightarrow Y'$ is a fibration, the map

$$E_{f', v_{X'}} = E_{f'} \Big|_{X'(v_{X'}, *)}^{Y'(v_{X'} V_{f'}, *)} : X'(v_{X'}, *) \rightarrow Y'(v_{X'} V_{f'}, *)$$

is surjective; cf. Definition 127.(1).

We have $v_{X'} V_{f'} = e_Y E_b s_{Y'}$, and therefore $e_Y E_b \in Y'(v_{X'} V_{f'}, *)$.

So we may choose an edge $(e_Y)_{X'} \in X'(v_{X'}, *)$ such that $(e_Y)_{X'} E_{f'} = e_Y E_b$.

So for the edge $e_Y \in E_Y \setminus E_X$, we let $e_Y E_h := (e_Y)_{X'} \in E_{X'}$.

Finally, suppose given a vertex $v_Y \in V_Y \setminus V_X$.

By (2), there exists a unique edge $e_{v_Y} \in E_Y$ such that $e_{v_Y} t_Y = v_Y$.

Note that $e_{v_Y} \in E_Y \setminus E_X$ since $e_{v_Y} t_Y = v_Y \in V_Y \setminus V_X$.

We let $v_Y V_h := (e_{v_Y})_{X'} t_{X'} \in V_{X'}$.

Before we show that $h : Y \rightarrow X'$ is a graph morphism we make a remark.

Suppose given an edge $e_Y \in E_Y \setminus E_X$.

We *assume* that $e_Y t_Y \in V_X$.

We have $e_Y s_Y \in V_X$; cf. property (1).

Since $X \subseteq Y$ is a full subgraph, we have $e_Y \in E_X$. *Contradiction.*

So for the edge $e_Y \in E_Y \setminus E_X$ we have $e_Y t_Y \in V_Y \setminus V_X$.

In particular, we have $e_Y = e_{e_Y t_Y}$. This finishes the remark.

We have to show that $h : Y \rightarrow X'$ is a graph morphism.

First, we have to show that $E_h s_{X'} \stackrel{!}{=} s_Y V_h$.

Suppose given an edge $e_Y \in E_Y$. We have to show that $e_Y E_h s_{X'} \stackrel{!}{=} e_Y s_Y V_h$.

Case $e_Y \in E_X$.

We have $e_Y E_h s_{X'} = e_Y E_a s_{X'} = e_Y s_X V_a \stackrel{e_Y s_X \in V_X}{=} e_Y s_X V_h = e_Y s_Y V_h$.

Case $e_Y \in E_Y \setminus E_X$.

We have $e_Y E_h s_{X'} = (e_Y)_{X'} s_{X'} \stackrel{(e_Y)_{X'} \in X'(v_{X'}, *)}{=} v_{X'} \stackrel{\text{Def. } v_{X'}}{=} e_Y s_Y V_a \stackrel{(1)}{=} e_Y s_Y V_h$.

Second, we have to show that $E_h t_{X'} \stackrel{!}{=} t_Y V_h$.

Suppose given an edge $e_Y \in E_Y$. We have to show that $e_Y E_h t_{X'} \stackrel{!}{=} e_Y t_Y V_h$.

Case $e_Y \in E_X$.

We have $e_Y E_h t_{X'} = e_Y E_a t_{X'} = e_Y t_X V_a \stackrel{e_Y t_X \in V_X}{=} e_Y t_X V_h = e_Y t_Y V_h$.

Case $e_Y \in E_Y \setminus E_X$.

Now we need the remark made above.

Writing $v_Y := e_Y t_Y \in V_Y \setminus V_X$, we obtain

$$e_Y t_Y V_h = v_Y V_h = (e_{v_Y})_{X'} t_{X'} = e_{v_Y} E_h t_{X'} = e_Y E_h t_{X'} .$$

So $h : Y \rightarrow X'$ is in fact a graph morphism.

We now show that $a \stackrel{!}{=} \iota h$.

Therefor we have to show that $V_a \stackrel{!}{=} V_\iota V_h$ and $E_a \stackrel{!}{=} E_\iota E_h$.

Suppose given a vertex $v_X \in V_X$. We have $v_X V_\iota V_h = v_X V_h = v_X V_a$.

Suppose given an edge $e_X \in E_X$. We have $e_X E_\iota E_h = e_X E_h = e_X E_a$.

We now show that $h f' \stackrel{!}{=} b$.

First, we have to show that $V_h V_{f'} \stackrel{!}{=} V_b$.

Suppose given a vertex $v_Y \in V_Y$. We have to show that $v_Y V_h V_{f'} \stackrel{!}{=} v_Y V_b$.

Case $v_Y \in V_X$.

We have $v_Y V_h V_{f'} = v_Y V_a V_{f'} = v_Y V_\iota V_b = v_Y V_b$.

Case $v_Y \in V_Y \setminus V_X$.

We have $v_Y V_h V_{f'} = (e_{v_Y})_{X'} t_{X'} V_{f'} = (e_{v_Y})_{X'} E_{f'} t_{Y'} \stackrel{\text{choice of } (e_{v_Y})_{X'}}{=} e_{v_Y} E_b t_{Y'} = e_{v_Y} t_Y V_b = v_Y V_b$.

Second, we have to show that $E_h E_{f'} \stackrel{!}{=} E_b$.

Suppose given $e_Y \in E_Y$. We have to show that $e_Y E_h E_{f'} \stackrel{!}{=} e_Y E_b$.

Case $e_Y \in E_X$.

We have $e_Y E_h E_{f'} = e_Y E_a E_{f'} = e_Y E_\iota E_b = e_Y E_b$.

Case $e_Y \in E_Y \setminus E_X$.

We have $e_Y E_h E_{f'} = (e_Y)_{X'} E_{f'} \stackrel{\text{choice of } (e_Y)_{X'}}{=} e_Y E_b$.

$$\begin{array}{ccc} X & \xrightarrow{a} & X' \\ \downarrow \iota & \nearrow h & \downarrow f' \\ Y & \xrightarrow{b} & Y' \end{array}$$

□

Lemma 174 We have $\text{AcCofib} \subseteq \square \text{Fib}$.

Proof. Suppose given an acyclic cofibration $f : X \twoheadrightarrow Y$.

We have to show that f satisfies $(\text{LLP}_{\text{Fib}})$; cf. Definition 13.

Suppose given a fibration $g : X' \twoheadrightarrow Y'$.

Suppose given graph morphisms $u : X \rightarrow X'$ and $r : Y \rightarrow Y'$ such that $ug = fr$.

$$\begin{array}{ccc} X & \xrightarrow{u} & X' \\ \downarrow f \circ \circlearrowleft & & \downarrow g \\ Y & \xrightarrow{r} & Y' \end{array}$$

We have to show that there exists a graph morphism $h : Y \rightarrow X'$ such that $fh = u$ and $hg = r$. Since (AcCofib 1, 2) hold for the acyclic cofibration $f : X \dashrightarrow Y$, the graph morphism $f|^{Xf} : X \rightarrow Xf \subseteq Y$ is bijective, i.e. a graph isomorphism; cf. Remark 66. Thus $f|^{Xf} \in \text{Iso} \subseteq \square \text{Fib}$; cf. Remark 18.

Moreover, the graph Xf is a full subgraph in Y ; cf. Remark 170.

We shall define full subgraphs $Y_n \subseteq Y$ recursively.

Let $Y_0 := Xf$.

For $n \in \mathbb{Z}_{\geq 1}$, let Y_n be the full subgraph of Y with

$$V_{Y_n} := V_{Y_{n-1}} \cup \{v \in V_Y : \exists e \in Y(V_{Y_{n-1}}, v)\};$$

cf. Definition 47.(3) and Notation 51.(2).

We now have a countable chain of subgraphs

$$Xf = Y_0 \subseteq Y_1 \subseteq Y_2 \subseteq \dots \subseteq Y.$$

Note that we have the full subgraph $Y_k \subseteq Y_{k+1}$ for $k \in \mathbb{Z}_{\geq 0}$.

Suppose given a vertex $v_Y \in V_Y \setminus V_{Xf}$.

Since (AcCofib 5) holds for the acyclic cofibration $f : X \dashrightarrow Y$, we may choose $n \geq 1$ and edges $e_i \in E_Y$ for $i \in [1, n]$ such that $e_1 s_Y \in V_{Xf}$, such that $e_i t_Y = e_{i+1} s_Y$ for $i \in [1, n-1]$ and such that $e_n t_Y = v_Y$.

We have $e_1 s_Y \in V_{Xf} = V_{Y_0}$.

So we have $e_1 t_Y \in V_{Y_1}$.

Since $e_2 s_Y = e_1 t_Y \in V_{Y_1}$, we have $e_2 t_Y \in V_{Y_2}$.

Since $e_3 s_Y = e_2 t_Y \in V_{Y_2}$, we have $e_3 t_Y \in V_{Y_3}$.

Etc.

So we deduce that $v_Y = e_n t_Y \in V_{Y_n} \subseteq \bigcup_{k \in \mathbb{Z}_{\geq 0}} V_{Y_k}$.

So we deduce that $\bigcup_{k \in \mathbb{Z}_{\geq 0}} V_{Y_k} = V_Y$.

Since the graph $\bigcup_{k \in \mathbb{Z}_{\geq 0}} Y_k$ is a full subgraph of Y , we have $\bigcup_{k \in \mathbb{Z}_{\geq 0}} Y_k = Y$; cf. Remarks 104 and 50.

Suppose given $k \in \mathbb{Z}_{\geq 0}$. The graph Y_k is a full subgraph of Y_{k+1} . In order to apply Lemma 173, we will show that the following properties (1, 2) hold.

(1) We have $e_{Y_{k+1}} s_{Y_{k+1}} \in V_{Y_k}$ for $e_{Y_{k+1}} \in E_{Y_{k+1}} \setminus E_{Y_k}$.

(2) For $v_{Y_{k+1}} \in V_{Y_{k+1}} \setminus V_{Y_k}$, there exists a unique $e_{Y_{k+1}} \in E_{Y_{k+1}}$ such that $e_{Y_{k+1}} t_{Y_{k+1}} = v_{Y_{k+1}}$.

Ad (1). Suppose given an edge $e \in E_{Y_{k+1}} \setminus E_{Y_k}$. We have to show that $e s_Y \in V_{Y_k}$.

We have $e t_Y \in V_{Y_{k+1}}$.

There exists a unique minimal $l \in [0, k + 1]$ such that $e t_Y \in V_{Y_l}$.

We have $e \in E_{Y_{k+1}} \setminus E_{Y_k} \subseteq E_Y \setminus E_{Xf}$.

Since (AcCofib 4) holds for the acyclic cofibration $f : X \dashrightarrow Y$, we have $e t_Y \in V_Y \setminus V_{Xf} = V_Y \setminus V_{Y_0}$.

So we deduce that $l \geq 1$.

By minimality of l , we have $e t_Y \in V_{Y_l} \setminus V_{Y_{l-1}}$.

By construction of V_{Y_l} , we may choose an edge $\tilde{e} \in E_Y$ such that $\tilde{e} s_Y \in V_{Y_{l-1}}$ and $\tilde{e} t_Y = e t_Y$.

Note that we have $\tilde{e} t_Y = e t_Y \in V_{Y_l} \setminus V_{Y_{l-1}} \subseteq V_Y \setminus V_{Xf}$.

Since (AcCofib 3) holds for the acyclic cofibration $f : X \dashrightarrow Y$, we infer $e = \tilde{e}$.

So we have $e s_Y = \tilde{e} s_Y \in V_{Y_{l-1}} \subseteq V_{Y_k}$.

Ad (2). Suppose given a vertex $v \in V_{Y_{k+1}} \setminus V_{Y_k}$. We have to show that there exists a unique edge $e \in E_{Y_{k+1}}$ such that $e t_Y \stackrel{!}{=} v$.

By definition of $V_{Y_{k+1}}$, we may choose an edge $e \in E_Y$ such that $e s_Y \in V_{Y_k}$ and $e t_Y = v$.

Since $Y_{k+1} \subseteq Y$ is a full subgraph, we have $e \in E_{Y_{k+1}}$.

Uniqueness follows since $v \in V_{Y_{k+1}} \setminus V_{Y_k} \subseteq V_Y \setminus V_{Xf}$ by (AcCofib 3) for the acyclic cofibration $f : X \dashrightarrow Y$.

So the inclusion morphism $\text{id}_{Y_{k+1}}|_{Y_k} : Y_k \rightarrow Y_{k+1}$ is in $\square\text{Fib}$; cf. Lemma 173.

Since the inclusion morphisms $\text{id}_{Y_{k+1}}|_{Y_k} : Y_k \rightarrow Y_{k+1}$ are in $\square\text{Fib}$ for $k \geq 0$ and since $\bigcup_{k \in \mathbb{Z}_{\geq 0}} Y_k = Y$, the graph morphism $\text{id}_Y|_{Y_0} : Y_0 \rightarrow Y$ is in $\square\text{Fib}$; cf. Lemma 106.

Altogether, $f = f|^{Xf} \cdot \text{id}_Y|_{Y_0}$ is in $\square\text{Fib}$; cf. Remark 19. \square

The following lemma is due to BISSON and TSEMO [3, Prop. 4.5].

Lemma 175 In Gph, we have $\text{AcCofib} \subseteq \text{Cofib} \cap \text{Qis}$; cf. Definitions 115, 144 and 162.

Proof. We claim that $\text{AcCofib} \stackrel{!}{\subseteq} \text{Qis}$.

Suppose given an acyclic cofibration $f : X \dashrightarrow Y$. We have to show that f is a quasiisomorphism.

Suppose given $n \geq 1$ and a graph morphism $u : C_n \rightarrow Y$. We have to show that $C_n u \stackrel{!}{\subseteq} Xf$; cf. Remark 66.

$$\begin{array}{ccc} & X & \\ & \downarrow f|^{Xf} & \\ & Xf & \\ & \downarrow \text{id}_Y|_{Xf} & \\ C_n & \xrightarrow{u} & Y \end{array}$$

We assume that $C_n u \not\subseteq Xf$. So $V_{C_n} V_u \not\subseteq V_{Xf}$ or $E_{C_n} E_u \not\subseteq E_{Xf}$.

By (AcCofib 4), it is impossible to have $V_{C_n} V_u \subseteq V_{Xf}$ and $E_{C_n} E_u \not\subseteq E_{Xf}$.

So $V_{C_n} V_u \not\subseteq V_{Xf}$.

Hence, we may choose $k \in \mathbb{Z}/n\mathbb{Z}$ with $v_k V_u \in V_Y \setminus V_{Xf}$.

Because of (AcCofib 5) we may choose $m \geq 1$ and $e_i \in E_Y$ for $i \in [1, m]$ such that $e_1 s_Y \in V_{Xf}$, such that $e_i t_Y = e_{i+1} s_Y$ for $i \in [1, m-1]$ and such that $e_m t_Y = v_k V_u$.

Without loss of generality, we have $e_i t_Y \in V_Y \setminus V_{Xf}$ for $i \in [1, m]$.

And we have $e_{k-1} E_u t_Y = e_{k-1} t_{C_n} V_u = v_k V_u = e_m t_Y$.

Because of (AcCofib 3) we have $e_m = e_{k-1} E_u$.

If $m \geq 2$, then we have $e_{m-1} t_Y = e_m s_Y = e_{k-1} E_u s_Y = e_{k-1} s_{C_n} V_u = v_{k-1} V_u = e_{k-2} t_{C_n} V_u = e_{k-2} E_u t_Y$, in $V_Y \setminus V_{Xf}$.

Because of (AcCofib 3), we obtain $e_{m-1} = e_{k-2} E_u$.

If $m \geq 3$, then, by the same argument, we obtain $e_{m-2} = e_{k-3} E_u$.

If $m \geq 4$, then, by the same argument, we obtain $e_{m-3} = e_{k-4} E_u$.

Etc.

Continuing this procedure, we obtain $e_1 = e_{k-m} E_u$.

So we have $V_{Xf} \ni e_1 s_Y = e_{k-m} E_u s_Y = e_{k-m} s_{C_n} V_u = v_{k-m} V_u$.

We have $e_k E_u s_Y = e_k s_{C_n} V_u = v_k V_u \in V_Y \setminus V_{Xf}$.

So we have $e_k E_u \in E_Y \setminus E_{Xf}$. Because of (AcCofib 4) we have $e_{k+1} E_u s_Y = e_{k+1} s_{C_n} V_u = v_{k+1} V_u = e_k t_{C_n} V_u = e_k E_u t_Y \in V_Y \setminus V_{Xf}$.

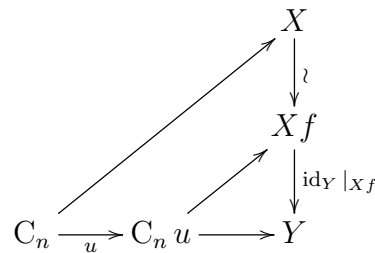
So $e_{k+1} E_u \in E_Y \setminus E_{Xf}$. By the same argument, we conclude that $e_{k+2} E_u s_Y = v_{k+2} V_u = e_{k+1} E_u t_Y \in V_Y \setminus V_{Xf}$.

So $e_{k+2} E_u \in E_Y \setminus E_{Xf}$. By the same argument, we conclude that $e_{k+3} E_u s_Y = v_{k+3} V_u = e_{k+2} E_u t_Y \in V_Y \setminus V_{Xf}$.

Etc.

Continuing this procedure, we obtain $v_{k-m} V_u = v_{k+l} V_u \in V_Y \setminus V_{Xf}$, where $l \geq 1$ is such that $k+l = k-m \in \mathbb{Z}/n\mathbb{Z}$.

But $v_{k-m} V_u \in V_{Xf}$. This *contradiction* proves the *claim*.



Moreover, we have $\text{AcCofib} \stackrel{\text{Lemma 174}}{\subseteq} \square \text{Fib} \subseteq \square \text{AcFib} \stackrel{\text{Def. 144}}{=} \text{Cofib}$.

Altogether, we have $\text{AcCofib} \subseteq \text{Cofib} \cap \text{Qis}$. □

In Lemma 175 we have shown the inclusion $\text{AcCofib} \subseteq \text{Cofib} \cap \text{Qis}$.

In Lemma 185 below, we will show that the inclusion $\text{AcCofib} \stackrel{!}{\supseteq} \text{Cofib} \cap \text{Qis}$ also holds, using the factorization provided by Lemma 184.

Remark 176 In Gph , a pullback of an acyclic cofibration is a quasiisomorphism.

Proof. We have $\text{AcCofib} \subseteq \text{Qis}$; cf. Lemma 175.

So an acyclic cofibration is a quasiisomorphism.

A pullback of a quasiisomorphism is a quasiisomorphism; cf. Remark 125.

So a pullback of an acyclic cofibration is a quasiisomorphism. \square

Example 177 Suppose given $n \in \mathbb{N}$.

The graph morphism $\iota_{C_n} : \emptyset \rightarrow C_n$ is not an acyclic cofibration.

Proof. Since $V_{\emptyset \iota_{C_n}} = \emptyset$ and $V_{C_n} \neq \emptyset$, the graph morphism ι_{C_n} does not satisfy (AcCofib 5).

Alternatively, since $(C_n, \emptyset) = \emptyset \neq (C_n, C_n)$, the graph morphism ι_{C_n} is not a quasiisomorphism, and thus not an acyclic cofibration; cf. Lemma 175. \square

Remark 178 A graph G is a tree if and only if there exists an acyclic cofibration $c : D_0 \rightarrow G$.

Proof.

Suppose given an acyclic cofibration $c : D_0 \rightarrow G$.

We have to show that the graph G is a tree, i.e. that the properties (Tree 1–3) hold; cf. Definition 108.

We let $r := \hat{v}_0 V_c$. So we have $\{r\} = V_{D_0 c}$.

Ad (Tree 1). Because of (AcCofib 3) we have $|\{e \in E_G : e t_G = v_G\}| = 1$ for $v_G \in V_G \setminus V_{D_0 c} = V_G \setminus \{r\}$.

Ad (Tree 2). Because of (AcCofib 4) we have $e t_G \in V_G \setminus V_{D_0 c} = V_G \setminus \{r\}$ for $e \in E_G \setminus E_{D_0 c} = E_G \setminus \emptyset = E_G$. So we have $e t_G \neq r$ for $e \in E_G$ and thus (Tree 2).

Ad (Tree 3). Suppose given $v \in V_G$.

We have to show that there exists a path from r to v , i.e. $n \geq 0$ and a graph morphism $p : D_n \rightarrow G$ such that $\hat{v}_0 V_p = r$ and $\hat{v}_n V_p = v$.

If $v = r$, then we may take $n = 0$ and $p = c$.

So suppose that $v \in V_G \setminus \{r\}$.

Because of (AcCofib 5) we may choose $n \geq 1$ and $e_i \in E_G$ for $i \in [1, n]$ such that $e_1 s_G \in \{r\}$, such that $e_i t_G = e_{i+1} s_G$ for $i \in [1, n-1]$ and such that $e_n t_G = v$.

So for $v \in V_G$ we let $\hat{e}_i E_p := e_{i+1}$ for $i \in [0, n-1]$ and $\hat{v}_i V_p := e_{i+1} s_G$ for $i \in [0, n-1]$ and $\hat{v}_n V_p := e_n t_G$.

Then $p : D_n \rightarrow G$ is a graph morphism because we have $\hat{e}_i E_p s_G = e_{i+1} s_G = \hat{v}_i V_p = \hat{e}_i s_{D_n} V_p$ for $i \in [0, n-1]$ and since we have $\hat{e}_i E_p t_G = e_{i+1} t_G = e_{i+2} s_G = \hat{v}_{i+1} V_p = \hat{e}_i t_{D_n} V_p$ for $i \in [0, n-2]$ and $\hat{e}_{n-1} E_p t_G = e_n t_G = \hat{v}_n V_p = \hat{e}_n t_{D_n} V_p$.

Suppose given a tree G .

We have to show that there exists an acyclic cofibration $c : D_0 \rightarrow G$.

We let $\hat{v}_0 V_c := r$.

We have to show that the graph morphism c is an acyclic cofibration; cf. Definition 162.

Ad (AcCofib 1, 2). The graph morphism $c : D_0 \rightarrow G$ is injective; i.e. the maps V_c and E_c are injective.

Ad (AcCofib 3). Because of (Tree 1) we have $|\{e \in E_G : (e) t_G = v\}| = 1$ for $v \in V_G \setminus V_{D_0 c} = V_G \setminus \{r\}$.

Ad (AcCofib 4). Because of (Tree 2) we have $(e) t_G \neq r$ for $e \in E_G$ and thus $(e) t_G \in V_G \setminus \{r\}$ for $e \in E_G = E_G \setminus E_{D_0 c}$.

Ad (AcCofib 5). Suppose given $v \in V_G \setminus V_{D_0 c}$. Because of (Tree 3) we have a path from r to v , i.e. we have a graph morphism $p : D_n \rightarrow G$ such that $\hat{v}_0 V_p = r$ and $\hat{v}_n V_p = v$.

We let $e_i := \hat{e}_{i-1} E_p$ for $i \in [1, n]$. So we have $e_1 s_G = \hat{e}_0 E_p s_G = \hat{e}_0 s_G V_p = \hat{v}_0 V_p = r \in V_{D_n c}$. And we have $e_i t_G = \hat{e}_{i-1} E_p t_G = \hat{e}_{i-1} t_G V_p = \hat{e}_i s_G V_p = \hat{e}_i E_p s_G = e_{i+1} s_G$ for $i \in [1, n-1]$ and $e_n t_G = \hat{e}_{n-1} E_p t_G = \hat{e}_{n-1} t_G V_p = \hat{v}_n V_p = v$. \square

Example 179 Let $n \in \mathbb{N}$.

Since the graph morphism $\iota_{0,n} : D_0 \rightarrow D_n$ is an acyclic cofibration, the graph D_n is a tree; cf. Definition 108, Remark 166 and Remark 178.

This can also be verified directly. The graph D_n has root $r = \hat{v}_0$. Then (Tree 1) and (Tree 2) hold since we have $\hat{e}_{k-1} t_{D_n} = \hat{v}_k$ for $k \in [1, n]$. Then (Tree 3) holds since we have the paths $\iota_{k,n} : D_k \rightarrow D_n$ for $k \in [0, n]$.

Now we can give another proof of Remark 112.

Remark 180 Suppose given a graph G and a vertex $x \in V_G$.

Then the graph $\text{Tree}(x, G)$ is a tree with root $r := (x;) \in V_{\text{Tree}(x, G)}$.

Proof. We have to show that there exists an acyclic cofibration $c : D_0 \rightarrow \text{Tree}(x, G)$.

Let $\hat{v}_0 V_c := (x;) \in V_{\text{Tree}(x, G)}$.

We will show that the graph morphism $c : D_0 \rightarrow \text{Tree}(x, G)$ with $\hat{v}_0 V_c := (x;)$ is an acyclic cofibration.

Ad (AcCofib 1, 2). The maps V_c and E_c are injective.

Ad (AcCofib 3). We have to show that $|\{e \in E_{\text{Tree}(x, G)} : e t_{\text{Tree}(x, G)} = v\}| \stackrel{!}{=} 1$ for $v \in V_{\text{Tree}(x, G)} \setminus \{(x;)\}$.

For a path $(x; e_1, \dots, e_n) \in V_{\text{Tree}(x, G)}$ we have $\{e \in E_{\text{Tree}(x, G)} : e t_{\text{Tree}(x, G)} = (x; e_1, \dots, e_n)\} = \{(x; e_1, \dots, e_{n-1}), e_n, (x; e_1, \dots, e_n)\}$.

Ad (AcCofib 4). Suppose given $e \in E_{\text{Tree}(x, G)} \setminus E_{D_0} = E_{\text{Tree}(x, G)} \setminus \emptyset$.

We have $e = (\gamma, e_n, \gamma \cdot e_n)$ for a path γ and an edge $e_n \in E_G$.

Then we have $\gamma \cdot e_n \neq (x;)$.

Ad (AcCofib 5) Suppose given $(x; e_1, \dots, e_n) \in V_{\text{Tree}(x, G)}$.

Then there exists

$$(x;) \xrightarrow{((x;), e_1, (x; e_1))} (x; e_1) \xrightarrow{((x; e_1), e_2, (x; e_1 e_2))} (x; e_1, e_2) \xrightarrow{((x; e_1, e_2), e_3, (x; e_1, e_2, e_3))} \dots \quad \dots \longrightarrow (x; e_1, \dots, e_n) .$$

□

3.7 Summary of some notations

Reminder 181 Suppose given a graph morphism $f : G \rightarrow H$.

To indicate that f is a quasiisomorphism, we often write $G \xrightarrow{\sim} H$; cf. Definition 115.

To indicate that f is a fibration, we often write $G \xrightarrow{\dashv} H$; cf. Definition 127.(1).

To indicate that f is an acyclic fibration, we often write $G \xrightarrow{\dashv} H$; cf. Definition 138.

To indicate that f is a cofibration, we often write $G \xrightarrow{\dashv} H$; cf. Definition 144.

To indicate that f is an acyclic cofibration, we often write $G \xrightarrow{\dashv} H$; cf. Definition 162.

Reminder 182

(1) We denote $\text{Qis} = \{f \in \text{Mor}(\text{Gph}) : f \text{ is a quasiisomorphism}\}$; cf. Definition 115.

(2) We denote $\text{Fib} = \{f \in \text{Mor}(\text{Gph}) : f \text{ is a fibration}\}$; cf. Definition 127.(1).

We denote $\text{AcFib} = \text{Fib} \cap \text{Qis} = \{f \in \text{Mor}(\text{Gph}) : f \text{ is an acyclic fibration}\}$; cf. Definition 138.

(3) We denote $\text{Cofib} = \{f \in \text{Mor}(\text{Gph}) : f \text{ is a cofibration}\}$; cf. Definition 144.

We denote $\text{AcCofib} = \{f \in \text{Mor}(\text{Gph}) : f \text{ is an acyclic cofibration}\}$; cf. Definition 162 and Lemma 175.

Chapter 4

Factorization of graph morphisms

Remark 183 Suppose given a graph morphism $f : X \rightarrow Y$.

Consider the graph $F := \coprod_{x \in V_X} \text{Tree}(x V_f, Y)$; cf. Definitions 109 and 90.

Consider the discrete subgraph $\dot{X} \subseteq X$; cf. Definition 71.

We define the graph morphism

$$\begin{aligned} a : \quad \dot{X} &\rightarrow F = \coprod_{x \in V_X} \text{Tree}(x V_f, Y) \\ V_a : \quad V_X = V_{\dot{X}} \ni x &\mapsto (x, (x V_f;)) \\ E_a : \quad E_{\dot{X}} &= \emptyset. \end{aligned}$$

Recall that $(x V_f;)$ is an empty path at $x V_f$; cf. Definition 107.

Then the graph morphism $a : \dot{X} \rightarrow F$ is an acyclic cofibration.

Cf. also Remark 178.

Proof.

Note that for $x \in V_X$, the graph $\text{Tree}(x V_f, Y)$ is a tree with root $r := (x V_f;)$; cf. Remark 112 and Definition 108.

We have to show that (AcCofib 1–5) hold for $a : \dot{X} \rightarrow F$; cf. Definition 162.

Ad (AcCofib 1, 2): The maps V_a and E_a are injective.

Ad (AcCofib 3): Suppose given $v_F \in V_F \setminus V_{\dot{X}a}$.

There exists a unique vertex $x \in X$, a unique integer $n \geq 1$ and a unique path $p : D_n \rightarrow Y$ in Y from $x V_f$ such that $v_F = (x, p)$. Note that $\hat{v}_0 V_p = x V_f$.

So the path p is not the root of the graph $\text{Tree}(x V_f, Y)$; i.e. $p \neq r$; cf. Definition 108.

So because of (Tree 1) we have $|\{e \in E_{\text{Tree}(x V_f, Y)} : (e) t_{\text{Tree}(x V_f, Y)} = p\}| = 1$ since $p \in V_{\text{Tree}(x V_f, Y)} \setminus \{r\}$.

Let $e \in E_{\text{Tree}(x V_f, Y)}$ be this edge with $e t_{\text{Tree}(x V_f, Y)} = p$.

So we have $(x, e) t_F \stackrel{\text{Def. 90}}{=} (x, e t_{\text{Tree}(x V_f, Y)}) = (x, p)$.

Conversely, suppose given an edge $(\tilde{x}, \tilde{e}) \in E_F$ such that $(\tilde{x}, \tilde{e}) t_F = (x, p)$. Then we have $(x, p) = (\tilde{x}, \tilde{e}) t_F = (\tilde{x}, \tilde{e} t_{\text{Tree}(\tilde{x} V_f, Y)})$.

So we have $x = \tilde{x}$ and $p = \tilde{e} t_{\text{Tree}(\tilde{x} V_f, Y)} = \tilde{e} t_{\text{Tree}(x V_f, Y)}$.

Because of (Tree 1) we deduce that $e = \tilde{e}$.

Ad (AcCofib 4).

Suppose given $e_F \in E_F \setminus E_{\dot{X}_a} = E_F$.

There exist a unique vertex $x \in V_X$ and a unique edge $e \in E_{\text{Tree}(x V_f, Y)}$ such that $e_F = (x, e)$.

So we have $e_F t_F = (x, e) t_F = (x, e t_{\text{Tree}(x V_f, Y)})$.

Because of (Tree 2) we have $e t_{\text{Tree}(x V_f, Y)} \neq (x V_f;)$.

So we have $e_F t_F \in V_F \setminus V_{\dot{X}_a}$.

Ad (AcCofib 5).

Suppose given $v_F \in V_F \setminus V_{\dot{X}_a}$.

Then there exist a vertex $p \in V_{\text{Tree}(x V_f, Y)}$ such that $v_F = (x, p)$ and $p \neq (x V_f;) =: r$.

Because of (Tree 3) we may choose $n \geq 0$ and a path $\alpha : D_n \rightarrow \text{Tree}(x V_f, Y)$ from $\hat{v}_0 V_\alpha = r = (x V_f;)$ to $\hat{v}_n V_\alpha = p$.

$$\begin{array}{ccc} \text{Tree}(x V_f, Y) & \xrightarrow{\iota_x} & F \\ \alpha \uparrow & \nearrow \alpha \cdot \iota_x & \\ D_n & & \end{array}$$

Then $\alpha \cdot \iota_x$ is a path from $\hat{v}_0(\alpha \cdot \iota_x) = r \iota_x = (x, r) = (x, x V_f;) = x V_a$ to $\hat{v}_n(\alpha \cdot \iota_x) = p \iota_x = (x, p) = v_F$; cf. Remark 164.

So the graph morphism $a : \dot{X} \rightarrow F$ is an acyclic cofibration. □

The following lemma is due to BISSON and TSEMO [3, Prop. 3.5].

Lemma 184 Suppose given a graph morphism $f : X \rightarrow Y$.

Then there exists a commutative triangle in Gph as follows.

$$\begin{array}{ccc} & G & \\ g \nearrow & & \searrow q \\ X & \xrightarrow{f} & Y \end{array}$$

Proof. We consider the discrete subgraph $\dot{X} \subseteq X$ and the inclusion morphism $\text{o}_X : \dot{X} \rightarrow X$; cf. Definition 71.

Consider the graph $F := \coprod_{x \in V_X} \text{Tree}(x V_f, Y)$.

We define the graph morphism

$$\begin{array}{l} a : \dot{X} \rightarrow F = \coprod_{x \in V_X} \text{Tree}(x V_f, Y) \\ V_a : V_X = V_{\dot{X}} \ni x \mapsto (x, (x V_f;)) \\ E_a : E_{\dot{X}} = \emptyset. \end{array}$$

Recall that $(x V_f ;)$ is an empty path at $x V_f$.

Note that the graph morphism $a : \dot{X} \rightarrow F$ is an acyclic cofibration; cf. Remark 183; cf. also Remark 178.

We now form the pushout as in Construction 83; cf. Remark 171.

$$\begin{array}{ccc} \dot{X} & \xrightarrow{a} & F \\ \circ_X \downarrow & & \downarrow h \\ X & \xrightarrow{g} & \tilde{F} \end{array}$$

We have the graph morphism $p_{x V_f} : \text{Tree}(x V_f, Y) \rightarrow Y$ for $x \in V_X$; cf. Definition 113.

So because of the universal property of the coproduct we have the graph morphism

$$p := (p_{x V_f})_{x \in V_X} : F = \coprod_{x \in V_X} \text{Tree}(x V_f, Y) \rightarrow Y$$

with $(x, v_{\text{Tree}(x V_f, Y)}) V_p = v_{\text{Tree}(x V_f, Y)} V_{p_{x V_f}}$ and $(x, e_{\text{Tree}(x V_f, Y)}) E_p = e_{\text{Tree}(x V_f, Y)} E_{p_{x V_f}}$; cf. Definitions 90 and 91.

We have the following commutative diagram.

$$\begin{array}{ccc} \dot{X} & \xrightarrow{a} & F \\ \circ_X \downarrow & & \downarrow p \\ X & \xrightarrow{f} & Y \end{array}$$

In fact, suppose given $x \in V_{\dot{X}} = V_X$.

We have $(x V_a) V_p = (x, (x V_f ;)) V_p = (x V_f ;) V_{p_{x V_f}} \stackrel{\text{Def. 113}}{=} x V_f = x V_{\circ_X} V_f$.

Because \tilde{F} is a pushout, there exists a unique graph morphism $q : \tilde{F} \rightarrow Y$ such that $hq = p$ and $gq = f$. We *claim* that the graph morphism $q : \tilde{F} \rightarrow Y$ is a fibration.

$$\begin{array}{ccc} \dot{X} & \xrightarrow{a} & F \\ \circ_X \downarrow & & \downarrow h \\ X & \xrightarrow{g} & \tilde{F} \end{array} \begin{array}{c} \searrow p \\ \downarrow q \\ \searrow f \end{array} \begin{array}{c} \\ \\ Y \end{array}$$

Suppose given a vertex $z \in V_{\tilde{F}} = V_{Xg} \cup V_{Fh}$; cf. Remark 37.

We consider two cases.

Case 1: $z \in V_{Xg}$.

We choose a vertex $x \in V_X$ such that $z = x V_g$.

We have to show that the map $E_q|_{\tilde{F}(z, *)}^{Y(z V_q, *)} : \tilde{F}(z, *) \rightarrow Y(z V_q, *)$ is surjective.

So suppose given an edge $e \in Y(z V_q, *) \subseteq E_Y$. We have $e_{s_Y} = z V_q = x V_g V_q = x V_f$. We write $y := e_{t_Y} \in V_Y$. So for the edge $e \in E_Y$ we have the path $(x V_f ; e)$ in Y of length

1 from $xV_f = es_Y$ to $y = et_Y$. This path $(xV_f; e)$ is a vertex in $\text{Tree}(xV_f, Y)$. So we have the vertex $(x, (xV_f; e)) \in V_F$. We have the vertex $(x, (xV_f;)) \in V_F$. We have the edge $e_T := (x, ((xV_f;), e, (xV_f; e))) \in E_F$; cf. Definition 109. So $e_T s_F = (x, (xV_f;))$ and $e_T t_F = (x, (xV_f; e))$; cf. Definition 90.

We have $e_T E_h \in E_{\tilde{F}}$, where its source vertex is $e_T E_h s_{\tilde{F}} = e_T s_F V_h = (x, (xV_f;)) V_h = xV_a V_h = xV_{o_X} V_g = xV_g = z$. So the edge $e_T E_h \in E_{\tilde{F}}$ has source z and thus $e_T E_h \in \tilde{F}(z, *)$.

We show that $(e_T E_h) E_q \stackrel{!}{=} e \in Y(zV_q, *)$.

We have $E_h E_q = E_p$. So we have

$$e_T E_h E_q = e_T E_p = (x, ((xV_f;), e, (xV_f; e))) E_p \stackrel{\text{Def. 91}}{=} ((xV_f;), e, (xV_f; e)) E_{p_{xV_f}} \stackrel{\text{Def. 113}}{=} e.$$

Case 2: $z \in V_{Fh}$.

We may choose a vertex in V_F , consisting of a vertex $x \in V_X$ and a vertex v in $\text{Tree}(xV_f, Y)$, i.e. a path $v = (xV_f; e_0, \dots, e_{n-1})$ in Y , such that $z = (x, v) V_h$.

We write $y_n := v V_{p_{xV_f}} \in V_Y$; cf. Definition 113.

We have to show that the map $E_q|_{\tilde{F}(z, *)}^{Y(zV_q, *)} : \tilde{F}(z, *) \rightarrow Y(zV_q, *)$ is surjective.

We have $zV_q = (x, v) V_h V_q \stackrel{V_h V_q = V_p}{=} (x, v) V_p \stackrel{\text{Def. 91}}{=} v V_{p_{xV_f}} = y_n$.

So suppose given an edge $e \in Y(zV_q, *) = Y(y_n, *)$. Note that $es_Y = y_n$.

We have to find an edge e_F in $\tilde{F}(z, *)$ such that $e_F E_q \stackrel{!}{=} e$.

We first consider the edge $\tau := (x, ((xV_f; e_0, \dots, e_{n-1}), e, (xV_f; e_0, \dots, e_{n-1}, e))) \in E_F$ which has the source vertex $(x, v) = (x, (xV_f; e_0, \dots, e_{n-1}))$ and the target vertex $(x, (xV_f; e_0, \dots, e_{n-1}, e))$.

We have $\tau E_h s_{\tilde{F}} = \tau s_F V_h = (x, v) V_h = z$. So we have $\tau E_h \in \tilde{F}(z, *)$.

We now show that $\tau E_h E_q \stackrel{!}{=} e$.

We have $hq = p$ and thus we have $E_h E_q = E_p$.

So we have $\tau E_h E_q = \tau E_p \stackrel{\text{Def. 91}}{=} ((xV_f; e_0, \dots, e_{n-1}), e, (xV_f; e_0, \dots, e_{n-1}, e)) E_{p_{xV_f}} \stackrel{\text{Def. 113}}{=} e$.

So the graph morphism $q : \tilde{F} \rightarrow Y$ is a fibration.

$$\begin{array}{ccc}
 \dot{X} & \xrightarrow{a} & F \\
 \circ_X \downarrow & & \downarrow h \\
 X & \xrightarrow{g} & \tilde{F} \\
 & \searrow f & \downarrow q \\
 & & Y
 \end{array}$$

□

Lemma 185 In \mathbf{Gph} , we have $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$.

Proof. We show that $\text{AcCofib} \stackrel{!}{\supseteq} \text{Cofib} \cap \text{Qis}$.

Suppose given a quasiisomorphism $f : X \rightarrow Y$ that is a cofibration, i.e. that satisfies $(\text{LLP}_{\text{AcFib}})$; cf. Definition 144.

We have to show that f is an acyclic cofibration, i.e. that (AcCofib 1–5) hold for the graph morphism f ; cf. Definition 162.

By Lemma 184 we may choose an acyclic cofibration $w : X \rightarrow Z$ and a fibration $p : Z \rightarrow Y$ such that $f = wp$.

$$\begin{array}{ccc} & Z & \\ w \circlearrowleft & \nearrow & p \\ X & \xrightarrow[\approx]{f} & Y \end{array}$$

Because the graph morphism $f : X \dashrightarrow Y$ is a quasiisomorphism and because the graph morphism $w : X \circlearrowleft Z$ is an acyclic cofibration and thus a quasiisomorphism, we may conclude by (2 of 3) that the fibration $p : Z \rightarrow Y$ is a quasiisomorphism and thus an acyclic fibration; cf. Remark 123 and Definition 138.

So we consider the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{w} & Z \\ f \bullet \Downarrow & & \Downarrow p \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

Because the graph morphism $f : X \rightarrow Y$ is a cofibration and thus in $\square \text{AcFib}$, there exists a graph morphism $h : Y \rightarrow Z$ such that $fh = w$ and such that $hp = \text{id}_Y$.

$$\begin{array}{ccc} X & \xrightarrow{w} & Z \\ f \bullet \Downarrow & \nearrow h & \Downarrow p \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

We consider the following commutative diagram.

$$\begin{array}{ccccc} & & \text{id}_X & & \\ & & \curvearrowright & & \\ X & \xrightarrow{\text{id}_X} & X & \xrightarrow{\text{id}_X} & X \\ f \downarrow & & \downarrow w & & \downarrow f \\ Y & \xrightarrow{h} & Z & \xrightarrow{p} & Y \\ & & \text{id}_Y & & \end{array}$$

Since the set of acyclic cofibrations AcCofib is closed under retracts, the graph morphism $f : X \rightarrow Y$ is an acyclic cofibration; cf. Remark 172.

So we have $\text{AcCofib} \supseteq \text{Cofib} \cap \text{Qis}$.

We have $\text{AcCofib} \subseteq \text{Cofib} \cap \text{Qis}$; cf. Lemma 175.

Altogether, we have $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$. □

Lemma 186 Suppose given a graph morphism $f : X \rightarrow Y$.

Then there exist a commutative triangle in Gph as follows, where $(C_n, f')_{\text{Gph}}$ is surjective for $n \geq 1$.

$$\begin{array}{ccc} & \tilde{X} & \\ g' \nearrow & & \searrow f' \\ X & \xrightarrow{f} & Y \end{array}$$

Proof. For $n \geq 1$, we let $M_n := (C_n, Y)_{\text{Gph}} \setminus \text{Im}(C_n, g)_{\text{Gph}}$.

Let $M := \coprod_{n \geq 1} M_n = \{(n, u) : n \geq 1, u \in M_n\}$.

Let $C := \coprod_{(n,u) \in M} C_n$.

We have $V_C = \{((n, u), v_i) : n \geq 1, u \in M_n, i \in \mathbb{Z}/n\mathbb{Z}\}$.

And we have $E_C = \{((n, u), e_i) : n \geq 1, u \in M_n, i \in \mathbb{Z}/n\mathbb{Z}\}$.

We write

$$(n, u, v_i) := ((n, u), v_i)$$

and

$$(n, u, e_i) := ((n, u), e_i).$$

Using Definition 91, we have the graph morphism

$$\begin{aligned} h = (u)_{(n,u) \in M} : \quad C &\rightarrow Y \\ (n, u, v_i) &\mapsto v_i V_u \text{ for } i \in \mathbb{Z}/n\mathbb{Z} \\ (n, u, e_i) &\mapsto e_i E_u \text{ for } i \in \mathbb{Z}/n\mathbb{Z}. \end{aligned}$$

Let $X' := X \sqcup C$; cf. Definition 85. Then

$$V_{X'} = V_X \sqcup V_C = \{(1, v_X) : v_X \in V_X\} \cup \{(2, v_C) : v_C \in V_C\}$$

and

$$E_{X'} = E_X \sqcup E_C = \{(1, e_X) : e_X \in E_X\} \cup \{(2, e_C) : e_C \in E_C\}.$$

Moreover,

$$\begin{aligned} s_{X'} : \quad E_{X'} &\rightarrow V_{X'} \\ (1, e_X) &\mapsto (1, e_X s_X) \\ (2, e_C) &\mapsto (2, e_C s_C) \end{aligned}$$

and

$$\begin{aligned} t_{X'} : \quad E_{X'} &\rightarrow V_{X'} \\ (1, e_X) &\mapsto (1, e_X t_X) \\ (2, e_C) &\mapsto (2, e_C t_C). \end{aligned}$$

We have the graph morphism $f' := \begin{pmatrix} g \\ h \end{pmatrix} : X \sqcup C \rightarrow Y$ satisfying $\iota_1 f' = g$ and $\iota_2 f' = h$; cf. Definition 86.

Writing $g' := \iota_1 : X \rightarrow X \sqcup C$, we have $g' f' = g$.

Claim 1. The graph morphism $g' : X \rightarrow X' = X \sqcup C$ is a cofibration; cf. Definition 144.

The cyclic graph C_n is cofibrant for $n \geq 1$; cf. Remark 151. So the graph $C = \coprod_{(n,u) \in M} C_n$ is cofibrant, i.e. the graph morphism $\iota_C : \emptyset \rightarrow C$ is a cofibration; cf. Definitions 70, 150 and Remark 152.

So the graph morphism $g' : X \rightarrow X \sqcup C$ is a cofibration as a pushout of a cofibration; cf. Remark 157.

This proves *Claim 1*.

Claim 2. Given $n \geq 1$. The map $(C_n, f')_{\text{Gph}} : (C_n, X \sqcup C)_{\text{Gph}} \rightarrow (C_n, Y)_{\text{Gph}}$ is surjective.

Suppose given a graph morphism $u : C_n \rightarrow Y$ in $(C_n, Y)_{\text{Gph}}$.

Case 1: $u \in \text{Im}(C_n, g)_{\text{Gph}}$.

Then we may choose a graph morphism $\tilde{u} : C_n \rightarrow X$ in $(C_n, X)_{\text{Gph}}$ such that $u = \tilde{u}g$.

$$\begin{array}{ccc}
 X \sqcup C & \xrightarrow{f' = \begin{pmatrix} g \\ h \end{pmatrix}} & Y \\
 g' \uparrow & \nearrow g & \\
 X & & \\
 \tilde{u} \uparrow & \nearrow u & \\
 C_n & &
 \end{array}$$

So we have $\tilde{u}g' \in (C_n, X \sqcup C)_{\text{Gph}}$ and $(\tilde{u}g')(C_n, f')_{\text{Gph}} = \tilde{u}g' f' = \tilde{u}g = u$.

Case 2: $u \in (C_n, Y)_{\text{Gph}} \setminus \text{Im}(C_n, g)_{\text{Gph}}$.

So we have $u \in M_n$, i.e. $(n, u) \in M$.

We have the graph morphism $\iota_{(n,u)} : C_n \rightarrow C$; cf. Definition 90.

We have $\iota_{(n,u)} h = \iota_{(n,u)}(u)_{(n,u) \in M} \stackrel{\text{Def. 91}}{=} u$.

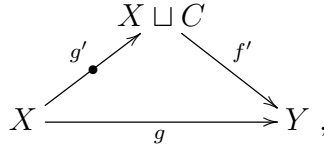
Consider the graph morphism $\iota_2 : C \rightarrow X \sqcup C$.

We have $\iota_{(n,u)} \cdot \iota_2 \in (C_n, X \sqcup C)_{\text{Gph}}$ and $(\iota_{(n,u)} \cdot \iota_2)(C_n, f')_{\text{Gph}} = \iota_{(n,u)} \cdot \iota_2 \cdot f' = \iota_{(n,u)} \cdot h = u$.

$$\begin{array}{ccc}
 X \sqcup C & \xrightarrow{f' = \begin{pmatrix} g \\ h \end{pmatrix}} & Y \\
 \iota_2 \uparrow & \nearrow h & \\
 C & & \\
 \iota_{n,u} \uparrow & \nearrow u & \\
 C_n & &
 \end{array}$$

This proves *Claim 2*.

So we have the commutative triangle

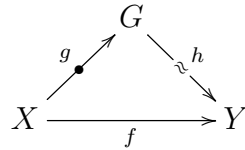


where $(C_n, f')_{\text{Gph}}$ is surjective for $n \geq 1$. □

The following lemma is an iterated version of an argument of BISSON and TSEMO [3, Prop. 4.6].

Lemma 187 Suppose given a graph morphism $f : X \rightarrow Y$ such that $(C_n, f)_{\text{Gph}}$ is surjective for $n \geq 1$.

Then there exists a commutative triangle in Gph as follows.



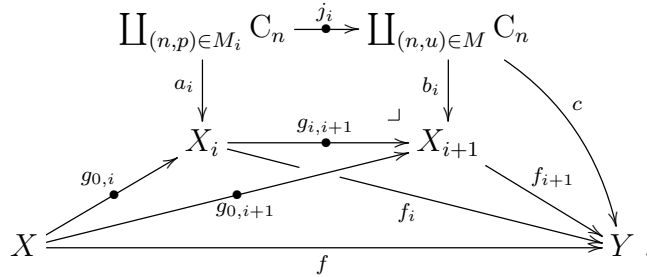
Proof. We have the graph morphism $f_0 := f : X \rightarrow Y$.

Let $X_0 := X$.

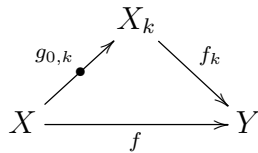
Let $g_{0,0} := \text{id}_X : X \rightarrow X_0$.

Then $g_{0,0} \cdot f_0 = f$.

For $i \geq 0$, we shall recursively construct a commutative diagram in Gph as follows for suitably defined sets M_i and M and j_i resulting from Lemma 156.



Suppose that the commutative triangle



is constructed for a given $k \geq 0$.

Let $M := \{(n, u) : n \geq 1, C_n \xrightarrow{u} Y\}$.

Then we have the graph $\coprod_M C_n := \coprod_{(n,u) \in M} C_n$ with

$$\begin{aligned} V_{\coprod_M C_n} &= \{(n, u, v_i) : n \geq 1, C_n \xrightarrow{u} Y, i \in \mathbb{Z}/n\mathbb{Z}\} \\ E_{\coprod_M C_n} &= \{(n, u, e_i) : n \geq 1, C_n \xrightarrow{u} Y, i \in \mathbb{Z}/n\mathbb{Z}\}. \end{aligned}$$

Let $M_k := \{(n, p) : n \geq 1, C_n \xrightarrow{p} X_k\}$.

Then we have the graph $\coprod_{M_k} C_n := \coprod_{(n,p) \in M_k} C_n$ with

$$\begin{aligned} V_{\coprod_{M_k} C_n} &= \{(n, p, v_i) : n \geq 1, C_n \xrightarrow{p} X_k, i \in \mathbb{Z}/n\mathbb{Z}\} \\ E_{\coprod_{M_k} C_n} &= \{(n, p, e_i) : n \geq 1, C_n \xrightarrow{p} X_k, i \in \mathbb{Z}/n\mathbb{Z}\}. \end{aligned}$$

Using the maps $\mu : M_k \rightarrow M : (n, p) \mapsto (n, pf_k)$ and $\nu : M \rightarrow \mathbb{N} : (n, u) \mapsto n$, Lemma 156 gives the cofibration

$$\begin{aligned} j_k &: \coprod_{M_k} C_n \rightarrow \coprod_M C_n \\ V_{j_k} &: (n, p, v_i) \mapsto (n, pf_k, v_i) \\ E_{j_k} &: (n, p, e_i) \mapsto (n, pf_k, e_i). \end{aligned}$$

Using Definition 91, we obtain the following graph morphisms.

$$\begin{aligned} a_k &:= (p)_{(n,p)} : \coprod_{M_k} C_n \rightarrow X_k \\ V_{a_k} &: (n, p, v_i) \mapsto v_i V_p \text{ for } v_i \in V_{C_n} \\ E_{a_k} &: (n, p, e_i) \mapsto e_i E_p \text{ for } e_i \in E_{C_n} \\ c &:= (u)_{(n,u) \in M} : \coprod_M C_n \rightarrow Y \\ V_c &: (n, u, v_i) \mapsto v_i V_u \text{ for } v_i \in V_{C_n} \\ E_c &: (n, u, e_i) \mapsto e_i E_u \text{ for } e_i \in E_{C_n} \end{aligned}$$

We note that since $(C_n, f_k)_{\text{Gph}}$ is surjective, the graph morphism j_k is surjective.

We show that we have $j_k c \stackrel{!}{=} a_k f_k$.

Suppose given $(n, p, v_i) \in V_{\coprod_{M_i} C_n}$.

We have $(n, p, v_i) V_{j_k} V_c = (n, pf_k, v_i) V_c = v_i V_{pf_k} = v_i V_p V_{f_k} = (n, p, v_i) V_{a_k} V_{f_k}$.

Suppose given $(n, p, e_i) \in E_{\coprod_{M_i} C_n}$.

We have $(n, p, e_i) E_{j_k} E_c = (n, pf_k, e_i) E_c = e_i E_{pf_k} = e_i E_p E_{f_k} = (n, p, e_i) E_{a_k} E_{f_k}$.

So we have $j_k c = a_k f_k$.

Forming the pushout, we obtain the following commutative diagram.

$$\begin{array}{ccc} \coprod_{(n,p) \in M_i} C_n & \xrightarrow{j_i} & \coprod_{(n,u) \in M} C_n \\ \downarrow a_i & & \downarrow b_i \\ X_i & \xrightarrow{g_{i,i+1}} & X_{i+1} \\ \nearrow g_{0,i} & & \searrow f_{i+1} \\ X & \xrightarrow{f} & Y \\ \nearrow g_{0,i+1} & & \end{array}$$

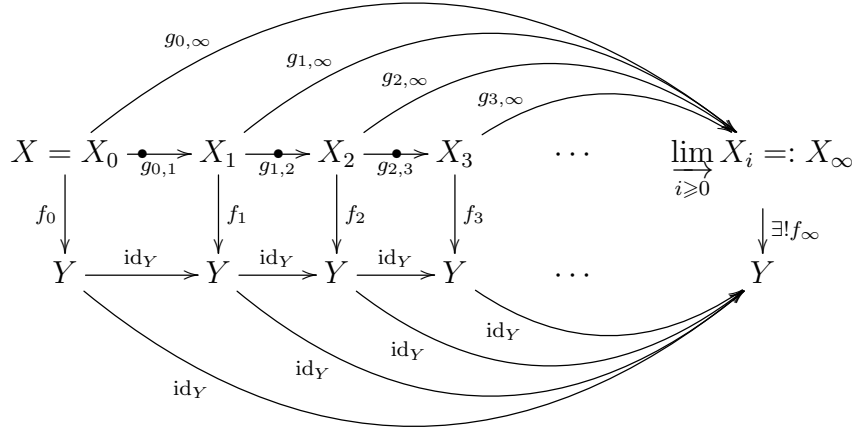
In this diagram, the graph morphism $g_{k,k+1} : X_k \rightarrow X_{k+1}$ is a cofibration; cf. Remark 148.

With respect to the morphisms $X_i \xrightarrow{g_{i,i+1}} X_{i+1}$ for $i \geq 0$, we let $X_\infty := \varinjlim_{i \geq 0} X_i$; cf. Definition 98 and Lemma 99.

We have graph morphisms $g_{i,\infty} : X_i \rightarrow X_\infty$ such that $g_{i,i+1} \cdot g_{i+1,\infty} = g_{i,\infty}$ for $i \geq 0$; cf. Definition 98.

Since $g_{i,i+1}$ is a cofibration for $i \geq 0$, we conclude that $g_{0,\infty} : X_0 \rightarrow X_\infty$ is a cofibration; cf. Lemma 102.

There exists a unique graph morphism $f_\infty : X_\infty \rightarrow Y$ such that $g_{i,\infty} \cdot f_\infty = f_i$ for $i \geq 0$; cf. Lemma 99.



By the surjectivity of $(C_n, f)_{\text{Gph}} = (C_n, f_0)_{\text{Gph}} = (C_n, g_{0,\infty})_{\text{Gph}} \cdot (C_n, f_\infty)_{\text{Gph}}$, we conclude that the map $(C_n, f_\infty)_{\text{Gph}}$ is surjective for $n \geq 0$.

We *claim* that the graph morphism $f_\infty : X_\infty \rightarrow Y$ is a quasiisomorphism.

Suppose given $n \geq 1$. It suffices to show that the map $(C_n, f_\infty)_{\text{Gph}}$ is injective.

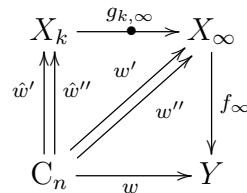
Suppose given graph morphisms $w', w'' : C_n \rightarrow X_\infty$ such that $w' f_\infty = w'' f_\infty$.

We have to show that $w' \stackrel{!}{=} w''$.

We write $w := w' f_\infty = w'' f_\infty$.

There exist $k \geq 0$ and graph morphisms $\hat{w}', \hat{w}'' : C_n \rightarrow X_k$ such that $\hat{w}' g_{k,\infty} = w'$ and such that $\hat{w}'' g_{k,\infty} = w''$; cf. Lemma 101.(2).

We have $\hat{w}' f_k = \hat{w}' g_{k,\infty} f_\infty = w' f_\infty = w = w'' f_\infty = \hat{w}'' g_{k,\infty} f_\infty = \hat{w}'' f_k$.



It now suffices to show that $\hat{w}' g_{k,k+1} \stackrel{!}{=} \hat{w}'' g_{k,k+1}$.

Because then we have $w' = \hat{w}' g_{k,\infty} = \hat{w}' g_{k,k+1} g_{k+1,\infty} = \hat{w}'' g_{k,k+1} g_{k+1,\infty} = \hat{w}'' g_{k,\infty} = w''$.

We have $(n, \hat{w}'), (n, \hat{w}'') \in M_k$.

We show that $\hat{w}'g_{k,k+1} \stackrel{!}{=} \hat{w}''g_{k,k+1}$.

So suppose given $v_i \in V_{C_n}$, where $i \in \mathbb{Z}/n\mathbb{Z}$.

We have to show that $v_i V_{\hat{w}'g_{k,k+1}} \stackrel{!}{=} v_i V_{\hat{w}''g_{k,k+1}}$.

$$\begin{aligned} \text{We have } v_i V_{\hat{w}'g_{k,k+1}} &= v_i V_{\hat{w}'} V_{g_{k,k+1}} \stackrel{\text{Def. } a_k}{=} (n, \hat{w}', v_i) V_{a_k} V_{g_{k,k+1}} = (n, \hat{w}', v_i) V_{a_k g_{k,k+1}} = \\ (n, \hat{w}', v_i) V_{j_k b_k} &= (n, \hat{w}', v_i) V_{j_k} V_{b_k} \stackrel{\text{Def. } j_k}{=} (n, w, v_i) V_{b_k} \stackrel{\text{Def. } j_k}{=} (n, \hat{w}'', v_i) V_{j_k} V_{b_k} = \\ (n, \hat{w}'', v_i) V_{j_k b_k} &= (n, \hat{w}'', v_i) V_{a_k g_{k,k+1}} = (n, \hat{w}'', v_i) V_{a_k} V_{g_{k,k+1}} \stackrel{\text{Def. } a_k}{=} v_i V_{\hat{w}''} V_{g_{k,k+1}} = v_i V_{\hat{w}''g_{k,k+1}}. \end{aligned}$$

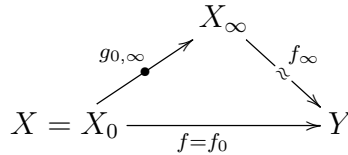
Now suppose given an edge $e_i \in E_{C_n}$.

We have to show that $e_i E_{\hat{w}'g_{k,k+1}} \stackrel{!}{=} e_i E_{\hat{w}''g_{k,k+1}}$.

$$\begin{aligned} \text{We have } e_i E_{\hat{w}'g_{k,k+1}} &= e_i E_{\hat{w}'} E_{g_{k,k+1}} \stackrel{\text{Def. } a_k}{=} (n, \hat{w}', e_i) E_{a_k} E_{g_{k,k+1}} = (n, \hat{w}', e_i) E_{a_k g_{k,k+1}} = \\ (n, \hat{w}', e_i) E_{j_k b_k} &= (n, \hat{w}', e_i) E_{j_k} E_{b_k} \stackrel{\text{Def. } j_k}{=} (n, w, e_i) E_{b_k} \stackrel{\text{Def. } j_k}{=} (n, \hat{w}'', e_i) E_{j_k} E_{b_k} = \\ (n, \hat{w}'', e_i) E_{j_k b_k} &= (n, \hat{w}'', e_i) E_{a_k g_{k,k+1}} = (n, \hat{w}'', e_i) E_{a_k} E_{g_{k,k+1}} \stackrel{\text{Def. } a_k}{=} e_i E_{\hat{w}''} E_{g_{k,k+1}} = e_i E_{\hat{w}''g_{k,k+1}}. \end{aligned}$$

So we have $\hat{w}'g_{k,k+1} = \hat{w}''g_{k,k+1}$.

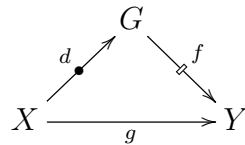
Altogether, we have obtained a commutative triangle



as was to be shown. □

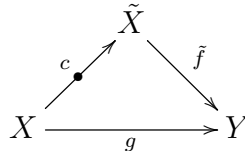
Lemma 188 Suppose given a graph morphism $g : X \rightarrow Y$.

Then there exists a commutative triangle in Gph as follows.

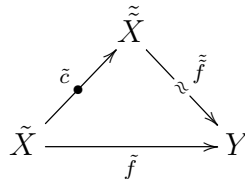


Proof.

Step 1. There exists a commutative triangle in Gph as follows, where $(C_n, \tilde{f})_{\text{Gph}}$ is surjective for $n \geq 1$; cf. Lemma 186.



Step 2. By Lemma 187 there exists a commutative triangle in Gph as follows.



Step 3. By Lemma 184 there exists a commutative triangle in Gph as follows.

$$\begin{array}{ccc}
 & G & \\
 \tilde{c} \circ \tilde{\circ} \nearrow & & \searrow f \\
 \tilde{X} & \xrightarrow{\approx} & Y \\
 & \tilde{f} &
 \end{array}$$

Because Qis satisfies (2 of 3) and because $\text{AcCofib} \subseteq \text{Qis}$, the fibration $f : G \twoheadrightarrow Y$ is a quasiisomorphism and thus an acyclic fibration; cf. Lemma 175 and Definition 138.

In conclusion we have the following commutative diagram in Gph.

$$\begin{array}{ccccc}
 & & \tilde{X} & \xrightarrow{\tilde{c}} & G \\
 & & \tilde{\circ} \nearrow & & \downarrow f \\
 & \tilde{X} & & \tilde{f} \searrow & \\
 c \nearrow & & & & \\
 X & \xrightarrow{g} & Y & & \\
 & & \tilde{f} \searrow & &
 \end{array}$$

The acyclic cofibration $\tilde{c} : \tilde{X} \twoheadrightarrow G$ is in particular a cofibration; cf. Lemma 185.

Since the composite of cofibrations is a cofibration, the graph morphism $d := c\tilde{c} : X \rightarrow G$ is a cofibration; cf. Remark 147. \square

Chapter 5

Subsets of $\text{Mor}(\text{Gph})$ and their lifting sets

Lemma 189 In Gph , we have $\text{AcCofib}^{\square} \subseteq \text{Fib}$.

Proof. Suppose given a graph morphism $f : X \rightarrow Y$ in AcCofib^{\square} ; cf. Definitions 162 and 14.

We have to show that the graph morphism $f : X \rightarrow Y$ is in Fib , i.e. that f is a fibration.

Suppose given a vertex $x \in V_X$.

We have to show that the map $E_{f,x} = E_{f|_{X(x,*)} : Y(xV_f,*)} : X(x,*) \rightarrow Y(xV_f,*)$ is surjective; cf. Definition 127.(1).

Suppose given an edge $e \in Y(xV_f,*)$, i.e. $e \in E_Y$ with $e s_Y = xV_f$.

We write $y := e t_Y \in V_Y$.

We have to find an edge $\tilde{e} \in X(x,*)$ such that $\tilde{e} E_f \stackrel{!}{=} e$.

Let $a : D_0 \rightarrow X$ be defined by $\hat{v}_0 V_a := x$. So $a = (x;)$; cf. Definition 107.

Let $b : D_1 \rightarrow X$ be defined by $\hat{e}_0 E_b := e$, $\hat{v}_0 V_b := e s_Y = xV_f$ and $\hat{v}_1 V_b := e t_Y = y$. So $b = (xV_f; e)$; cf. Definition 107.

In fact, $b : D_1 \rightarrow Y$ is a graph morphism since $\hat{e}_0 s_{D_1} V_b = \hat{v}_0 V_b = e s_Y = \hat{e}_0 E_b s_Y$ and since $\hat{e}_0 t_{D_1} V_b = \hat{v}_1 V_b = e t_Y = \hat{e}_0 E_b t_Y$.

Note that the graph morphism $\iota_{0,1} : D_0 \rightarrow D_1$ is an acyclic cofibration; cf. Remark 166.

We have the following commutative diagram

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & X \\ \downarrow \iota_{0,1} & & \downarrow f \\ D_1 & \xrightarrow{b} & Y \end{array}$$

since $\hat{v}_0 V_a V_f \stackrel{\text{Def. } a}{=} xV_f \stackrel{\text{Def. } b}{=} \hat{v}_0 V_b = \hat{v}_0 V_{\iota_{0,1}} V_b$.

We have $\hat{e}_0 s_{D_1} = \hat{v}_0 = \hat{v}_0 V_{\iota_{0,1}}$.

Because the graph morphism $f : X \rightarrow Y$ is in AcCofib^{\square} there exists a graph morphism

$h : D_1 \rightarrow X$ such that $\iota_{0,1}h = a$ and $hf = b$.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & X \\ \downarrow \iota_{0,1} & \nearrow h & \downarrow f \\ D_1 & \xrightarrow{b} & Y \end{array}$$

Let $\tilde{e} := \hat{e}_0 E_h \in E_X$.

Then we have $\tilde{e} s_X = \hat{e}_0 E_h s_X = \hat{e}_0 s_{D_1} V_h = \hat{v}_0 V_h = \hat{v}_0 V_{\iota_{0,1}} V_h = \hat{v}_0 V_a \stackrel{\text{Def. } a}{=} x$.

So we have $\tilde{e} \in X(x, *)$.

Furthermore, $\tilde{e} E_f = \hat{e}_0 E_h E_f = \hat{e}_0 E_b \stackrel{\text{Def. } b}{=} e$. □

Lemma 190 In Gph , we have $\square \text{Fib} \subseteq \text{AcCofib}$.

Proof. Suppose given a graph morphism $f : X \rightarrow Y$ in $\square \text{Fib}$, i.e. such that $f \square \text{Fib}$.

We have to show that $f \in \text{AcCofib}$, i.e. that $f : X \rightarrow Y$ is an acyclic cofibration.

There exist an acyclic cofibration $X \xrightarrow{a} Z$ and a fibration $Z \xrightarrow{b} Y$ such that $f = ab$; cf. Lemma 184.

We consider the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{a} & Z \\ f \downarrow & & \downarrow b \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

Because $f \square \text{Fib}$ there exists a graph morphism $h : Y \rightarrow Z$ such that $fh = a$ and $hb = \text{id}_Y$.

$$\begin{array}{ccc} X & \xrightarrow{a} & Z \\ f \downarrow & \nearrow h & \downarrow b \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

We consider the following commutative diagram.

$$\begin{array}{ccccc} & & \text{id}_X & & \\ & & \curvearrowright & & \\ X & \xrightarrow{\text{id}_X} & X & \xrightarrow{\text{id}_X} & X \\ f \downarrow & & \downarrow a & & \downarrow f \\ Y & \xrightarrow{h} & Z & \xrightarrow{b} & Y \\ & & \text{id}_Y & & \end{array}$$

Because the set of acyclic cofibrations AcCofib is closed under retracts, the graph morphism $f : X \rightarrow Y$ is an acyclic cofibration; cf. Remark 172 □

Lemma 191 In Gph , we have $\text{AcCofib} = \square \text{Fib}$.

Proof.

We have $\text{AcCofib} \subseteq \square \text{Fib}$; cf. Lemma 174.

We have $\text{AcCofib} \supseteq \square \text{Fib}$; cf. Lemma 190. □

Lemma 192 In Gph , we have $\text{AcCofib}^\square = \text{Fib}$.

Proof.

We have $\text{AcCofib}^\square \subseteq \text{Fib}$; cf. Lemma 189.

We have $\text{AcCofib}^\square \supseteq \text{Fib}$, i.e. $\text{AcCofib} \square \text{Fib}$, i.e. $\text{AcCofib} \subseteq \square \text{Fib}$; cf. Lemma 174. □

Lemma 193 In Gph , we have $\text{Cofib}^\square = \text{AcFib}$.

Proof.

We have $\text{Cofib} \subseteq \square \text{AcFib}$, i.e. $\text{Cofib} \square \text{AcFib}$, i.e. $\text{Cofib}^\square \supseteq \text{AcFib}$; cf. Definition 144.

We have to show that $\text{Cofib}^\square \stackrel{!}{\subseteq} \text{AcFib}$.

So suppose given a graph morphism $f : X \rightarrow Y$ in Cofib^\square .

We have to show that the graph morphism $f : X \rightarrow Y$ is an acyclic fibration; i.e. a fibration that is a quasiisomorphism.

Since $\text{AcCofib} \subseteq \text{Cofib}$ we have $\text{Cofib}^\square \subseteq \text{AcCofib}^\square$; cf. Lemma 175.

So we have $f \in \text{Cofib}^\square \subseteq \text{AcCofib}^\square = \text{Fib}$; cf. Lemma 192.

So the graph morphism $f : X \rightarrow Y$ is a fibration.

We now have to show that the fibration $f : X \rightarrow Y$ is a quasiisomorphism.

Let $n \geq 1$.

The graph morphism $\iota_{C_n} : \emptyset \rightarrow C_n$ is a cofibration; cf. Remark 151.

Suppose given a graph morphism $p : C_n \rightarrow Y$.

We have to show that there exists a unique graph morphism $q : C_n \rightarrow X$ such that $qf = p$.

Existence. We have the commutative diagram

$$\begin{array}{ccc} \emptyset & \longrightarrow & X \\ \downarrow \iota_{C_n} & & \downarrow f \\ C_n & \xrightarrow{p} & Y \end{array};$$

cf. Remark 151.

Since the fibration $f : X \rightarrow Y$ is in Cofib^\square there exists a graph morphism $q : C_n \rightarrow X$ such that $qf = p$.

$$\begin{array}{ccc} \emptyset & \longrightarrow & X \\ \downarrow \iota_{C_n} & \nearrow q & \downarrow f \\ C_n & \xrightarrow{p} & Y \end{array}$$

Uniqueness.

Suppose given graph morphisms $q, \tilde{q} : C_n \rightarrow X$ such that $qf = \tilde{q}f = p$.

We have to show that $q \stackrel{!}{=} \tilde{q}$.

We consider the cofibration $d_{C_n} : C_n \sqcup C_n \rightarrow C_n$; cf. Remark 155.

We consider the following commutative diagram.

$$\begin{array}{ccccc}
 & & C_n \sqcup C_n & \xrightarrow{\begin{pmatrix} q \\ \tilde{q} \end{pmatrix}} & X \\
 & \nearrow \iota_1 & \downarrow d_{C_n} & & \downarrow f \\
 C_n & \xrightarrow{\text{id}_{C_n}} & C_n & \xrightarrow{p} & Y
 \end{array}$$

To show that the diagram is commutative we have to show that $d_{C_n} \stackrel{!}{=} \begin{pmatrix} p \\ \tilde{p} \end{pmatrix} f$.

It suffices to show that $p = \iota_1 d_{C_n} p \stackrel{!}{=} \iota_1 \begin{pmatrix} q \\ \tilde{q} \end{pmatrix} f = qf$ and that $p = \iota_2 d_{C_n} p \stackrel{!}{=} \iota_2 \begin{pmatrix} q \\ \tilde{q} \end{pmatrix} f = \tilde{q}f$. This holds since $qf = p = \tilde{q}f$.

Since the graph morphism $f : X \rightarrow Y$ is in Cofib^\square , there exists a graph morphism $q' : C_n \rightarrow X$ such that $q'f = p$ and $d_{C_n} q' = \begin{pmatrix} q \\ \tilde{q} \end{pmatrix}$.

So we have $q' = \text{id}_{C_n} q' = \iota_1 d_{C_n} q' = \iota_1 \begin{pmatrix} q \\ \tilde{q} \end{pmatrix} = q$ and $q' = \text{id}_{C_n} q' = \iota_2 d_{C_n} q' = \iota_2 \begin{pmatrix} q \\ \tilde{q} \end{pmatrix} = \tilde{q}$.

So we have $q = q' = \tilde{q}$. □

Using Lemma 188, we can alternatively prove Lemma 193 as follows.

Proof. We now have to show that the fibration $f : X \rightarrow Y$ is a quasiisomorphism.

We have the factorization $f = ab$ with $X \xrightarrow{a} G$ and $G \xrightarrow{b} Y$; cf. Lemma 188 below.

We consider the following commutative diagram.

$$\begin{array}{ccc}
 X & \xrightarrow{\text{id}_X} & X \\
 a \downarrow & & \downarrow f \\
 G & \xrightarrow{b} & Y
 \end{array}$$

Because $f \in \text{Cofib}^\square$ there exists a graph morphism $k : G \rightarrow X$ such that $ak = \text{id}_X$ and $kf = b$.

$$\begin{array}{ccc}
 X & \xrightarrow{\text{id}_X} & X \\
 a \downarrow & \nearrow k & \downarrow f \\
 G & \xrightarrow{b} & Y
 \end{array}$$

We consider the following commutative diagram.

$$\begin{array}{ccccc}
 & & \text{id}_X & & \\
 & & \curvearrowright & & \\
 X & \xrightarrow{a} & G & \xrightarrow{k} & X \\
 \downarrow f & & \downarrow b & & \downarrow f \\
 Y & \xrightarrow{\text{id}_Y} & Y & \xrightarrow{\text{id}_Y} & Y \\
 & & \text{id}_Y & & \\
 & & \curvearrowleft & &
 \end{array}$$

Since the set of quasiisomorphisms $\text{Qis} \subseteq \text{Mor}(\text{Gph})$ is closed under retracts the graph morphism $f : X \rightarrow Y$ is a quasiisomorphism; cf. Remark 117

So the graph morphism $f : X \rightarrow Y$ is an acyclic fibration.

Since the set of acyclic fibrations $\text{AcFib} \subseteq \text{Mor}(\text{Gph})$ is closed under retracts the graph morphism $f : X \rightarrow Y$ is an acyclic fibration; cf. Remark 141

Hence we have $\text{Cofib}^\square \subseteq \text{AcFib}$ and so we conclude $\text{Cofib}^\square = \text{AcFib}$. □

Remark 194 We have $\text{AcCofib}^\square \cap \text{AcFib}^\square = \text{Qis}^\square$.

Proof.

Ad $\text{Qis}^\square \subseteq \text{AcCofib}^\square \cap \text{AcFib}^\square$.

Since $\text{AcCofib} \subseteq \text{Qis}$, we have $\text{Qis}^\square \subseteq \text{AcCofib}^\square$; cf. Lemma 175.

Since $\text{AcFib} \subseteq \text{Qis}$, we have $\text{Qis}^\square \subseteq \text{AcFib}^\square$; cf. Definition 138.

So we have $\text{Qis}^\square \subseteq \text{AcCofib}^\square \cap \text{AcFib}^\square$.

Ad $\text{AcCofib}^\square \cap \text{AcFib}^\square \subseteq \text{Qis}^\square$.

Suppose given a graph morphism $f : X \rightarrow Y$ in $\text{AcCofib}^\square \cap \text{AcFib}^\square$.

We have to show that the graph morphism f is in Qis^\square .

Suppose given a commutative diagram in Gph as follows.

$$\begin{array}{ccc}
 G & \xrightarrow{a} & X \\
 \downarrow q & & \downarrow f \\
 H & \xrightarrow{b} & Y
 \end{array}$$

We show that there exists a graph morphism $h : H \rightarrow X$ such that $qh = a$ and $hf = b$.

By Lemma 184 and since Qis satisfies (2 of 3), we have a commutative triangle as follows.

$$\begin{array}{ccc}
 & Z & \\
 c \nearrow & & \searrow d \\
 G & \xrightarrow{\approx} & H \\
 & q &
 \end{array}$$

So we have the following commutative diagram.

$$\begin{array}{ccc}
 G & \xrightarrow{a} & X \\
 \downarrow c \circ \circlearrowleft & & \downarrow f \\
 Z & & Y \\
 \downarrow d \dashv & & \\
 H & \xrightarrow{b} & Y
 \end{array}$$

Since f is in AcCofib^{\square} , there exists a graph morphism $g : Z \rightarrow X$ such that $cg = a$ and $gf = db$.

$$\begin{array}{ccc}
 G & \xrightarrow{a} & X \\
 \downarrow c \circ \circlearrowleft & \nearrow g & \downarrow f \\
 Z & & Y \\
 \downarrow d \dashv & & \\
 H & \xrightarrow{b} & Y
 \end{array}$$

Since f is in AcFib^{\square} and since $gf = db$, there exists a graph morphism $h : H \rightarrow X$ such that $dh = g$ and $hf = b$.

$$\begin{array}{ccc}
 G & \xrightarrow{a} & X \\
 \downarrow c \circ \circlearrowleft & \nearrow g & \downarrow f \\
 Z & \nearrow h & Y \\
 \downarrow d \dashv & & \\
 H & \xrightarrow{b} & Y
 \end{array}$$

So we have $cdh = cg = a$ and $hf = b$.

So f is in Qis^{\square} . □

Remark 195 In Gph , the following statements (1–4) hold.

- (1) We have $\text{AcCofib} = \square \text{Fib}$; cf. Lemma 191.
- (2) We have $\text{AcCofib}^{\square} = \text{Fib}$; cf. Lemma 192.
- (3) We have $\text{Cofib} = \square \text{AcFib}$; cf. Definition 144.
- (4) We have $\text{Cofib}^{\square} = \text{AcFib}$; cf. Lemma 193.

Remark 196 In \mathbf{Gph} , the following statements (1–4) hold.

- (1) We have $(\square \mathbf{Fib})^\square = \mathbf{AcCofib}^\square = \mathbf{Fib}$; cf. Remark 195.
- (2) We have $(\square \mathbf{AcFib})^\square = \mathbf{Cofib}^\square = \mathbf{AcFib}$; cf. Remark 195.
- (3) We have $\square(\mathbf{Cofib}^\square) = \square \mathbf{AcFib} = \mathbf{Cofib}$; cf. Remark 195.
- (4) We have $\square(\mathbf{AcCofib}^\square) = \square \mathbf{Fib} = \mathbf{AcCofib}$; cf. Remark 195.

Remark 197 In \mathbf{Gph} , we have

$$\mathbf{Iso} = \mathbf{Fib} \cap \square \mathbf{Fib} \stackrel{\text{Lemma 191}}{=} \mathbf{Fib} \cap \mathbf{AcCofib} \stackrel{\text{Lemma 185}}{=} \mathbf{Fib} \cap \mathbf{Cofib} \cap \mathbf{Qis} .$$

Proof. We show that $\mathbf{Fib} \cap \mathbf{AcCofib} \stackrel{!}{=} \mathbf{Iso}$.

$$\text{Ad } \mathbf{Iso} \stackrel{!}{\subseteq} \mathbf{Fib} \cap \mathbf{AcCofib}.$$

We have $\mathbf{Iso} \subseteq \mathbf{Fib}$; cf. Remark 129.

We have $\mathbf{Iso} \subseteq \mathbf{AcCofib}$; cf. Remark 165.

$$\text{Ad } \mathbf{Fib} \cap \mathbf{AcCofib} \stackrel{!}{\subseteq} \mathbf{Iso}.$$

Suppose given a fibration $f : G \rightarrow H$ that is an acyclic cofibration.

We consider the following commutative diagram.

$$\begin{array}{ccc} G & \xrightarrow{\text{id}_G} & G \\ f \circ \downarrow & & \downarrow f \\ H & \xrightarrow{\text{id}_H} & H \end{array}$$

We have $\mathbf{AcCofib} = \square \mathbf{Fib}$ and $\mathbf{AcCofib}^\square = \mathbf{Fib}$; cf. Lemmas 191 and 192.

So there exists a graph morphism $h : H \rightarrow G$ such that the following diagram is commutative.

$$\begin{array}{ccc} G & \xrightarrow{\text{id}_G} & G \\ f \circ \downarrow & \nearrow h & \downarrow f \\ H & \xrightarrow{\text{id}_H} & H \end{array}$$

Thus f is an isomorphism. □

Chapter 6

Gph is a model category

In Gph, we have defined subsets $\text{Qis}(\text{Gph})$, $\text{Fib}(\text{Gph})$ and $\text{Cofib}(\text{Gph})$ of $\text{Mor}(\text{Gph})$ in Definitions 115, 127.(1) and 144.

We want to show that the category Gph, together with these subsets, is a model category in the sense of Definition 198 below.

We will use notations introduced in Definitions 5, 9, 14, 13 and 15.

Definition 198 Suppose given a category \mathcal{M} having an initial object \mathfrak{j} and a terminal object $\mathfrak{!}$. Suppose given subsets $\text{Fib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ of *fibrations*, $\text{Cofib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ of *cofibrations* and $\text{Qis}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ of *quasiisomorphisms*. We consider the axioms A_{Fib} , A_{Cofib} , A_{Qis} , A_{Lift} , A_{Fact} , M_{Fib} and M_{Cofib} below.

- (1) The category \mathcal{M} together with $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ is called a *basic model category* if $(A_{\text{Fib}} 1, 2)$, $(A_{\text{Cofib}} 1, 2)$, $(A_{\text{Qis}} 1, 2)$, A_{Lift} , A_{Fact} , $(M_{\text{Fib}} 1, 2)$ and $(M_{\text{Cofib}} 1, 2)$ hold.
- (2) The category \mathcal{M} together with $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ is called a *proper basic model category* if $(A_{\text{Fib}} 1, 2)$, $(A_{\text{Cofib}} 1, 2)$, $(A_{\text{Qis}} 1, 2)$, A_{Lift} , A_{Fact} , M_{Fib} and M_{Cofib} hold.
- (3) The category \mathcal{M} together with $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ is called a *Quillen model category* if M_{PO} , M_{PB} , $(A_{\text{Fib}} 1, 2)$, $(A_{\text{Cofib}} 1, 2)$, $(A_{\text{Qis}} 1, 2)$, A_{Lift} , A_{Fact} , $(M_{\text{Fib}} 1, 2)$ and $M_{\text{Cofib}}(1, 2)$ hold.
- (4) The category \mathcal{M} together with $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ is called a *Quillen closed model category* if M_{PO} , M_{PB} , A_{Fib} , A_{Cofib} , A_{Qis} , A_{Lift} , A_{Fact} , $(M_{\text{Fib}} 1, 2)$ and $(M_{\text{Cofib}} 1, 2)$ hold.

We define the subset $\text{AcFib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ of *acyclic fibrations* to be

$$\text{AcFib}(\mathcal{M}) := \text{Fib}(\mathcal{M}) \cap \text{Qis}(\mathcal{M}).$$

We define the subset $\text{AcCofib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ of *acyclic cofibrations* to be

$$\text{AcCofib}(\mathcal{M}) := \text{Cofib}(\mathcal{M}) \cap \text{Qis}(\mathcal{M}).$$

Suppose given a morphism $f : G \rightarrow H$ in $\text{Mor}(\mathcal{M})$.

To indicate that f is a quasiisomorphism, we often write $G \xrightarrow{\sim} H$.

To indicate that f is a fibration, we often write $G \xrightarrow{f} H$.

To indicate that f is an acyclic fibration, we often write $G \xrightarrow{\dashrightarrow} H$.

To indicate that f is a cofibration, we often write $G \xrightarrow{\bullet} H$.

To indicate that f is an acyclic cofibration, we often write $G \xrightarrow{\dashrightarrow} H$.

Suppose given $G \in \text{Ob}(\mathcal{M})$. Then G is called *fibrant*, if $G \dashrightarrow !$. Moreover G is called *cofibrant*, if $! \dashrightarrow G$.

We consider the following properties.

(M_{PO}) Suppose given the diagram

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow & & \\ X' & & \end{array}$$

in \mathcal{M} . Then there exists a pushout in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

(M_{PB}) Suppose given the diagram

$$\begin{array}{ccc} & & Y \\ & & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

in \mathcal{M} . Then there exists a pullback in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow & \lrcorner & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

- $A_{\text{Fib}} := (A_{\text{Fib}} 1) \wedge (A_{\text{Fib}} 2) \wedge (A_{\text{Fib}} 3)$ where:

(A_{Fib} 1) We have $\text{Iso}(\mathcal{M}) \subseteq \text{Fib}(\mathcal{M})$.

(A_{Fib} 2) Suppose given $X \xrightarrow{f} Y$ and $Y \xrightarrow{g} Z$ in $\text{Mor}(\mathcal{M})$. Then we have $X \xrightarrow{fg} Z$.

(A_{Fib} 3) The subset $\text{Fib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ is closed under retracts; cf. Definition 23.

- $A_{\text{Cofib}} := (A_{\text{Cofib}} 1) \wedge (A_{\text{Cofib}} 2) \wedge (A_{\text{Cofib}} 3)$, where:

(A_{Cofib} 1) We have $\text{Iso}(\mathcal{M}) \subseteq \text{Cofib}(\mathcal{M})$.

(A_{Cofib} 2) Suppose given $X \xrightarrow{\bullet f} Y$ and $Y \xrightarrow{\bullet g} Z$ in $\text{Mor}(\mathcal{M})$. Then we have $X \xrightarrow{\bullet fg} Z$.

(A_{Cofib} 3) The subset $\text{Cofib}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ is closed under retracts; cf. Definition 23.

- A_{Qis} := (A_{Qis} 1) \wedge (A_{Qis} 2) \wedge (A_{Qis} 3), where:

(A_{Qis} 1) We have $\text{Iso}(\mathcal{M}) \subseteq \text{Qis}(\mathcal{M})$.

(A_{Qis} 2) The set $\text{Qis}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ satisfies (2 of 3).

(A_{Qis} 3) The subset $\text{Qis}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ is closed under retracts; cf. Definition 23.

- A_{Lift} := (A_{Lift} 1) \wedge (A_{Lift} 2), where:

(A_{Lift} 1) We have $\text{AcCofib}(\mathcal{M}) \boxtimes \text{Fib}(\mathcal{M})$; cf. Definition 15.

(A_{Lift} 2) We have $\text{Cofib}(\mathcal{M}) \boxtimes \text{AcFib}(\mathcal{M})$; cf. Definition 15.

- $\hat{\text{A}}_{\text{Lift}}$:= ($\hat{\text{A}}_{\text{Lift}}$ 1) \wedge ($\hat{\text{A}}_{\text{Lift}}$ 2), where:

($\hat{\text{A}}_{\text{Lift}}$ 1) We have $\text{AcCofib}(\mathcal{M}) \hat{\boxtimes} \text{Fib}(\mathcal{M})$; cf. Definition 27.

($\hat{\text{A}}_{\text{Lift}}$ 2) We have $\text{Cofib}(\mathcal{M}) \hat{\boxtimes} \text{AcFib}(\mathcal{M})$; cf. Definition 27.

- A_{Fact} := (A_{Fact} 1) \wedge (A_{Fact} 2), where:

(A_{Fact} 1) For $X \xrightarrow{f} Y$ in \mathcal{M} there exists a commutative diagram in \mathcal{M} as follows.

$$\begin{array}{ccc} & U & \\ g \nearrow & & \searrow h \\ X & \xrightarrow{f} & Y \end{array}$$

(A_{Fact} 2) For $X \xrightarrow{f} Y$ in \mathcal{M} there exists a commutative diagram in \mathcal{M} as follows.

$$\begin{array}{ccc} & U & \\ g \nearrow \bullet & & \searrow \bullet h \\ X & \xrightarrow{f} & Y \end{array}$$

- M_{Fib} := (M_{Fib} 1) \wedge (M_{Fib} 2) \wedge (M_{Fib} 3), where:

(M_{Fib} 1) Suppose given the diagram

$$\begin{array}{ccc} & Y & \\ & \downarrow g & \\ X' & \xrightarrow{u'} & Y' \end{array}$$

in \mathcal{M} . Then there exists a pullback in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow f & \lrcorner & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

(M_{Fib} 2) Suppose given the diagram

$$\begin{array}{ccc} & & Y \\ & & \downarrow \dashv g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

in \mathcal{M} . Then there exists a pullback in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow \dashv f & \lrcorner & \downarrow \dashv g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

(M_{Fib} 3) Suppose given the diagram

$$\begin{array}{ccc} & & Y \\ & & \downarrow \dashv g \\ X' & \xrightarrow{\approx} & Y' \end{array}$$

in \mathcal{M} . Then there exists a pullback in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{\approx} & Y \\ \downarrow \dashv f & \lrcorner & \downarrow \dashv g \\ X' & \xrightarrow{\approx} & Y' \end{array}$$

- M_{Cofib} := (M_{Cofib} 1) \wedge (M_{Cofib} 2) \wedge (M_{Cofib} 3), where:

(M_{Cofib} 1) Suppose given the diagram

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow \bullet f & & \\ & & X' \end{array}$$

in \mathcal{M} . Then there exists a pushout in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow \bullet f & & \downarrow \bullet g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

(M_{Cofib} 2) Suppose given the diagram

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow \circ f & & \\ & & X' \end{array}$$

in \mathcal{M} . Then there exists a pushout in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow f \circ & & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array} \quad \lrcorner$$

(M_{Cofib} 3) Suppose given the diagram

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow f \bullet & & \\ X' & & \end{array}$$

in \mathcal{M} . Then there exists a pushout in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ \downarrow f \bullet & & \downarrow g \bullet \\ X' & \xrightarrow{u'} & Y' \end{array} \quad \lrcorner$$

Remark 199

- (1) We have $\hat{A}_{\text{Lift}} \wedge (A_{\text{Qis}} 3)$ if and only if $A_{\text{Lift}} \wedge (A_{\text{Fib}} 3) \wedge (A_{\text{Cofib}} 3) \wedge (A_{\text{Qis}} 3) \wedge A_{\text{Fact}}$.
- (2) The category \mathcal{M} together with $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ is a Quillen closed model category if and only if M_{PO} , M_{PB} , $(A_{\text{Fib}} 1, 2)$, $(A_{\text{Cofib}} 1, 2)$, A_{Qis} , \hat{A}_{Lift} , $(M_{\text{Fib}} 1, 2)$ and $(M_{\text{Cofib}} 1, 2)$ hold.

Proof.

Ad (1).

Ad \Rightarrow .

We have $\text{AcCofib}(\mathcal{M}) \hat{\boxtimes} \text{Fib}(\mathcal{M})$ and $\text{Cofib}(\mathcal{M}) \hat{\boxtimes} \text{AcFib}(\mathcal{M})$ and the subset $\text{Qis}(\mathcal{M}) \subseteq \text{Mor}(\mathcal{M})$ is closed under retracts.

Because of (C 3) we have A_{Fact} ; cf. Definition 27.

Since $\text{AcCofib}(\mathcal{M}) \hat{\boxtimes} \text{Fib}(\mathcal{M})$, we have $\text{AcCofib}(\mathcal{M}) \boxtimes \text{Fib}(\mathcal{M})$ and $(A_{\text{Fib}} 3)$; cf. Remark 28.

Since $\text{Cofib}(\mathcal{M}) \hat{\boxtimes} \text{AcFib}(\mathcal{M})$, we have $\text{Cofib}(\mathcal{M}) \boxtimes \text{AcFib}(\mathcal{M})$ and $(A_{\text{Cofib}} 3)$; cf. Remark 28.

So because of $\text{AcCofib}(\mathcal{M}) \boxtimes \text{Fib}(\mathcal{M})$ and $\text{Cofib}(\mathcal{M}) \boxtimes \text{AcFib}(\mathcal{M})$ we have A_{Lift} .

Ad \Leftarrow .

The subsets $\text{Fib}(\mathcal{M})$, $\text{Cofib}(\mathcal{M})$ and $\text{Qis}(\mathcal{M})$ in $\text{Mor}(\mathcal{M})$ are closed under retracts by $(A_{\text{Fib}} 3)$, $(A_{\text{Cofib}} 3)$ and $(A_{\text{Qis}} 3)$.

Hence the subsets $\text{AcFib}(\mathcal{M})$ and $\text{AcCofib}(\mathcal{M})$ in $\text{Mor}(\mathcal{M})$ are closed under retracts.

Because of $(A_{\text{Fact}} 1)$, $(A_{\text{Lift}} 1)$, $(A_{\text{Fib}} 3)$ and $\text{AcCofib}(\mathcal{M})$ being closed under retracts we have $(\hat{A}_{\text{Lift}} 1)$ by Remark 28.

Because of $(A_{\text{Fact}} 2)$, $(A_{\text{Lift}} 2)$, $\text{AcFib}(\mathcal{M})$ being closed under retracts and $(A_{\text{Cofib}} 3)$ we have $(\hat{A}_{\text{Lift}} 2)$ by Remark 28. \square

Ad (2).

This follows using (1). \square

Reminder 200 We recall the meaning of $\text{AcCofib}(\mathcal{M}) \boxtimes \text{Fib}(\mathcal{M})$; cf. A_{Lift} , Definition 15.

Suppose given an acyclic cofibration $X \xrightarrow{f} X'$ in $\text{AcCofib}(\mathcal{M})$ and a fibration $Y \xrightarrow{g} Y'$ in $\text{Fib}(\mathcal{M})$. Suppose given a commutative diagram in \mathcal{M} as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow & & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

Then there exists a $h : X' \rightarrow Y$ in $\text{Mor}(\mathcal{M})$ such that $fh = u$ and such that $hg = u'$, i.e. we have the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow & \exists h \nearrow & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

The morphism h is also called a *lift* of this quadrangle.

Reminder 201 We recall the meaning of $\text{Cofib}(\mathcal{M}) \boxtimes \text{AcFib}(\mathcal{M})$; cf. A_{Lift} , Definition 15.

Suppose given a cofibration $X \xrightarrow{f} X'$ in $\text{Cofib}(\mathcal{M})$ and an acyclic fibration $Y \xrightarrow{g} Y'$ in $\text{AcFib}(\mathcal{M})$. Suppose given a commutative diagram as follows.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow \bullet & & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

Then there exists $h : X' \rightarrow Y$ in $\text{Mor}(\mathcal{M})$ such that $fh = u$ and such that $hg = u'$, i.e. we have the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{u} & Y \\ f \downarrow \bullet & \exists h \nearrow & \downarrow g \\ X' & \xrightarrow{u'} & Y' \end{array}$$

The morphism h is also called a *lift* of this quadrangle.

Remark 202

- (1) D. Quillen introduced Quillen model categories under the name *model categories* in [4, §I.1].
He introduced Quillen closed model categories under the name *closed model categories* in [4, §I.5]; cf. [4, §I.5, Prop. 2].
- (2) M. Ritter used basic proper model categories under the name *model categories* in [6, Def. 172].
- (3) The conditions $(M_{\text{Fib}} 3)$ and $(M_{\text{Cofib}} 3)$ were introduced in [5, §1, Def. 1.2].

Reminder 203 We defined the subsets $\text{Fib}(\text{Gph}), \text{Qis}(\text{Gph}) \subseteq \text{Mor}(\text{Gph})$ directly; cf. Definitions 127 and 115.

We defined $\text{AcFib}(\text{Gph}) = \text{Fib}(\text{Gph}) \cap \text{Qis}(\text{Gph})$; cf. Definition 138.

We defined $\text{Cofib}(\text{Gph}) = \square \text{AcFib}(\text{Gph})$; cf. Definition 144.

The following proposition is due to BISSON and TSEMO [3, Cor. 4.8].

Proposition 204 The category Gph together with $\text{Fib}(\text{Gph}), \text{Cofib}(\text{Gph})$ and $\text{Qis}(\text{Gph})$, is a Quillen closed model category; cf. Definition 198.(4), Reminder 203.

In addition, Gph satisfies $(M_{\text{Fib}} 3)$.

Moreover, for $\text{AcCofib}(\text{Gph})$, as defined by $(\text{AcCofib} 1\text{--}5)$ in Definition 162, we have the equality $\text{AcCofib}(\text{Gph}) = \text{Cofib}(\text{Gph}) \cap \text{Qis}(\text{Gph})$.

Finally, note that Gph satisfies \hat{A}_{Lift} ; cf. Remark 199.

Proof. We have $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$; cf. Lemma 185.

We have M_{PO} ; cf. Construction 83.

We have M_{PB} ; cf. Construction 97.

We have $(A_{\text{Fib}} 1)$; cf. Remark 129.

We have $(A_{\text{Fib}} 2)$; cf. Remark 131.

We have $(A_{\text{Fib}} 3)$; cf. Remark 130.

We have $(A_{\text{Cofib}} 1)$; cf. Remark 145.

We have $(A_{\text{Cofib}} 2)$; cf. Remark 147.

We have $(A_{\text{Cofib}} 3)$; cf. Remark 146.

We have $(A_{\text{Qis}} 1)$; cf. Remark 116.

We have $(A_{\text{Qis}} 2)$; cf. Remark 123.

We have $(A_{\text{Qis}} 3)$; cf. Remark 117.

We have $(A_{\text{Lift}} 1)$, i.e. $\text{AcCofib}(\text{Gph}) \square \text{Fib}(\text{Gph})$; cf. Remark 174.

We have $(A_{\text{Lift}} 2)$, i.e. $\text{Cofib}(\text{Gph}) \square \text{AcFib}(\text{Gph})$; cf. Definition 144.

We have (A_{Fact} 1); cf. Lemma 184.

We have (A_{Fact} 2); cf. Lemma 188.

We have (M_{Fib} 1); cf. Remark 133.

We have (M_{Fib} 2); cf. Remark 143.

We have (M_{Fib} 3); cf. Remarks 125 and 133.

We have (M_{Cofib} 1); cf. Remark 148.

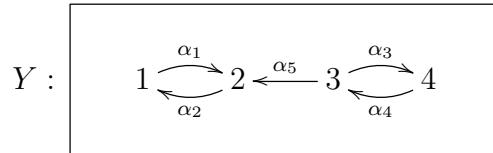
We have (M_{Cofib} 2); cf. Remark 171. □

Remark 205 In Gph, the axiom (M_{Cofib} 3) does **not** hold.

In other words, there exists a quasiisomorphism in Gph, whose pushout along a cofibration is not a quasiisomorphism.

Counterexample.

Consider the following thin graph.



We consider the cofibration $d_{C_2} : C_2 \sqcup C_2 \xrightarrow{\bullet} C_2$; cf. Definition 154.

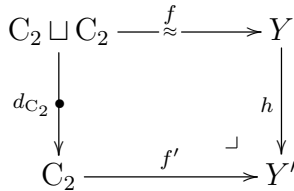
We consider the quasiisomorphism $f : C_2 \sqcup C_2 \rightarrow Y$ with

$$\begin{aligned} (1, v_0) V_f &:= 1, & (1, v_1) V_f &:= 2, \\ (2, v_0) V_f &:= 3, & (2, v_1) V_f &:= 4 \end{aligned}$$

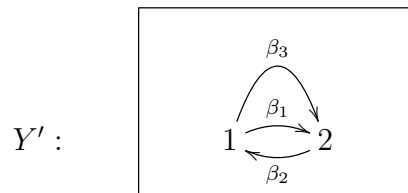
and with

$$\begin{aligned} (1, e_0) E_f &:= \alpha_1, & (1, e_1) E_f &:= \alpha_2, \\ (2, e_0) E_f &:= \alpha_3, & (2, e_1) E_f &:= \alpha_4. \end{aligned}$$

Constructing the pushout using Magma, we obtain



with



$$1 V_h = 1, \quad 2 V_h = 2, \quad 3 V_h = 1, \quad 4 V_h = 2,$$

$$\begin{aligned}\alpha_1 E_h &= \beta_1, & \alpha_2 E_h &= \beta_2, & \alpha_3 E_h &= \beta_1, \\ \alpha_4 E_h &= \beta_2, & \alpha_5 E_h &= \beta_3\end{aligned}$$

and

$$\begin{aligned}v_0 V_{f'} &= 1, & e_0 E_{f'} &= \beta_1, \\ v_1 V_{f'} &= 2, & e_1 E_{f'} &= \beta_2.\end{aligned}$$

But the resulting graph morphism $f' : C_2 \rightarrow Y'$ is **not** a quasiisomorphism because $|(C_2, C_2)_{\text{Gph}}| = 2 \neq 4 = |(C_2, Y')_{\text{Gph}}|$.

Via Magma [2] we proceed as follows, using the functions given in §10 below.

```
C2C2 := DCN(2)[1];
C2 := DCN(2)[2];
d := DCN(2)[3];
Y := <[1,2,3,4], [<1,1,2>, <2,2,1>, <3,3,4>, <4,4,3>, <3,5,2>]>;
f := IsSubgraph(C2C2, Y)[1];
PO := PushoutGraphs(C2C2, Y, C2, f, d);
Yp := PO[1]; // "Y prime" = Y'
fp := PO[2]; // "f prime" = f'
h := PO[3];
> C2;
<[ 1, 2 ], [ <1, 1, 2>, <2, 2, 1> ]>
> C2C2;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 1>, <3, 3, 4>, <4, 4, 3> ]>
> d;
<[ <1, 1>, <2, 2>, <3, 1>, <4, 2> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 2, 1>>, <<3, 3, 4>, <1, 1, 2>>, <<4, 4, 3>, <2, 2, 1>> ]>
> Y;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 1>, <3, 3, 4>, <4, 4, 3>, <3, 5, 2> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 2, 1>>, <<3, 3, 4>, <3, 3, 4>>, <<4, 4, 3>, <4, 4, 3>> ]>
> Yp;
<[ 1, 2 ], [ <1, 1, 2>, <2, 2, 1>, <1, 3, 2> ]>
> fp;
<[ <1, 1>, <2, 2> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>> ]>
> h;
<[ <1, 1>, <2, 2>, <3, 1>, <4, 2> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 2, 1>>, <<3, 3, 4>, <1, 1, 2>>, <<4, 4, 3>, <2, 2, 1>>,
  <<3, 5, 2>, <1, 3, 2>> ]>
> IsQis_Bound(fp, C2, Yp, 2);
false
> #ListOfnCycles(C2, 2);
2
> #ListOfnCycles(Yp, 2);
4
```

Note that the calculated graph morphism $h : Y \rightarrow Y'$ is a cofibration since cofibrations are stable under pushouts; cf. Remark 148.

In the similar situation of the pointed category of graphs, D. Vicinsky has constructed an example that shows that quasiisomorphisms are not stable under pushouts along cofibrations; cf. [7, Prop. 5.8].

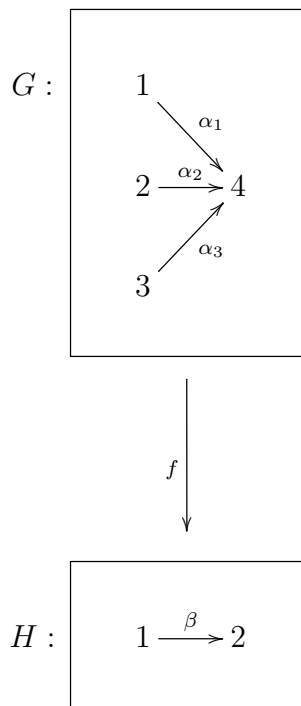
Chapter 7

A sufficient condition for a graph morphism to be a quasiisomorphism

Definition 206 Suppose given a graph morphism $f : G \rightarrow H$. An edge $e_H \in E_H$ is called *unitargeting* with respect to f , if

$$|\{\tilde{e} t_G : \tilde{e} \in E_G, \tilde{e} E_f = e_H\}| = 1.$$

Example 207 We consider the following graph morphism.



Here, $f : G \rightarrow H$ is the graph morphism mapping the vertices and the edges in a vertical way.

Then the edge $\beta \in E_H$ is unitargeting with respect to f , since we have

$$|\{\tilde{e} t_G : \tilde{e} \in E_G, \tilde{e} E_f = \beta\}| = |\{\alpha_1 t_G, \alpha_2 t_G, \alpha_3 t_G\}| = |\{4\}| = 1.$$

Lemma 208 Suppose given graphs G and H and an etale fibration $f : G \rightarrow H$; cf. Definition 127.(2).

Recall that f being an etale fibration means that the map

$$E_{f,v} : G(v, *) \rightarrow H(v V_f, *)$$

is bijective for $v \in V_G$.

Suppose given a commutative diagram in Gph as follows, where $\iota_n = \iota_{0,n} : D_0 \rightarrow D_n$; cf. Definition 56.(3).

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_n \downarrow & & \downarrow f \\ D_n & \xrightarrow{y} & H \end{array}$$

Then there exists a unique graph morphism $\hat{y} : D_n \rightarrow G$ such that the following diagram is commutative.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_n \downarrow & \nearrow \hat{y} & \downarrow f \\ D_n & \xrightarrow{y} & H \end{array}$$

Proof. We have to show that there exists a graph morphism $\hat{y} : D_n \rightarrow G$ such that $\iota_n \hat{y} \stackrel{!}{=} a$ and such that $\hat{y} f \stackrel{!}{=} y$.

Existence.

We *claim* that for $k \in [0, n]$ there exists $\hat{y}_k : D_k \rightarrow G$ such that $\iota_{0,k} \hat{y}_k = a$ and $\hat{y}_k f = \iota_{k,n} y$; cf. Definition 56.(3).

We carry out an *induction* for $k \in [0, n]$:

If $k = 0$, we let $\hat{y}_0 := a : D_0 \rightarrow G$. We have $\iota_{0,0} \hat{y}_0 = \text{id}_{D_0} a = a$ and $\hat{y}_0 f = \iota_{0,n} y$.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \text{id}_{D_0} \downarrow & \nearrow \hat{y}_0 = a & \downarrow f \\ D_0 & \xrightarrow{\iota_{0,n} y} & H \end{array}$$

Induction step: Suppose given $k \in [0, n - 1]$ and $\hat{y}_k : D_k \rightarrow G$ such that $\iota_{0,k} \hat{y}_k = a$ and $\hat{y}_k f = \iota_{k,n} y$.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_{0,k} \downarrow & \nearrow \hat{y}_k & \downarrow f \\ D_k & & \\ \iota_{k,k+1} \downarrow & & \\ D_{k+1} & \xrightarrow{\iota_{k+1,n} y} & H \end{array}$$

We show that there exists $\hat{y}_{k+1} : D_{k+1} \rightarrow G$ such that $\iota_{0,k+1}\hat{y}_{k+1} = a$ and $\hat{y}_{k+1}f = \iota_{k+1,n}y$.

We have $\hat{v}_k V_{\hat{y}_k} \in V_G$ and $\hat{v}_k V_{\hat{y}_k} V_f = \hat{v}_k V_y \in V_H$.

Since the map $E_{f, \hat{v}_k V_{\hat{y}_k}} : G(\hat{v}_k V_{\hat{y}_k}, *) \rightarrow H(\hat{v}_k V_y, *)$ is bijective and since $\hat{e}_k E_y s_H = \hat{e}_k s_{D_n} V_y = \hat{v}_k V_y$, there exists a unique edge $\tilde{e} \in G(\hat{v}_k V_{\hat{y}_k}, *) \subseteq E_G$ such that $\tilde{e} E_f = \hat{e}_k E_y \in H(\hat{v}_k V_y, *)$.

So let $\hat{y}_{k+1} : D_{k+1} \rightarrow G$ with $\hat{v}_i V_{\hat{y}_{k+1}} = \hat{v}_i V_{\hat{y}_k}$ for $i \in [0, k]$ and $\hat{v}_{k+1} V_{\hat{y}_{k+1}} = \tilde{e} t_G$, as well as $\hat{e}_i E_{\hat{y}_{k+1}} := \hat{e}_i E_{\hat{y}_k}$ for $i \in [0, k-1]$ and $\hat{e}_k E_{\hat{y}_{k+1}} := \tilde{e}$.

We have to show that \hat{y}_{k+1} is in fact a graph morphism.

We have to show that $\hat{e}_i E_{\hat{y}_{k+1}} s_G \stackrel{!}{=} \hat{e}_i s_{D_{k+1}} V_{\hat{y}_{k+1}}$ and that $\hat{e}_i E_{\hat{y}_{k+1}} t_G \stackrel{!}{=} \hat{e}_i t_{D_{k+1}} V_{\hat{y}_{k+1}}$ in V_G for $i \in [0, k]$.

If $i \in [0, k-1]$, then we have $\hat{e}_i E_{\hat{y}_{k+1}} s_G = \hat{e}_i E_{\hat{y}_k} s_G = \hat{e}_i s_{D_k} V_{\hat{y}_k} = \hat{v}_i V_{\hat{y}_k} = \hat{v}_i V_{\hat{y}_{k+1}} = (\hat{e}_i s_{D_{k+1}}) V_{\hat{y}_{k+1}}$.

If $i \in [0, k-1]$, then we have $\hat{e}_i E_{\hat{y}_{k+1}} t_G = \hat{e}_i E_{\hat{y}_k} t_G = \hat{e}_i t_{D_k} V_{\hat{y}_k} = \hat{v}_{i+1} V_{\hat{y}_k} = \hat{v}_{i+1} V_{\hat{y}_{k+1}} = (\hat{e}_i t_{D_{k+1}}) V_{\hat{y}_{k+1}}$.

If $i = k$, then we have $\hat{e}_k E_{\hat{y}_{k+1}} s_G = \tilde{e} s_G = \hat{v}_k V_{\hat{y}_k} = \hat{v}_k V_{\hat{y}_{k+1}} = \hat{e}_k s_{D_{k+1}} V_{\hat{y}_{k+1}}$.

If $i = k$, then we have $\hat{e}_k E_{\hat{y}_{k+1}} t_G = \tilde{e} t_G = \hat{v}_{k+1} V_{\hat{y}_{k+1}} = (\hat{e}_k t_{D_{k+1}}) V_{\hat{y}_{k+1}}$.

So \hat{y}_{k+1} is in fact a graph morphism.

We have to show that $\iota_{0,k+1}\hat{y}_{k+1} \stackrel{!}{=} a$.

In fact, we have $\hat{v}_0 V_{\iota_{0,k+1}} V_{\hat{y}_{k+1}} = \hat{v}_0 V_{\hat{y}_{k+1}} = \hat{v}_0 V_{\hat{y}_k} = \hat{v}_0 V_{\iota_{0,k}} V_{\hat{y}_k} = \hat{v}_0 V_a$.

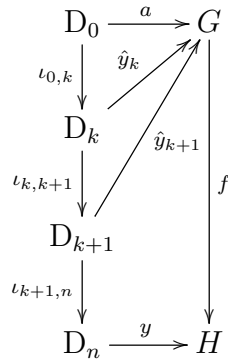
We have to show that $\hat{y}_{k+1}f \stackrel{!}{=} \iota_{k+1,n}y$.

Since the direct graph D_{k+1} is thin, it suffices to show that $V_{\hat{y}_{k+1}f} \stackrel{!}{=} V_{\iota_{k+1,n}y}$.

Suppose given $i \in [0, k+1]$.

If $i \in [0, k]$, then we have $\hat{v}_i V_{\hat{y}_{k+1}f} = \hat{v}_i V_{\hat{y}_{k+1}} V_f = \hat{v}_i V_{\hat{y}_k} V_f = \hat{v}_i V_{\iota_{k,n}y} = \hat{v}_i V_{\iota_{k,n}} V_y = \hat{v}_i V_y = \hat{v}_i V_{\iota_{k+1,n}y}$.

If $i = k+1$ then we have $\hat{v}_{k+1} V_{\hat{y}_{k+1}f} = \hat{v}_{k+1} V_{\hat{y}_{k+1}} V_f = \tilde{e} t_G V_f = \tilde{e} E_f t_H = \hat{e}_k E_y t_H = \hat{e}_k t_{D_n} V_y = \hat{v}_{k+1} V_y = \hat{v}_{k+1} V_{\iota_{k+1,n}y}$.



So we have shown in an inductive way, that there exists a graph morphism $\hat{y} := \hat{y}_n : D_n \rightarrow G$ such that $\iota_{0,n}\hat{y} = a$ and $\hat{y}f = y$.

Uniqueness. Suppose given commutative diagrams as follows.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_{0,n} \downarrow & \hat{y} \nearrow & \downarrow f \\ D_n & \xrightarrow{y} & H \end{array} \qquad \begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_{0,n} \downarrow & \hat{y}' \nearrow & \downarrow f \\ D_n & \xrightarrow{y} & H \end{array}$$

We have to show that $\hat{y} \stackrel{!}{=} \hat{y}'$.

Since the direct graph D_n is thin, it suffices to show that $V_{\hat{y}} \stackrel{!}{=} V_{\hat{y}'}$.

We want to show that $\hat{v}_k V_{\hat{y}} \stackrel{!}{=} \hat{v}_k V_{\hat{y}'}$ for $k \in [0, n]$.

We proceed by induction on k .

If $k = 0$ then we have $\hat{v}_0 V_{\hat{y}} \stackrel{\text{Def.}}{=} \iota_{0,n} \hat{v}_0 V_{\iota_{0,n} V_{\hat{y}}} \stackrel{\iota_{0,n} \hat{y} = a}{=} \hat{v}_0 V_a \stackrel{\iota_{0,n} \hat{y}' = a}{=} \hat{v}_0 V_{\iota_{0,n} V_{\hat{y}'}} \stackrel{\text{Def.}}{=} \iota_{0,n} \hat{v}_0 V_{\hat{y}'}$.

Induction step: Let $k \in [0, n - 1]$. We suppose that $\hat{v}_k V_{\hat{y}} = \hat{v}_k V_{\hat{y}'}$.

We have to show that $\hat{v}_{k+1} V_{\hat{y}} \stackrel{!}{=} \hat{v}_{k+1} V_{\hat{y}'}$.

Let $v_G := \hat{v}_k V_{\hat{y}} = \hat{v}_k V_{\hat{y}'}$. Since the graph morphism $f : G \rightarrow H$ is an etale fibration, the map $E_{f,v_G} : G(v_G, *) \rightarrow H(v_G V_f, *)$ is bijective.

We have $v_G V_f = \hat{v}_k V_{\hat{y}} V_f = \hat{v}_k V_y$.

We have $\hat{e}_k E_y \in H(v_G V_f, *)$ since $(\hat{e}_k E_y) s_H = \hat{e}_k s_{D_n} V_y = \hat{v}_k V_y = v_G V_f$.

We have $\hat{e}_k E_{\hat{y}} \in G(v_G, *)$ because $(\hat{e}_k E_{\hat{y}}) s_G = \hat{e}_k s_{D_n} V_{\hat{y}} = \hat{v}_k V_{\hat{y}} = v_G$.

Moreover, $(\hat{e}_k E_{\hat{y}}) E_{f,v_G} = \hat{e}_k E_{\hat{y}} E_f = (\hat{e}_k E_y)$.

Likewise, we have $\hat{e}_k E_{\hat{y}'} \in G(v_G, *)$ and $(\hat{e}_k E_{\hat{y}'}) E_{f,v_G} = (\hat{e}_k E_y)$.

Since the map E_{f,v_G} is bijective, we have $\hat{e}_k E_{\hat{y}} = \hat{e}_k E_{\hat{y}'}$ and so $\hat{v}_{k+1} V_{\hat{y}} = \hat{e}_k t_{D_n} V_{\hat{y}} = \hat{e}_k E_{\hat{y}} t_G = \hat{e}_k E_{\hat{y}'} t_G = \hat{e}_k t_G V_{\hat{y}'} = \hat{v}_{k+1} V_{\hat{y}'}$. \square

Lemma 209 Suppose given graphs G and H and an etale fibration $f : G \rightarrow H$; cf. Definition 127.(2).

Recall that f being an etale fibration means that the map

$$E_{f,v} := E_f \Big|_{G(v,*)}^{H(v V_f, *)} : G(v, *) \rightarrow H(v V_f, *)$$

is bijective for $v \in V_G$.

Suppose given graph morphisms $\hat{u} : C_n \rightarrow G$ and $\hat{u}' : C_n \rightarrow G$ such that $\hat{u}f = \hat{u}'f$ and $v_0 V_{\hat{u}} = v_0 V_{\hat{u}'}$.

Then $\hat{u} = \hat{u}'$.

Proof. Let $u := \hat{u}f = \hat{u}'f : C_n \rightarrow H$.

Consider the graph morphism $a : D_0 \rightarrow G$ with $\hat{v}_0 V_a = v_0 V_{\hat{u}} = v_0 V_{\hat{u}'}$, the graph morphism $r_n : D_n \rightarrow C_n$ with $\hat{v}_i V_r = v_i$ for $i \in [0, n]$ and the graph morphism $\iota_n = \iota_{0,n} : D_0 \rightarrow D_n$; cf.

Definition 56.(3). We have the following commutative diagrams.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \downarrow \iota_n & \nearrow \hat{u} & \downarrow f \\ D_n & \xrightarrow{r_n} C_n \xrightarrow{u} & H \end{array}$$

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \downarrow \iota_n & \nearrow \hat{u}' & \downarrow f \\ D_n & \xrightarrow{r_n} C_n \xrightarrow{u} & H \end{array}$$

Then we have $(r_n \hat{u})f = r_n u$ and $(r_n \hat{u}')f = r_n u$, and we have $\iota_n(r_n \hat{u}) = a$ and $\iota_n(r_n \hat{u}') = a$.

Because of Lemma 208 there exists a unique graph morphism $\hat{y} : D_n \rightarrow G$ such that $\iota_n \hat{y} = a$ and $\hat{y}f = r_n u$. Therefore, we have $r_n \hat{u} = \hat{y} = r_n \hat{u}'$.

Since r_n is surjective, we conclude that $\hat{u} = \hat{u}'$; cf. Remark 72. \square

Proposition 210 Suppose given graphs G and H .

Suppose given an etale fibration $f : G \rightarrow H$ that satisfies the following property (Uni).

(Uni) For $n \geq 1$ and each graph morphism $u : C_n \rightarrow H$, there exists $i \in \mathbb{Z}/n\mathbb{Z}$ such that $e_i E_u \in E_H$ is unitargeting with respect to f .

Then the graph morphism $f : G \rightarrow H$ is a quasiisomorphism; cf. Definition 115.

Note that altogether, f is an acyclic fibration; cf. Definition 138.

Proof. Suppose given $n \geq 1$. Suppose given a graph morphism $u_0 : C_n \rightarrow H$.

We have to show that there exists a unique graph morphism $\hat{u}_0 : C_n \rightarrow G$ such that $\hat{u}_0 f = u_0$.

Since (Uni) is satisfied, we may choose $i \in \mathbb{Z}/n\mathbb{Z}$ such that $e_i E_{u_0} \in E_H$ is unitargeting.

Let $u := a_{i+1} u_0$; cf. Definition 67. Then $u_0 = a_{-i-1} u$.

So the edge $e_i E_{u_0} = e_i a_{-i-1} E_u = e_{-1} E_u \in E_H$ is unitargeting.

We have

$$\begin{aligned} & \{\hat{u}_0 \in (C_n, G) : \hat{u}_0 f = u_0\} \\ &= \{a_{-i-1} \hat{u} : \hat{u} \in (C_n, G), a_{-i-1} \hat{u} f = u_0 = a_{-i-1} u\} \\ &= \{a_{-i-1} \hat{u} : \hat{u} \in (C_n, G)_{\text{Gph}}, \hat{u} f = u\} \\ &= (\{\hat{u} \in (C_n, G) : \hat{u} f = u\})(a_{-i-1}, G). \end{aligned}$$

So we have to show that there exists a unique graph morphism $\hat{u} : C_n \rightarrow G$ such that $\hat{u} f = u$.

$$\begin{array}{ccc} & & G \\ & \nearrow \hat{u} & \downarrow f \\ C_n & \xrightarrow{\hat{u}} & H \end{array}$$

Uniqueness.

Suppose given graph morphisms $\hat{u} : C_n \rightarrow G$ and $\hat{u}' : C_n \rightarrow G$ such that $\hat{u}f = u = \hat{u}'f$.

We have to show that $\hat{u} \stackrel{!}{=} \hat{u}'$.

We have $(e_{-1} E_{\hat{u}}) E_f = e_{-1} E_u = (e_{-1} E_{\hat{u}'}) E_f$.

Since $e_{-1} E_u$ is unitargeting, we have $e_{-1} E_{\hat{u}} t_G = e_{-1} E_{\hat{u}'} t_G$.

So we have $v_0 V_{\hat{u}} = e_{-1} t_{C_n} V_{\hat{u}} = e_{-1} E_{\hat{u}} t_G = e_{-1} E_{\hat{u}'} t_G = e_{-1} t_{C_n} V_{\hat{u}'} = v_0 V_{\hat{u}'}$.

By Lemma 209 we obtain $\hat{u} = \hat{u}'$.

Existence.

Let $\{v_G\} := \{\hat{e} t_G : \hat{e} \in E_G, \hat{e} E_f = e_{-1} E_u\}$, using that $e_{-1} E_u$ is unitargeting.

Consider the graph morphism $a : D_0 \rightarrow G$ with $\hat{v}_0 V_a := v_G$, the graph morphism $r_n : D_n \rightarrow C_n$ with $\hat{v}_i V_r = v_i$ for $i \in [0, n]$ and the graph morphism $\iota_n = \iota_{0,n} : D_0 \rightarrow D_n$; cf. Definition 56.(3). We have the following commutative diagram.

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_n \downarrow & & \downarrow f \\ D_n & \xrightarrow{r_n} C_n \xrightarrow{u} & H \end{array}$$

We show that it commutes, i.e. that $af \stackrel{!}{=} \iota_n r_n u$.

We choose $\hat{e} \in E_G$ with $\hat{e} E_f = e_{-1} E_u$. Then $\hat{v}_0 V_{af} = v_G V_f = \hat{e} t_G V_f = \hat{e} E_f t_H = e_{-1} E_u t_H = e_{-1} t_{C_n} V_u = v_0 V_u = \hat{v}_0 V_{ru} = \hat{v}_0 V_{\iota_n r_n u}$.

So the diagram is in fact commutative.

By Lemma 208 there exists a graph morphism $\hat{y} : D_n \rightarrow G$ such that

$$\begin{array}{ccc} D_0 & \xrightarrow{a} & G \\ \iota_n \downarrow & \nearrow \hat{y} & \downarrow f \\ D_n & \xrightarrow{r_n} C_n \xrightarrow{u} & H \end{array}$$

commutes.

Now we show that there exists a graph morphism $\hat{u} : C_n \rightarrow G$ such that $r_n \hat{u} \stackrel{!}{=} \hat{y}$.

Therefor we have to show that $\hat{v}_n V_{\hat{y}} \stackrel{!}{=} \hat{v}_0 V_{\hat{y}}$.

We have $\hat{e}_{n-1} E_{\hat{y}} \in E_G$.

We have $(\hat{e}_{n-1} E_{\hat{y}}) E_f = \hat{e}_{n-1} E_{r_n} E_u = e_{n-1} E_u = e_{-1} E_u$.

So we have $\hat{e}_{n-1} E_{\hat{y}} t_G = v_G$.

So we have $\hat{v}_n V_{\hat{y}} = \hat{e}_{n-1} t_{D_n} V_{\hat{y}} = \hat{e}_{n-1} E_{\hat{y}} t_G = v_G = \hat{v}_0 V_a = \hat{v}_0 V_{\iota_n} V_{\hat{y}} = \hat{v}_0 V_{\hat{y}}$.

So we have $\hat{v}_n V_{\hat{y}} = \hat{v}_0 V_{\hat{y}}$.

$$\begin{array}{ccc}
 D_0 & \xrightarrow{a} & G \\
 \downarrow \iota_n & \nearrow \hat{y} & \downarrow f \\
 D_n & \xrightarrow{r_n} C_n \xrightarrow{u} & H
 \end{array}$$

Now we have $r_n \hat{u} f = \hat{y} f = r_n u$.

Since r_n is surjective, we conclude that $\hat{u} f = u$; cf. Remark 72. □

Chapter 8

Duality

Definition 211

(1) Suppose given a graph G .

We define the *opposite* graph $G^{\text{op}} := (V_{G^{\text{op}}}, E_{G^{\text{op}}}, s_{G^{\text{op}}}, t_{G^{\text{op}}})$ by $V_{G^{\text{op}}} := V_G$, $E_{G^{\text{op}}} := E_G$, $s_{G^{\text{op}}} := t_G$ and $t_{G^{\text{op}}} := s_G$.

(2) Suppose given a graph morphism $f : G \rightarrow H$.

We define the *opposite* graph morphism $f^{\text{op}} = (V_{f^{\text{op}}}, E_{f^{\text{op}}}) : G^{\text{op}} \rightarrow H^{\text{op}}$ with $V_{f^{\text{op}}} := V_f : V_G \rightarrow V_H$ and $E_{f^{\text{op}}} := E_f : E_G \rightarrow E_H$.

In fact, we have $E_{f^{\text{op}}} s_{H^{\text{op}}} = E_f t_H = t_G V_f = s_{G^{\text{op}}} V_{f^{\text{op}}}$ and $E_{f^{\text{op}}} t_{H^{\text{op}}} = E_f s_H = s_G V_f = t_{G^{\text{op}}} V_{f^{\text{op}}}$.

Remark 212 Suppose given graph morphisms $f : X \rightarrow Y$ and $g : Y \rightarrow Z$.

(1) We have $(f^{\text{op}})^{\text{op}} = f$.

(2) We have $(fg)^{\text{op}} = f^{\text{op}}g^{\text{op}}$.

Proof.

Ad (1). We have $(X^{\text{op}})^{\text{op}} = X$ and $(Y^{\text{op}})^{\text{op}} = Y$; cf. Definition 211.(1).

We have $V_{(f^{\text{op}})^{\text{op}}} = V_{f^{\text{op}}} = V_f$ and $E_{(f^{\text{op}})^{\text{op}}} = E_{f^{\text{op}}} = E_f$; cf. Definition 211.(1).

Ad (2). We have $V_{(fg)^{\text{op}}} = V_{fg} = V_f V_g = V_{f^{\text{op}}} V_{g^{\text{op}}} = V_{f^{\text{op}}g^{\text{op}}}$ and $E_{(fg)^{\text{op}}} = E_{fg} = E_f E_g = E_{f^{\text{op}}} E_{g^{\text{op}}} = E_{f^{\text{op}}g^{\text{op}}}$; cf. Definition 211. \square

Remark 213 Suppose given $n \in \mathbb{N}$.

We define the graph isomorphism $\zeta_n : C_n \xrightarrow{\sim} C_n^{\text{op}}$ as follows.

We let

$$\begin{aligned} \zeta_n &: C_n \xrightarrow{\sim} C_n^{\text{op}} \\ V_{\zeta_n} &: V_{C_n} \rightarrow V_{C_n^{\text{op}}} \\ &v_k \mapsto v_{-k} \text{ for } k \in \mathbb{Z}/n\mathbb{Z} \\ E_{\zeta_n} &: E_{C_n} \rightarrow E_{C_n^{\text{op}}} \\ &e_k \mapsto e_{-1-k} \text{ for } k \in \mathbb{Z}/n\mathbb{Z} \end{aligned}$$

Suppose given an edge $e_k \in E_{C_n}$.

We have $e_k E_{\zeta_n} s_{C_n^{\text{op}}} = e_{-1-k} s_{C_n^{\text{op}}} = e_{-1-k} t_{C_n} = v_{-k} = v_k V_{\zeta_n} = e_k s_{C_n} V_{\zeta_n}$.

We have $e_k E_{\zeta_n} t_{C_n^{\text{op}}} = e_{-1-k} t_{C_n^{\text{op}}} = e_{-1-k} s_{C_n} = v_{-1-k} = v_{k+1} V_{\zeta_n} = e_k t_{C_n} V_{\zeta_n}$.

So $\zeta_n : C_n \xrightarrow{\sim} C_n^{\text{op}}$ is in fact a graph isomorphism.

We have $(C_n^{\text{op}})^{\text{op}} = C_n$; cf. Remark 212.

We have the graph isomorphism $\zeta_n^{\text{op}} := (V_{\zeta_n}, E_{\zeta_n}) : C_n^{\text{op}} \rightarrow C_n$; cf. Definition 211.

We show that we have $\zeta_n \zeta_n^{\text{op}} \stackrel{!}{=} \text{id}_{C_n}$ and $\zeta_n^{\text{op}} \zeta_n \stackrel{!}{=} \text{id}_{C_n^{\text{op}}}$.

Suppose given a vertex $v_k \in V_{C_n} = V_{C_n^{\text{op}}}$.

We have $v_k V_{\zeta_n \zeta_n^{\text{op}}} = v_k V_{\zeta_n} V_{\zeta_n^{\text{op}}} = v_{-k} V_{\zeta_n^{\text{op}}} = v_{-k} V_{\zeta_n} = v_{-(-k)} = v_k = v_k \text{Vid}_{C_n}$.

We have $v_k V_{\zeta_n^{\text{op}} \zeta_n} = v_k V_{\zeta_n^{\text{op}}} V_{\zeta_n} = v_k V_{\zeta_n} V_{\zeta_n} = v_{-k} V_{\zeta_n} = v_{-(-k)} = v_k = v_k \text{Vid}_{C_n^{\text{op}}}$.

Suppose given an edge $e_k \in E_{C_n} = E_{C_n^{\text{op}}}$.

We have $e_k E_{\zeta_n \zeta_n^{\text{op}}} = e_k E_{\zeta_n} E_{\zeta_n^{\text{op}}} = e_{-1-k} E_{\zeta_n^{\text{op}}} = e_{-1-k} E_{\zeta_n} = e_{-1-(-1-k)} = e_k = e_k \text{Eid}_{C_n}$.

We have $e_k E_{\zeta_n^{\text{op}} \zeta_n} = e_k E_{\zeta_n^{\text{op}}} E_{\zeta_n} = e_k E_{\zeta_n} E_{\zeta_n} = e_{-1-k} E_{\zeta_n} = e_{-1-(-1-k)} = e_k = e_k \text{Eid}_{C_n^{\text{op}}}$.

So we have $\zeta_n^{-1} = \zeta_n^{\text{op}}$. □

Remark 214 Suppose given a graph morphism $f : G \rightarrow H$.

Then the graph morphism $f^{\text{op}} : G^{\text{op}} \rightarrow H^{\text{op}}$ is a quasiisomorphism if and only if the graph morphism $f : G \rightarrow H$ is a quasiisomorphism.

Proof. Suppose given $k \geq 1$.

We will show that the map $(C_k, f)_{\text{Gph}}$ is bijective if and only if the map $(C_k^{\text{op}}, f^{\text{op}})$ is bijective, which is the case if and only if the map (C_k, f^{op}) is bijective.

We consider the following commutative diagram.

$$\begin{array}{ccccc} & & (C_k, G)_{\text{Gph}} & \xrightarrow{(C_k, f)_{\text{Gph}}} & (C_k, H)_{\text{Gph}} & & \\ & & \downarrow \wr \sigma & & \downarrow \tau \wr & & \\ g & & & & & & g \\ \downarrow & & (C_k^{\text{op}}, G^{\text{op}})_{\text{Gph}} & \xrightarrow{(C_k^{\text{op}}, f^{\text{op}})_{\text{Gph}}} & (C_k^{\text{op}}, H^{\text{op}})_{\text{Gph}} & & \downarrow \\ & & \downarrow \wr (\zeta_k, G^{\text{op}})_{\text{Gph}} & & \downarrow \wr (\zeta_k, H^{\text{op}})_{\text{Gph}} & & \\ & & (C_k, G^{\text{op}})_{\text{Gph}} & \xrightarrow{(C_k, f^{\text{op}})_{\text{Gph}}} & (C_k, H^{\text{op}})_{\text{Gph}} & & \\ & & & & & & \\ & & & & & & g^{\text{op}} \end{array}$$

□

Suppose given a graph morphism $u \in (C_k, G)_{\text{Gph}}$.

We have $u^{\text{op}}(C_k^{\text{op}}, f^{\text{op}})_{\text{Gph}} = u^{\text{op}} f^{\text{op}} \stackrel{\text{Rem. 212}}{=} (uf)^{\text{op}} = (u(C_k, f)_{\text{Gph}})^{\text{op}}$.

So the “upper quadrangle” commutes.

Suppose given a graph morphism $r \in (C_k^{\text{op}}, G^{\text{op}})_{\text{Gph}}$.

We have $r(\zeta_n, G^{\text{op}})_{\text{Gph}}(C_k, f^{\text{op}})_{\text{Gph}} = \zeta_n r(C_k, f^{\text{op}})_{\text{Gph}} = \zeta_n r f^{\text{op}} = r f^{\text{op}}(\zeta_n, H^{\text{op}})_{\text{Gph}} = r(C_k^{\text{op}}, f^{\text{op}})_{\text{Gph}}(\zeta_n, H^{\text{op}})_{\text{Gph}}$.

So the “lower quadrangle” commutes.

The maps σ and τ are bijective because of Remark 212.

The maps $(\zeta_n, G^{\text{op}})_{\text{Gph}}$ and $(\zeta_n, H^{\text{op}})_{\text{Gph}}$ are bijective because of Remark 213.

The graph morphism $f : G \rightarrow H$ is a quasiisomorphism if and only if the map $(C_k, f)_{\text{Gph}}$ is bijective for $k \geq 1$.

By the commutative diagram above, $(C_k, f)_{\text{Gph}}$ is bijective if and only if $(C_k, f^{\text{op}})_{\text{Gph}}$ is bijective.

Hence f is a quasiisomorphism if and only if f^{op} is a quasiisomorphism. □

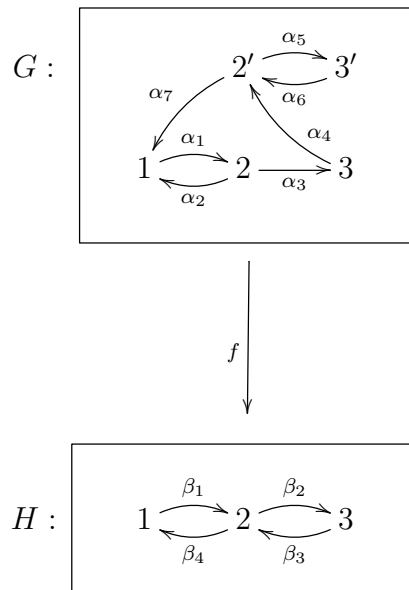
Chapter 9

Some examples and counterexamples

The functions that are used in this section to calculate graphs and graph morphisms are given in §10. They can also be obtained using the electronic appendix, cf. §A.

9.1 Some examples for quasiisomorphisms

Example 215 We consider the following graph morphism.



Here, $f = (V_f, E_f) : G \rightarrow H$ is the graph morphism mapping the vertices and the edges in a vertical way. So we let

$$\begin{aligned} 1 V_f &:= 1, & 2 V_f &:= 2, & 3 V_f &:= 3, \\ & & 2' V_f &:= 2, & 3' V_f &:= 3 \end{aligned}$$

and

$$\begin{aligned}\alpha_1 E_f &:= \beta_1, \\ \alpha_2 E_f &:= \beta_4, \\ \alpha_3 E_f &:= \beta_2, \\ \alpha_4 E_f &:= \beta_3, \\ \alpha_5 E_f &:= \beta_2, \\ \alpha_6 E_f &:= \beta_3, \\ \alpha_7 E_f &:= \beta_4.\end{aligned}$$

Then the graph morphism f is an acyclic fibration.

Proof. Via Magma [2] we will calculate that the graph morphism f is an etale fibration that satisfies (Uni), using the function `SuffCond`.

This will show that f is a quasiisomorphism; cf. Proposition 210.

The necessary Magma functions can be found in §10.7, §10.5 and in §10.6; cf. also §A.

The graph G can be obtained with the function `trygraph` by setting `n := 3`.

The graph H can be obtained with the function `c2chain` by setting `n := 3`.

The graph morphism f can be obtained with the function `tryacyclic` by setting `n := 3`.

So letting

```
G := trygraph(3);
H := c2chain(3);
f := tryacyclic(3);
> G;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>,
                    <5, 5, 4>, <3, 6, 4>, <4, 7, 1> ]>
> H;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
        <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>,
        <<5, 5, 4>, <3, 3, 2>>, <<3, 6, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>> ]>
```

we get

```
> IsGraphMorphism(f,G,H);
true
> SuffCond(f,G,H);
<true, true>
```

Alternative proof without Proposition 210.

Via Magma we calculate that the graph morphism f is in fact a fibration, using the function `IsFibration` given in §10 below.

```
> IsFibration(f,G,H);
true
```

We will now give a pedestrian proof that f is a quasiisomorphism by classifying all morphisms from C_n to G respectively to H .

It might be useful as a blueprint for a quasiisomorphy verification in a case where we cannot apply Proposition 210, or its dual statement using Remark 214.

Let $k \in \mathbb{N}$.

We remark that $(C_{2k-1}, H) = \emptyset$ and thus $(C_{2k-1}, G) = \emptyset$.

To prove that f is acyclic, we have to show that the map $(C_{2k}, f) : (C_{2k}, G) \rightarrow (C_{2k}, H)$ is bijective.

We have to show that for each graph morphism $u : C_{2k} \rightarrow H$ there exists a unique graph morphism $v : C_{2k} \rightarrow G$ such that $vf = u$.

$$\begin{array}{ccc} & & G \\ & \nearrow \exists! v & \downarrow f \\ C_{2k} & \xrightarrow{u} & H \end{array}$$

We first show that it suffices to show this for each graph morphism $u : C_{2k} \rightarrow H$ such that $v_0 V_u = 2$.

Write $(C_{2k}, H)_2 := \{u \in (C_{2k}, H) : u \text{ graph morphism, } v_0 V_u = 2\}$.

Write $(C_{2k}, G)_{2,2'} := \{u \in (C_{2k}, G) : u \text{ graph morphism, } v_0 V_u \in \{2, 2'\}\}$.

The map $(C_{2k}, f) : (C_{2k}, G) \rightarrow (C_{2k}, H)$ restricts to the map

$$(C_{2k}, f)_2 := (C_{2k}, f)|_{(C_{2k}, G)_{2,2'}} : (C_{2k}, G)_{2,2'} \rightarrow (C_{2k}, H)_2.$$

Recall that the vertices of the graph C_{2k} are written v_i for $i \in \mathbb{Z}/2k\mathbb{Z}$.

Recall that the edges of the graph C_{2k} are written e_i for $i \in \mathbb{Z}/2k\mathbb{Z}$.

Claim. If $(C_{2k}, f)_2$ is bijective, then (C_{2k}, f) is bijective.

Suppose given $u : C_{2k} \rightarrow H$. We have to show that there exists a unique graph morphism $v : C_{2k} \rightarrow G$ such that $vf = u$.

$$\begin{array}{ccc} & & G \\ & \nearrow \exists! v & \downarrow f \\ C_{2k} & \xrightarrow{u} & H \end{array}$$

A consideration of H shows that there exists $s \in \mathbb{Z}/2k\mathbb{Z}$ such that $v_s V_u = 2$.

Let $\tilde{u} := a_s \cdot u : C_{2k} \rightarrow H$.

We have $v_0 V_{\tilde{u}} = v_0 V_{a_s V_u} = v_s V_u = 2$.

Since $(C_{2k}, f)_2$ is bijective, there exists a unique graph morphism $\tilde{v} \in (C_{2k}, G)_{2,2'}$ such that $\tilde{v}f = \tilde{u}$.

$$\begin{array}{ccc} & & G \\ & \nearrow \exists! \tilde{v} & \downarrow f \\ C_{2k} & \xrightarrow{\tilde{u}} & H \end{array}$$

To prove *existence*, we let $v := a_s^{-1} \cdot \tilde{v}$.

We obtain $vf = a_s^{-1} \tilde{v}f = a_s^{-1} \tilde{u} = u$.

Uniqueness. Suppose given $v, v' \in (C_{2k}, G)$ such that $vf = u = v'f$.

We have to show that $v \stackrel{!}{=} v'$.

We have $a_s vf = a_s u = \tilde{u}$ and $a_s v'f = a_s u = \tilde{u}$.

Note that $V_f^{-1}(\{2\}) = \{2, 2'\}$.

In particular, $v_0 V_{\tilde{u}} = 2$ implies $v_0 V_{a_s v} \in \{2, 2'\}$ and $v_0 V_{a_s v'} \in \{2, 2'\}$, so $a_s v, a_s v' \in (C_{2k}, G)_{2, 2'}$.

Since $(C_{2k}, f)_2$ is bijective, we conclude that $a_s v = a_s v'$. Thus $v = v'$.

This proves the *claim*.

Note that the map

$$\begin{aligned} \mathbb{Z}/k\mathbb{Z} &\rightarrow \mathbb{Z}/2k\mathbb{Z} \\ i + k\mathbb{Z} &\mapsto 2i + 2k\mathbb{Z} \end{aligned}$$

exists.

Note that for a graph morphism $r : C_{2k} \rightarrow H$ in $(C_{2k}, H)_2$ we have $v_{2i} V_r \in \{2\}$, $v_{2i+1} V_r \in \{1, 3\}$, $e_{2i} E_r \in \{\beta_2, \beta_4\}$ and $e_{2i+1} E_r \in \{\beta_1, \beta_3\}$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

Note that for a graph morphism $r : C_{2k} \rightarrow G$ in $(C_{2k}, G)_{2, 2'}$ we have $v_{2i} V_r \in \{2, 2'\}$, $v_{2i+1} V_r \in \{1, 3, 3'\}$, $e_{2i} E_r \in \{\alpha_2, \alpha_3, \alpha_5, \alpha_7\}$ and $e_{2i+1} E_r \in \{\alpha_1, \alpha_4, \alpha_6\}$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

We will introduce bijective maps $\gamma : (C_{2k}, G)_{2, 2'} \rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$ and $\delta : (C_{2k}, H)_2 \rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$. In order to show that $(C_{2k}, f)_2$ is bijective, we will show that $\gamma = (C_{2k}, f)_2 \cdot \delta$.

We establish the bijection $\delta : (C_{2k}, H)_2 \rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$.

Let

$$\begin{aligned} \delta : (C_{2k}, H)_2 &\rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} \\ u &\mapsto u\delta : i \mapsto \begin{cases} 0 & \text{if } v_{2i+1} V_u = 1 \\ 1 & \text{if } v_{2i+1} V_u = 3. \end{cases} \end{aligned}$$

Let

$$\begin{aligned} \tilde{\delta} : \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} &\rightarrow (C_{2k}, H)_2 \\ \varphi &\mapsto \varphi\tilde{\delta} \end{aligned}$$

be defined as follows.

Because the graph H is thin it suffices to give $V_{\varphi\tilde{\delta}}$; cf. Remark 77.(1).

Let

$$\begin{aligned} V_{\varphi\tilde{\delta}} : V_{C_{2k}} &\rightarrow V_H = \{1, 2, 3\} \\ v_{2i} &\mapsto 2, \text{ for } i \in \mathbb{Z}/k\mathbb{Z} \\ v_{2i+1} &\mapsto \begin{cases} 1 & \text{if } i\varphi = 0 \\ 3 & \text{if } i\varphi = 1 \end{cases} \text{ for } i \in \mathbb{Z}/k\mathbb{Z}. \end{aligned}$$

To show that the graph morphism $\varphi\tilde{\delta}$ is well-defined, we show that there is an edge in E_H from $v_j V_{\varphi\tilde{\delta}}$ to $v_{j+1} V_{\varphi\tilde{\delta}}$ for $j \in \mathbb{Z}/2k\mathbb{Z}$; cf. Remark 79.

We consider two cases:

Case 1: $j = 2i$ for some $i \in \mathbb{Z}/k\mathbb{Z}$.

Then $v_j V_{\varphi\tilde{\delta}} = v_{2i} V_{\varphi\tilde{\delta}} = 2$ and $v_{j+1} V_{\varphi\tilde{\delta}} \in \{1, 3\}$.

Subcase: $i\varphi = 0$. Then $v_{j+1} V_{\varphi\tilde{\delta}} = v_{2i+1} V_{\varphi\tilde{\delta}} = 1$. And we have $\beta_4 \in H(2, 1)$.

Subcase: $i\varphi = 1$. Then $v_{j+1} V_{\varphi\tilde{\delta}} = v_{2i+1} V_{\varphi\tilde{\delta}} = 3$. And we have $\beta_2 \in H(2, 3)$.

Case 2: $j = 2i + 1$ for some $i \in \mathbb{Z}/k\mathbb{Z}$.

Then $v_j V_{\varphi\tilde{\delta}} = v_{2i+1} V_{\varphi\tilde{\delta}} \in \{1, 3\}$ and $v_{j+1} V_{\varphi\tilde{\delta}} = v_{2i+2} V_{\varphi\tilde{\delta}} = v_{2(i+1)} V_{\varphi\tilde{\delta}} = 2$.

Subcase: $i\varphi = 0$. Then $v_j V_{\varphi\tilde{\delta}} = v_{2i+1} V_{\varphi\tilde{\delta}} = 1$. And we have $\beta_1 \in H(1, 2)$.

Subcase: $i\varphi = 1$. Then $v_j V_{\varphi\tilde{\delta}} = v_{2i+1} V_{\varphi\tilde{\delta}} = 3$. And we have $\beta_3 \in H(3, 2)$.

We have to show that $\delta \cdot \tilde{\delta} \stackrel{!}{=} \text{id}_{(C_{2k}, H)_2}$ and that $\tilde{\delta} \cdot \delta \stackrel{!}{=} \text{id}_{\{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}}$.

Ad $\delta \cdot \tilde{\delta} \stackrel{!}{=} \text{id}_{(C_{2k}, H)_2}$. We have to show that $(u)\delta\tilde{\delta} \stackrel{!}{=} u$ for $u \in (C_{2k}, H)_2$.

We have to show that $V_{u\delta\tilde{\delta}} \stackrel{!}{=} V_u$; cf. Remark 77.(1).

We have $v_{2i} V_{(u\delta)\tilde{\delta}} = 2 = v_{2i} V_u$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

It remains to show that $v_{2i+1} V_{(u\delta)\tilde{\delta}} \stackrel{!}{=} v_{2i+1} V_u$ in $\{1, 3\}$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

So we have to show that $1 \stackrel{!}{=} v_{2i+1} V_u$ if $i(u\delta) = 0$ and that $3 \stackrel{!}{=} v_{2i+1} V_u$ if $i(u\delta) = 1$. This holds by definition of δ .

Ad $\tilde{\delta} \cdot \delta \stackrel{!}{=} \text{id}_{\{0,1\}^{\mathbb{Z}/k\mathbb{Z}}}$. We have to show that $\varphi\delta^{-1}\delta \stackrel{!}{=} \varphi$ for $\varphi \in \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$. We obtain

$$i((\varphi\tilde{\delta})\delta) \stackrel{\text{Def. } \delta}{=} \begin{cases} 0 & \text{if } v_{2i+1} V_{\varphi\tilde{\delta}} = 1 \\ 1 & \text{if } v_{2i+1} V_{\varphi\tilde{\delta}} = 3 \end{cases} \stackrel{\text{Def. } \tilde{\delta}}{=} \begin{cases} 0 & \text{if } i\varphi = 0 \\ 1 & \text{if } i\varphi = 1 \end{cases} = i\varphi$$

for $i \in \mathbb{Z}/k\mathbb{Z}$.

So we have $\tilde{\delta} = \delta^{-1}$.

We now establish the bijection $\gamma : (C_{2k}, G)_{2,2'} \rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$.

Let

$$\begin{aligned} \gamma : (C_{2k}, G)_{2,2'} &\rightarrow \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} \\ v &\mapsto v\gamma : i \mapsto \begin{cases} 0 & \text{if } v_{2i+1} V_v = 1 \\ 1 & \text{if } v_{2i+1} V_v \in \{3, 3'\} \end{cases} \end{aligned}$$

Let

$$\begin{aligned} \tilde{\gamma} : \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} &\rightarrow (C_{2k}, G)_{2,2'} \\ \varphi &\mapsto \varphi\tilde{\gamma} \end{aligned}$$

be defined as follows.

Because the graph G is thin it suffices to give $V_{\varphi\tilde{\gamma}}$; cf. Remark 77.(1).

Let

$$\begin{aligned} V_{\varphi\tilde{\gamma}} : V_{C_{2k}} &\rightarrow V_G = \{1, 2, 2', 3, 3'\} \\ v_{2i+1} &\mapsto \begin{cases} 1 & \text{if } i\varphi = 0 \\ 3 & \text{if } i\varphi = 1 \text{ and } v_{2i} V_{\varphi\tilde{\gamma}} = 2 \\ 3' & \text{if } i\varphi = 1 \text{ and } v_{2i} V_{\varphi\tilde{\gamma}} = 2' \end{cases} \text{ for } i \in \mathbb{Z}/k\mathbb{Z} \\ v_{2i+2} &\mapsto \begin{cases} 2 & \text{if } i\varphi = 0 \\ 2' & \text{if } i\varphi = 1 \end{cases} \text{ for } i \in \mathbb{Z}/k\mathbb{Z}. \end{aligned}$$

To show that the graph morphism $\varphi\tilde{\gamma}$ is well-defined, we show that there is an edge in E_G from $v_j V_{\varphi\tilde{\gamma}}$ to $v_{j+1} V_{\varphi\tilde{\gamma}}$ for $j \in \mathbb{Z}/2k\mathbb{Z}$; cf. Remark 79.

We consider two cases:

Case 1: $j = 2i + 2$ for some $i \in \mathbb{Z}/k\mathbb{Z}$.

Then $v_j V_{\varphi\tilde{\gamma}} = v_{2i+2} V_{\varphi\tilde{\gamma}} \in \{2, 2'\}$ and $v_{j+1} V_{\varphi\tilde{\gamma}} = v_{2i+3} V_{\varphi\tilde{\gamma}} \in \{1, 3, 3'\}$.

Subcase: $i\varphi = 0$. Then $v_j V_{\varphi\tilde{\gamma}} = v_{2i+2} V_{\varphi\tilde{\gamma}} = 2$.

Subsubcase: $(i+1)\varphi = 0$. Then $v_{j+1} V_{\varphi\tilde{\gamma}} = v_{2(i+1)+1} V_{\varphi\tilde{\gamma}} = v_{2i+3} V_{\varphi\tilde{\gamma}} = 1$. And we have $\alpha_2 \in G(2, 1)$.

Subsubcase: $(i+1)\varphi = 1$. Then $v_{j+1} V_{\varphi\tilde{\gamma}} = v_{2(i+1)+1} V_{\varphi\tilde{\gamma}} = v_{2i+3} V_{\varphi\tilde{\gamma}} = 3$. And we have $\alpha_3 \in G(2, 3)$.

Subcase: $i\varphi = 1$. Then $v_j V_{\varphi\tilde{\gamma}} = v_{2i+2} V_{\varphi\tilde{\gamma}} = 2'$.

Subsubcase: $(i+1)\varphi = 0$. Then $v_{j+1} V_{\varphi\tilde{\gamma}} = v_{2(i+1)+1} V_{\varphi\tilde{\gamma}} = v_{2i+3} V_{\varphi\tilde{\gamma}} = 1$. And we have $\alpha_7 \in G(2', 1)$.

Subsubcase: $(i+1)\varphi = 1$. Then $v_{j+1}V_{\varphi\tilde{\gamma}} = v_{2(i+1)+1}V_{\varphi\tilde{\gamma}} = v_{2i+3}V_{\varphi\tilde{\gamma}} = 3'$. And we have $\alpha_5 \in G(2', 3')$.

Case 2: $j = 2i + 1$ for some $i \in \mathbb{Z}/k\mathbb{Z}$.

Then $v_jV_{\varphi\tilde{\gamma}} = v_{2i+1}V_{\varphi\tilde{\gamma}} \in \{1, 3, 3'\}$ and $v_{j+1}V_{\varphi\tilde{\gamma}} = v_{2i+2}V_{\varphi\tilde{\gamma}} \in \{2, 2'\}$.

Subcase: $i\varphi = 0$. Then $v_jV_{\varphi\tilde{\gamma}} = v_{2i+1}V_{\varphi\tilde{\gamma}} = 1$ and $v_{j+1}V_{\varphi\tilde{\gamma}} = v_{2i+2}V_{\varphi\tilde{\gamma}} = 2$. And we have $\alpha_1 \in G(1, 2)$.

Subcase: $i\varphi = 1$. Then $v_jV_{\varphi\tilde{\gamma}} = v_{2i+1}V_{\varphi\tilde{\gamma}} \in \{3, 3'\}$ and $v_{j+1}V_{\varphi\tilde{\gamma}} = v_{2i+2}V_{\varphi\tilde{\gamma}} = 2'$. And we have $\alpha_4 \in G(3, 2')$ and $\alpha_6 \in G(3', 2')$.

We have to show that $\gamma \cdot \tilde{\gamma} \stackrel{\dagger}{=} \text{id}_{(C_{2k}, G)_{2, 2'}}$ and that $\tilde{\gamma} \cdot \gamma \stackrel{\dagger}{=} \text{id}_{\{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}}$.

Ad $\gamma \cdot \tilde{\gamma} \stackrel{\dagger}{=} \text{id}_{(C_{2k}, G)_{2, 2'}}$. We have to show that $(v)\gamma\tilde{\gamma} \stackrel{\dagger}{=} v$ for $v \in (C_{2k}, G)_{2, 2'}$.

We have to show that $V_{v\gamma\tilde{\gamma}} \stackrel{\dagger}{=} V_v$; cf. Remark 77.(1).

We have to show that $v_{2i+2}V_{(v\gamma)\tilde{\gamma}} \stackrel{\dagger}{=} v_{2i+2}V_v$ in $\{2, 2'\}$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

So we have to show that $2 \stackrel{\dagger}{=} v_{2i+2}V_v$ if $i(v\gamma) = 0$ and that $2' \stackrel{\dagger}{=} v_{2i+2}V_v$ if $i(v\gamma) = 1$.

If $i(v\gamma) = 0$ then we have $v_{2i+1}V_v = 1$ by definition of γ and consequently $v_{2i+2}V_v = 2$.

If $i(v\gamma) = 1$ then we have $v_{2i+1}V_v \in \{3, 3'\}$ by definition of γ and consequently $v_{2i+2}V_v = 2'$. This holds by definition of γ .

We have to show that $v_{2i+1}V_{(v\gamma)\tilde{\gamma}} \stackrel{\dagger}{=} v_{2i+1}V_v$ in $\{1, 3, 3'\}$ for $i \in \mathbb{Z}/k\mathbb{Z}$.

So we have to show that $1 \stackrel{\dagger}{=} v_{2i+1}V_v$ if $i(v\gamma) = 0$ and that $3 \stackrel{\dagger}{=} v_{2i+1}V_v$ if $i(v\gamma) = 1$ and $v_{2i}V_{(v\gamma)\tilde{\gamma}} = 2$ and that $3' \stackrel{\dagger}{=} v_{2i+1}V_v$ if $i(v\gamma) = 1$ and $v_{2i}V_{(v\gamma)\tilde{\gamma}} = 2'$.

If $i(v\gamma) = 0$ then we have $v_{2i+1}V_v = 1$ by definition of γ .

If $i(v\gamma) = 1$ then, by definition of γ , we have $v_{2i+1}V_v \in \{3, 3'\}$.

If $i(v\gamma) = 1$ and $v_{2i}V_{(v\gamma)\tilde{\gamma}} = 2$ then, by definition of $\tilde{\gamma}$, we have $(i-1)(v\gamma) = 0$. Then, by definition of γ , we have $v_{2(i-1)+1}V_v = v_{2i-1}V_v = 1$. By the structure of G we obtain $v_{2i+1}V_v = 3$.

If $i(v\gamma) = 1$ and $v_{2i}V_{(v\gamma)\tilde{\gamma}} = 2'$ then, by definition of $\tilde{\gamma}$, we have $(i-1)(v\gamma) = 1$. Then, by definition of γ , we have $v_{2(i-1)+1}V_v = v_{2i-1}V_v \in \{3, 3'\}$. By the structure of G we obtain $v_{2i+1}V_v = 3'$.

Ad $\tilde{\gamma} \cdot \gamma \stackrel{\dagger}{=} \text{id}_{\{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}}$. We have to show that $\varphi\tilde{\gamma}\gamma \stackrel{\dagger}{=} \varphi$ for $\varphi \in \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}}$. We obtain

$$i((\varphi\tilde{\gamma})\gamma) \stackrel{\text{Def. } \gamma}{=} \begin{cases} 0 & \text{if } v_{2i+1}V_{\varphi\tilde{\gamma}} = 1 \\ 1 & \text{if } v_{2i+1}V_{\varphi\tilde{\gamma}} \in \{3, 3'\} \end{cases} \stackrel{\text{Def. } \tilde{\gamma}}{=} \begin{cases} 0 & \text{if } i\varphi = 0 \\ 1 & \text{if } i\varphi = 1 \end{cases} = i\varphi$$

for $i \in \mathbb{Z}/k\mathbb{Z}$.

We have to show commutativity of the following diagram.

$$\begin{array}{ccc} (C_{2k}, G)_{2, 2'} & \xrightarrow{\gamma} & \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} \\ (C_{2k}, f) \downarrow & & \downarrow \text{id} \\ (C_{2k}, H)_2 & \xrightarrow{\delta} & \{0, 1\}^{\mathbb{Z}/k\mathbb{Z}} \end{array}$$

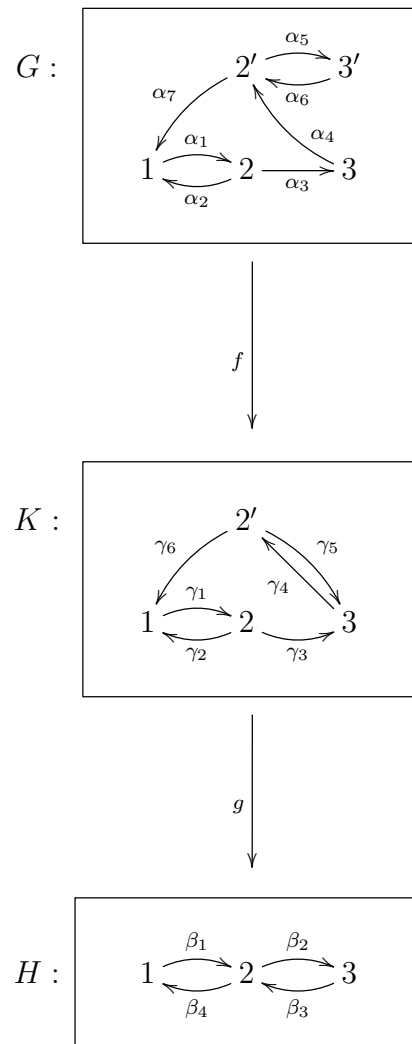
We have to show that $v\gamma \stackrel{\dagger}{=} (v \cdot f)\delta$ for $v \in (C_{2k}, G)_{2, 2'}$.

For $i \in \mathbb{Z}/k\mathbb{Z}$, we have

$$i((v \cdot f)\delta) = \begin{cases} 0 & \text{if } v_{2i+1}V_{v \cdot f} = 1 \\ 1 & \text{if } v_{2i+1}V_{v \cdot f} = 3 \end{cases} \stackrel{\text{pre-images}}{=} \begin{cases} 0 & \text{if } v_{2i+1}V_v = 1 \\ 1 & \text{if } v_{2i+1}V_v \in \{3, 3'\} \end{cases} = i(v\gamma).$$

□

Example 216 We consider the following graph morphisms, where G and H are as in Example 215.



Here, the graph morphisms $f: G \rightarrow K$ and $g: K \rightarrow H$ map the vertices and the edges in a vertical way, where $2'V_f = 2', 2V_f = 2$.

We will verify that the graph morphisms f, g and fg are étale fibrations that satisfy (Uni).

This will show that f, g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma [2] we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `trygraph` by setting `n := 3`.

The graph K can be obtained with the function `idtrygraph` by setting `n := 3`.

The graph H can be obtained with the function `c2chain` by setting `n := 3`.

The graph morphism f can be obtained with the function `tryfactorization` by setting `n := 3`.

The graph morphism g can be obtained with the function `idtryacyclic` by setting `n := 3`.

The graph morphism fg can be obtained with the function `tryacyclic` by setting `n := 3`.

So letting

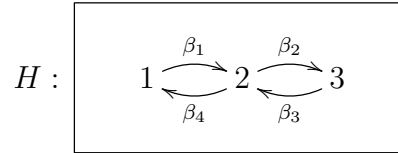
```
G := trygraph(3);
K := idtrygraph(3);
H := c2chain(3);
f := tryfactorization(3);
g := idtryacyclic(3);
fg := tryacyclic(3);
> G;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>, <5, 5, 4>,
                    <3, 6, 4>, <4, 7, 1> ]>
> K;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>, <3, 5, 4>,
                  <4, 7, 1> ]>
> H;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 2, 1>>, <<2, 3, 3>, <2, 3, 3>>, <<4, 4, 5>, <4, 4, 3>>,
      <<5, 5, 4>, <3, 5, 4>>, <<3, 6, 4>, <3, 5, 4>>, <<4, 7, 1>, <4, 7, 1>> ]>
> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 3>, <2, 2, 3>>,
      <<3, 5, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>,
      <<5, 5, 4>, <3, 3, 2>>, <<3, 6, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>> ]>
```

we get

```
> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
> SuffCond(g,K,H);
<true, true>
> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>
```

Varying the input n , we obtain further quasiisomorphisms; cf. Examples 218 and 219 below. This seems to hold independently of $n \geq 3$.

Example 217 We consider the thin graph H from Example 215.

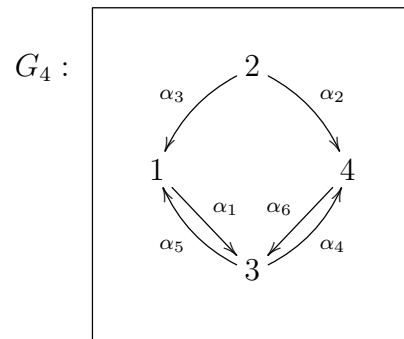
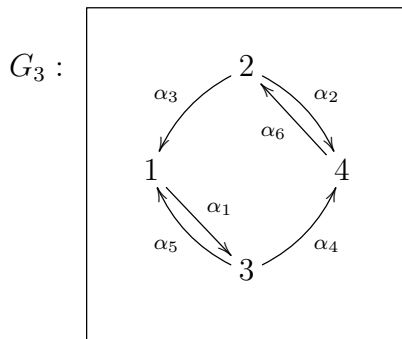
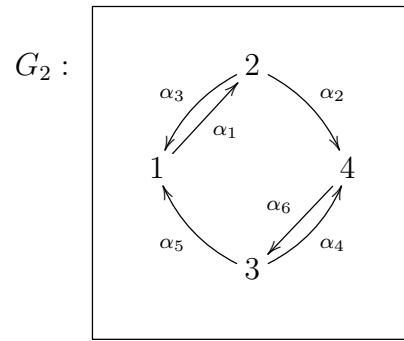
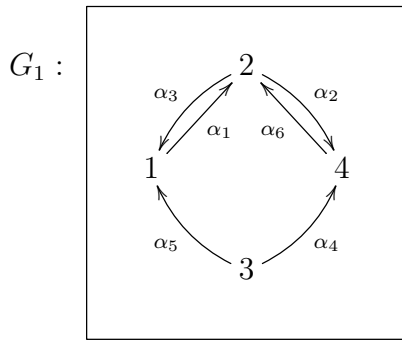


We want to find graphs G and graph morphisms $f : G \rightarrow H$ such that $V_G = \{1, 2, 3, 4\}$ and $1V_f := 1, 2V_f := 2, 3V_f := 2$ and $4V_f := 3$ and such that f is an étale fibration that satisfies (Uni).

Via Magma we proceed as follows.

```
L := EFU(c2chain(3), [1, 2, 1]);
> L;
[
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 2>, <3, 3, 2>> ]>,
  <[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
    <4, 6, 2> ]>, <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 3>, <3, 3, 2>> ]>,
  <[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
    <4, 6, 3> ]>, <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 3>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 2>, <3, 3, 2>> ]>,
  <[ 1, 2, 3, 4 ], [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
    <4, 6, 2> ]>, <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 3>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 3>, <3, 3, 2>> ]>,
  <[ 1, 2, 3, 4 ], [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
    <4, 6, 3> ]>, <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>
]

> [x[2] : x in L];
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
  <4, 6, 2> ]>
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
  <4, 6, 3> ]>
<[ 1, 2, 3, 4 ], [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
  <4, 6, 2> ]>
<[ 1, 2, 3, 4 ], [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>,
  <4, 6, 3> ]>
```

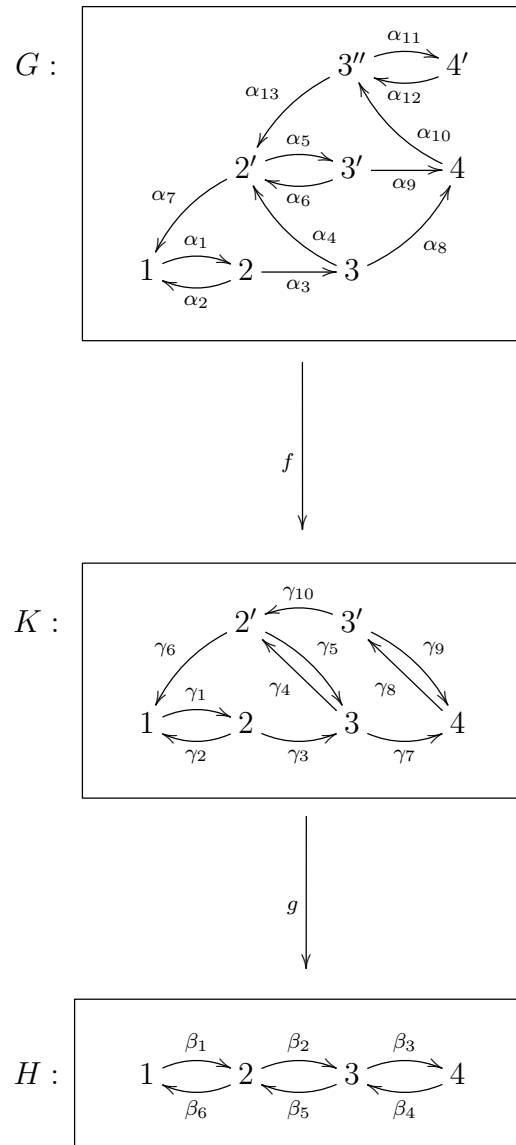


Note that $G_1 \simeq G_4$ and $G_2 \simeq G_3$, with isomorphisms respecting the morphisms to H .

Since our sufficient condition is satisfied the graph morphisms are quasiisomorphisms; cf. Proposition 210.

Cf. also $g : K \rightarrow H$ in Example 216, where K is isomorphic to G_3 .

Example 218 We consider the following graph morphisms.



Here, the graph morphisms $f : G \rightarrow K$ and $g : K \rightarrow H$ map the vertices and the edges in a vertical way, where $2' V_f = 2'$, $2 V_f = 2$, $3'' V_f = 3'$, $3' V_f = 3$ and $3 V_f = 3$.

We will verify that the graph morphisms f , g and fg are etale fibrations that satisfy (Uni).

This will show that f , g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `trygraph` by setting `n := 4`.

The graph K can be obtained with the function `idtrygraph` by setting `n := 4`.

The graph H can be obtained with the function `c2chain` by setting `n := 4`.

The graph morphism f can be obtained with the function `tryfactorization` by setting `n := 4`.

The graph morphism g can be obtained with the function `idtryacyclic` by setting `n := 4`.

The graph morphism fg can be obtained with the function `tryacyclic` by setting `n := 4`.

So letting

```
G := trygraph(4);
K := idtrygraph(4);
H := c2chain(4);
f := tryfactorization(4);
g := idtryacyclic(4);
fg := tryacyclic(4);
> G;
<[ 1, 2, 3, 4, 5, 6, 7, 8 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>,
  <5, 5, 4>, <5, 6, 6>, <7, 7, 8>, <8, 8, 7>, <3, 9, 4>, <4, 10, 1>,
  <3, 11, 6>, <6, 12, 7>, <7, 13, 4> ]>
> K;
<[ 1, 2, 3, 4, 6, 7 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>,
  <3, 5, 4>, <3, 6, 6>, <7, 7, 6>, <6, 8, 7>, <4, 10, 1>, <7, 13, 4> ]>
> H;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>,
  <4, 4, 3>, <3, 5, 2>, <2, 6, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3>, <6, 6>, <7, 7>, <8, 6> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>>, <<2, 3, 3>, <2, 3, 3>>,
  <<4, 4, 5>, <4, 4, 3>>, <<5, 5, 4>, <3, 5, 4>>, <<5, 6, 6>, <3, 6, 6>>,
  <<7, 7, 8>, <7, 7, 6>>, <<8, 8, 7>, <6, 8, 7>>, <<3, 9, 4>, <3, 5, 4>>,
  <<4, 10, 1>, <4, 10, 1>>, <<3, 11, 6>, <3, 6, 6>>,
  <<6, 12, 7>, <6, 8, 7>>, <<7, 13, 4>, <7, 13, 4>> ]>
> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <6, 4>, <7, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 6, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 3>, <2, 2, 3>>,
  <<3, 5, 4>, <3, 5, 2>>, <<3, 6, 6>, <3, 3, 4>>, <<7, 7, 6>, <3, 3, 4>>,
  <<6, 8, 7>, <4, 4, 3>>, <<4, 10, 1>, <2, 6, 1>>, <<7, 13, 4>, <3, 5, 2>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3>, <6, 4>, <7, 3>, <8, 4> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 6, 1>>, <<2, 3, 3>, <2, 2, 3>>,
  <<4, 4, 5>, <2, 2, 3>>, <<5, 5, 4>, <3, 5, 2>>, <<5, 6, 6>, <3, 3, 4>>,
  <<7, 7, 8>, <3, 3, 4>>, <<8, 8, 7>, <4, 4, 3>>, <<3, 9, 4>, <3, 5, 2>>,
  <<4, 10, 1>, <2, 6, 1>>, <<3, 11, 6>, <3, 3, 4>>, <<6, 12, 7>,
  <4, 4, 3>>, <<7, 13, 4>, <3, 5, 2>> ]>
```

we get

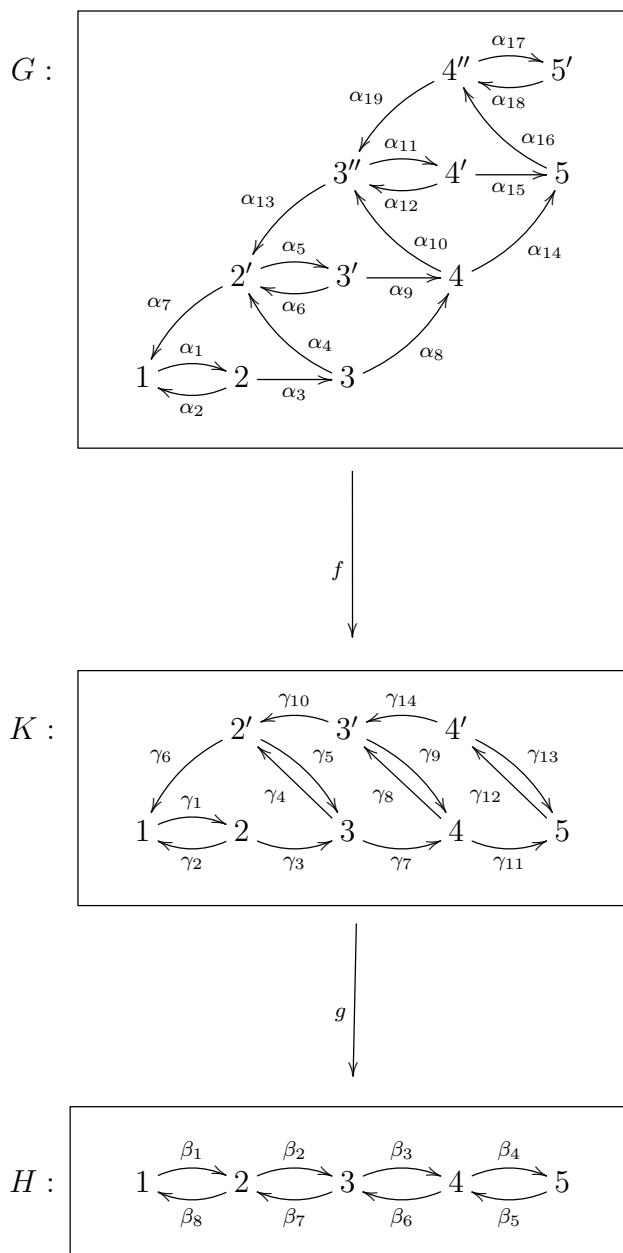
```
> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
```

```

> SuffCond(g,K,H);
<true, true>
> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>

```

Example 219 We consider the following graph morphisms.



Here, the graph morphisms $f : G \rightarrow K$ and $g : K \rightarrow H$ map the vertices and the edges in a vertical way, where $2 \mathcal{V}_f = 2$, $2' \mathcal{V}_f = 2'$, $3 \mathcal{V}_f = 3$, $3' \mathcal{V}_f = 3$, $3'' \mathcal{V}_f = 3'$, $4 \mathcal{V}_f = 4$, $4' \mathcal{V}_f = 4$ and $4'' \mathcal{V}_f = 4'$.

We will verify that the graph morphisms f , g and fg are étale fibrations that satisfy (Uni).

This will show that f , g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `trygraph` by setting `n := 5`.

The graph K can be obtained with the function `idtrygraph` by setting `n := 5`.

The graph H can be obtained with the function `c2chain` by setting `n := 5`.

The graph morphism f can be obtained with the function `tryfactorization` by setting `n := 5`.

The graph morphism g can be obtained with the function `idtryacyclic` by setting `n := 5`.

The graph morphism fg can be obtained with the function `tryacyclic` by setting `n := 5`.

So letting

```
G := trygraph(5);
K := idtrygraph(5);
H := c2chain(5);
f := tryfactorization(5);
g := idtryacyclic(5);
fg := tryacyclic(5);
> G;
<[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>,
  <4, 4, 5>, <5, 5, 4>, <5, 6, 6>, <7, 7, 8>, <8, 8, 7>, <8, 9, 9>,
  <10, 10, 11>, <11, 11, 10>, <3, 12, 4>, <4, 13, 1>, <3, 14, 6>,
  <6, 15, 7>, <7, 16, 4>, <6, 17, 9>, <9, 18, 10>, <10, 19, 7> ]>
> K;
<[ 1, 2, 3, 4, 6, 7, 9, 10 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>,
  <3, 5, 4>, <3, 6, 6>, <7, 7, 6>, <6, 8, 7>, <6, 9, 9>, <10, 10, 9>,
  <9, 11, 10>, <4, 13, 1>, <7, 16, 4>, <10, 19, 7> ]>
> H;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>, <4, 4, 5>, <5, 5, 4>,
  <4, 6, 3>, <3, 7, 2>, <2, 8, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3>, <6, 6>, <7, 7>, <8, 6>, <9, 9>,
  <10, 10>, <11, 9> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>>,
  <<2, 3, 3>, <2, 3, 3>>, <<4, 4, 5>, <4, 4, 3>>, <<5, 5, 4>, <3, 5, 4>>,
  <<5, 6, 6>, <3, 6, 6>>, <<7, 7, 8>, <7, 7, 6>>, <<8, 8, 7>, <6, 8, 7>>,
  <<8, 9, 9>, <6, 9, 9>>, <<10, 10, 11>, <10, 10, 9>>,
  <<11, 11, 10>, <9, 11, 10>>, <<3, 12, 4>, <3, 5, 4>>,
  <<4, 13, 1>, <4, 13, 1>>, <<3, 14, 6>, <3, 6, 6>>,
  <<6, 15, 7>, <6, 8, 7>>, <<7, 16, 4>, <7, 16, 4>>,
  <<6, 17, 9>, <6, 9, 9>>, <<9, 18, 10>, <9, 11, 10>>,
  <<10, 19, 7>, <10, 19, 7>> ]>
```

```

> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <6, 4>, <7, 3>, <9, 5>, <10, 4> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 8, 1>>, <<2, 3, 3>, <2, 2, 3>>,
    <<4, 4, 3>, <2, 2, 3>>, <<3, 5, 4>, <3, 7, 2>>, <<3, 6, 6>, <3, 3, 4>>,
    <<7, 7, 6>, <3, 3, 4>>, <<6, 8, 7>, <4, 6, 3>>, <<6, 9, 9>, <4, 4, 5>>,
    <<10, 10, 9>, <4, 4, 5>>, <<9, 11, 10>, <5, 5, 4>>,
    <<4, 13, 1>, <2, 8, 1>>, <<7, 16, 4>, <3, 7, 2>>,
    <<10, 19, 7>, <4, 6, 3>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3>, <6, 4>, <7, 3>, <8, 4>, <9, 5>,
  <10, 4>, <11, 5> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 8, 1>>,
  <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>, <<5, 5, 4>, <3, 7, 2>>,
  <<5, 6, 6>, <3, 3, 4>>, <<7, 7, 8>, <3, 3, 4>>, <<8, 8, 7>, <4, 6, 3>>,
  <<8, 9, 9>, <4, 4, 5>>, <<10, 10, 11>, <4, 4, 5>>,
  <<11, 11, 10>, <5, 5, 4>>, <<3, 12, 4>, <3, 7, 2>>,
  <<4, 13, 1>, <2, 8, 1>>, <<3, 14, 6>, <3, 3, 4>>,
  <<6, 15, 7>, <4, 6, 3>>, <<7, 16, 4>, <3, 7, 2>>,
  <<6, 17, 9>, <4, 4, 5>>, <<9, 18, 10>, <5, 5, 4>>,
  <<10, 19, 7>, <4, 6, 3>> ]>

```

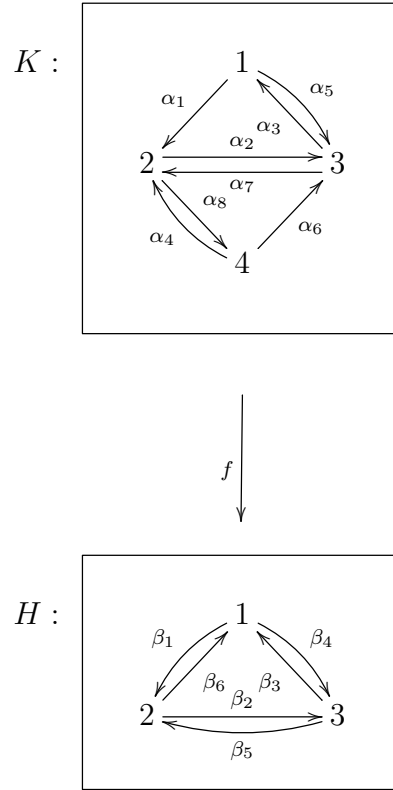
we get

```

> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
> SuffCond(g,K,H);
<true, true>
> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>

```

Example 220 We consider the following graph morphism.



Here, the graph morphism $f : K \rightarrow H$ maps the vertices and the edges in a vertical way, i.e. we let

$$1 V_f := 1, \quad 2 V_f := 2, \quad 3 V_f := 3, \quad 4 V_f := 1$$

and

$$\begin{aligned} \alpha_1 E_f &:= \beta_1, \\ \alpha_2 E_f &:= \beta_2, \\ \alpha_3 E_f &:= \beta_3, \\ \alpha_4 E_f &:= \beta_1, \\ \alpha_5 E_f &:= \beta_4, \\ \alpha_6 E_f &:= \beta_4, \\ \alpha_7 E_f &:= \beta_5, \\ \alpha_8 E_f &:= \beta_6. \end{aligned}$$

Then the graph morphism f is an acyclic fibration.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph K can be obtained with the function `idTrygraph` by setting `n := 3`.

The graph H can be obtained with the function `Doublecyclic` by setting `n := 3`.

The graph morphism f can be obtained with the function `idTryacyclic` by setting `n := 3`.

So letting

```

K := idTrygraph(3);
H := Doublecyclic(3);
f := idTryacyclic(3);
> K;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>, <3, 5, 4>,
  <4, 7, 1>, <1, 8, 3>, <3, 9, 1> ]>
> H;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>,
  <2, 4, 1>, <1, 5, 3>, <3, 6, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 3>, <2, 2, 3>>,
  <<3, 5, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>>, <<1, 8, 3>, <1, 5, 3>>,
  <<3, 9, 1>, <3, 6, 1>> ]>

```

we get

```

> IsGraphMorphism(f,K,H);
true

```

Every edge of H is unitargeting with respect to f .

We obtain that f is an acyclic fibration, in particular a quasiisomorphism, since it satisfies the sufficient condition of Proposition 210:

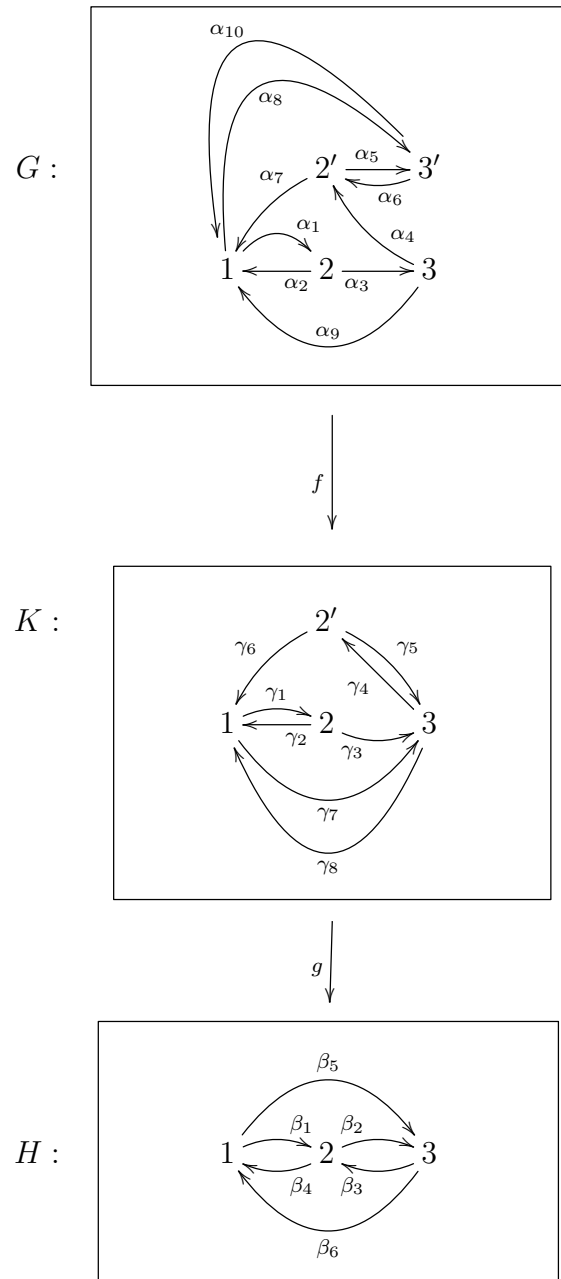
```

> SuffCond(f,K,H);
<true, true>

```

We extend Example 220 to a commutative triangle consisting of three quasiisomorphisms in Example 221.

Example 221 We consider the following graph morphisms.



Here, the graph morphisms $f : G \rightarrow K$ and $g : K \rightarrow H$ map the vertices and the edges in a vertical way, where $2' V_f = 2'$ and $2 V_f = 2$.

We will verify that the graph morphisms f , g and fg are etale fibrations that satisfy (Uni).

This will show that f , g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `Trygraph` by setting $n := 3$.

The graph K can be obtained with the function `idTrygraph` by setting $n := 3$.

The graph H can be obtained with the function `Doublecyclic` by setting $n := 3$.

The graph morphism f can be obtained with the function `Tryfactorization` by setting $n := 3$.

The graph morphism g can be obtained with the function `idTryacyclic` by setting `n := 3`.

The graph morphism fg can be obtained with the function `Tryacyclic` by setting `n := 3`.

So letting

```
G := Trygraph(3);
K := idTrygraph(3);
H := Doublecyclic(3);
f := Tryfactorization(3);
g := idTryacyclic(3);
fg := Tryacyclic(3);
> G;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>, <5, 5, 4>,
                    <3, 6, 4>, <4, 7, 1>, <1, 8, 5>, <5, 9, 1>, <3, 10, 1> ]>
> K;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>, <3, 5, 4>,
                  <4, 7, 1>, <1, 8, 3>, <3, 9, 1> ]>
> H;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1>, <1, 5, 3>,
               <3, 6, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 2, 1>>, <<2, 3, 3>, <2, 3, 3>>, <<4, 4, 5>, <4, 4, 3>>,
      <<5, 5, 4>, <3, 5, 4>>, <<3, 6, 4>, <3, 5, 4>>, <<4, 7, 1>, <4, 7, 1>>,
      <<1, 8, 5>, <1, 8, 3>>, <<5, 9, 1>, <3, 9, 1>>, <<3, 10, 1>, <3, 9, 1>> ]>
> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 3>, <2, 2, 3>>,
      <<3, 5, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>>, <<1, 8, 3>, <1, 5, 3>>,
      <<3, 9, 1>, <3, 6, 1>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>,
      <<5, 5, 4>, <3, 3, 2>>, <<3, 6, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>>,
      <<1, 8, 5>, <1, 5, 3>>, <<5, 9, 1>, <3, 6, 1>>, <<3, 10, 1>, <3, 6, 1>> ]>
```

we get

```
> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
> SuffCond(g,K,H);
<true, true>
```

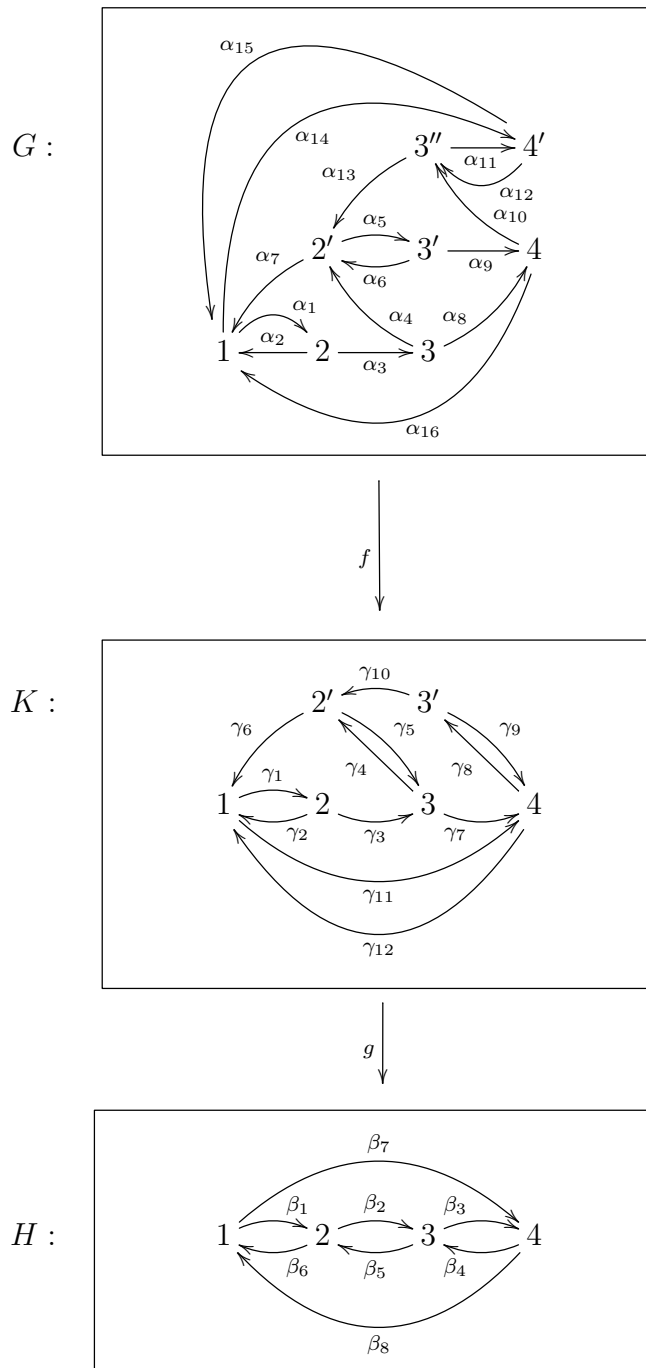
```

> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>

```

Varying the input n , we obtain further quasiisomorphisms; cf. Examples 222 and 223 below. This seems to hold independently of $n \geq 3$.

Example 222 We consider the following graph morphisms.



Here, the graph morphisms $f : G \rightarrow K$ and $g : K \rightarrow H$ map the vertices and the edges in a vertical way, where $2V_f = 2$, $2'V_f = 2'$, $3V_f = 3$, $3'V_f = 3$ and $3''V_f = 3'$.

We will verify that the graph morphisms f , g and fg are étale fibrations that satisfy (Uni).

This will show that f , g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `Trygraph` by setting `n := 4`.

The graph K can be obtained with the function `idTrygraph` by setting `n := 4`.

The graph H can be obtained with the function `Doublecyclic` by setting `n := 4`.

The graph morphism f can be obtained with the function `Tryfactorization` by setting `n := 4`.

The graph morphism g can be obtained with the function `idTryacyclic` by setting `n := 4`.

The graph morphism fg can be obtained with the function `Tryacyclic` by setting `n := 4`.

So letting

```
G := Trygraph(4);
K := idTrygraph(4);
H := Doublecyclic(4);
f := Tryfactorization(4);
g := idTryacyclic(4);
fg := Tryacyclic(4);
> G;
<[ 1, 2, 3, 4, 5, 6, 7, 8 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>,
  <5, 5, 4>, <5, 6, 6>, <7, 7, 8>, <8, 8, 7>, <3, 9, 4>, <4, 10, 1>, <3, 11, 6>,
  <6, 12, 7>, <7, 13, 4>, <1, 14, 8>, <8, 15, 1>, <6, 16, 1> ]>
> K;
<[ 1, 2, 3, 4, 6, 7 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>, <3, 5, 4>,
  <3, 6, 6>, <7, 7, 6>, <6, 8, 7>, <4, 10, 1>, <7, 13, 4>, <1, 14, 6>, <6, 15, 1> ]>
> H;
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>, <4, 4, 3>,
  <3, 5, 2>, <2, 6, 1>, <1, 7, 4>, <4, 8, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3>, <6, 6>, <7, 7>, <8, 6> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>>, <<2, 3, 3>, <2, 3, 3>>,
  <<4, 4, 5>, <4, 4, 3>>, <<5, 5, 4>, <3, 5, 4>>, <<5, 6, 6>, <3, 6, 6>>,
  <<7, 7, 8>, <7, 7, 6>>, <<8, 8, 7>, <6, 8, 7>>, <<3, 9, 4>, <3, 5, 4>>,
  <<4, 10, 1>, <4, 10, 1>>, <<3, 11, 6>, <3, 6, 6>>, <<6, 12, 7>, <6, 8, 7>>,
  <<7, 13, 4>, <7, 13, 4>>, <<1, 14, 8>, <1, 14, 6>>, <<8, 15, 1>, <6, 15, 1>>,
  <<6, 16, 1>, <6, 15, 1>> ]>
```

```

> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <6, 4>, <7, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 1>, <2, 6, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 3>, <2, 2, 3>>,
  <<3, 5, 4>, <3, 5, 2>>, <<3, 6, 6>, <3, 3, 4>>, <<7, 7, 6>, <3, 3, 4>>,
  <<6, 8, 7>, <4, 4, 3>>, <<4, 10, 1>, <2, 6, 1>>, <<7, 13, 4>, <3, 5, 2>>,
  <<1, 14, 6>, <1, 7, 4>>, <<6, 15, 1>, <4, 8, 1>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3>, <6, 4>, <7, 3>, <8, 4> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 6, 1>>, <<2, 3, 3>, <2, 2, 3>>,
  <<4, 4, 5>, <2, 2, 3>>, <<5, 5, 4>, <3, 5, 2>>, <<5, 6, 6>, <3, 3, 4>>,
  <<7, 7, 8>, <3, 3, 4>>, <<8, 8, 7>, <4, 4, 3>>, <<3, 9, 4>, <3, 5, 2>>,
  <<4, 10, 1>, <2, 6, 1>>, <<3, 11, 6>, <3, 3, 4>>, <<6, 12, 7>, <4, 4, 3>>,
  <<7, 13, 4>, <3, 5, 2>>, <<1, 14, 8>, <1, 7, 4>>, <<8, 15, 1>, <4, 8, 1>>,
  <<6, 16, 1>, <4, 8, 1>> ]>

```

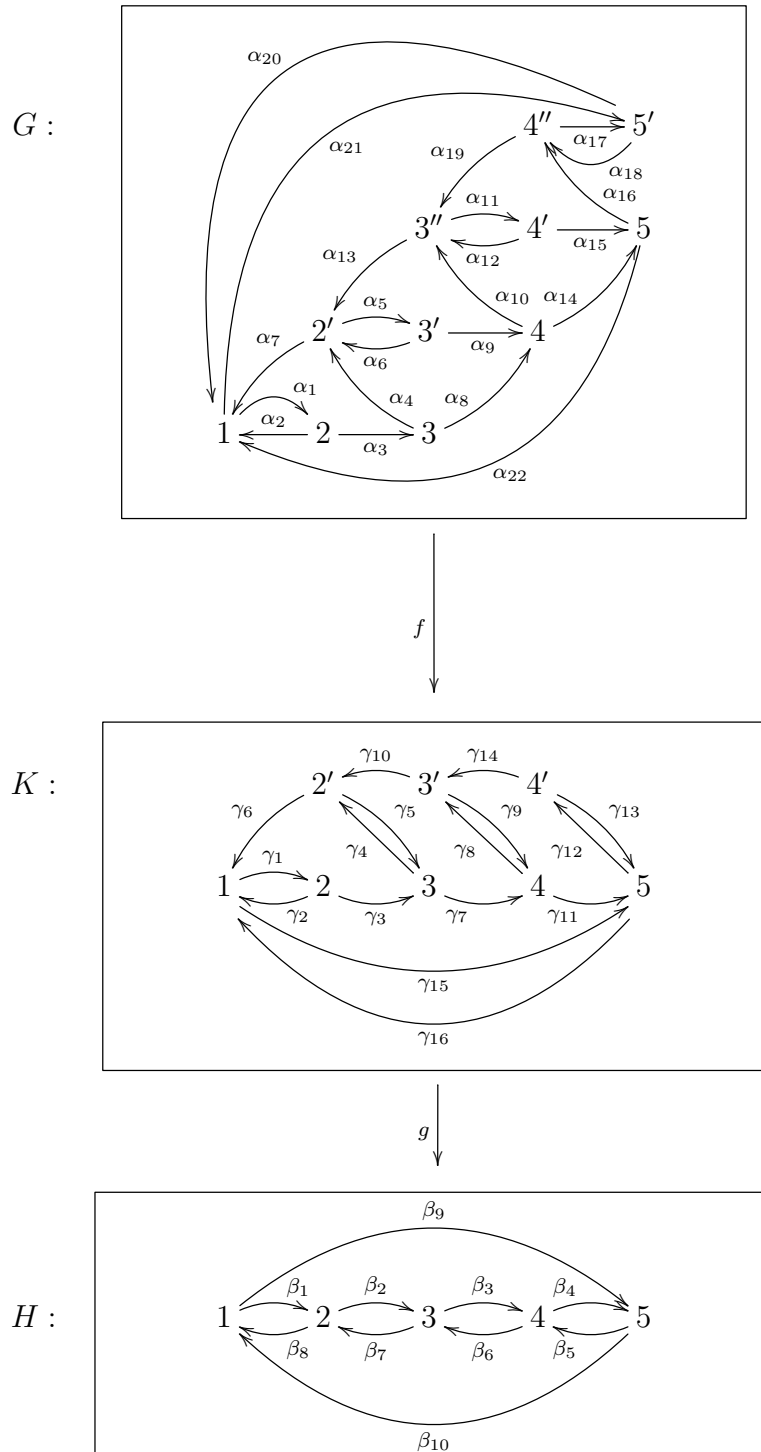
we get

```

> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
> SuffCond(g,K,H);
<true, true>
> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>

```

Example 223 We consider the following graph morphisms.



Here, the graph morphisms $f : G \rightarrow K$ and $g : K \rightarrow H$ map the vertices and the edges in a vertical way, where $2 V_f = 2$, $2' V_f = 2'$, $3 V_f = 3$, $3' V_f = 3$, $3'' V_f = 3'$, $4 V_f = 4$, $4' V_f = 4$ and $4'' V_f = 4'$.

We will verify that the graph morphisms f , g and fg are étale fibrations that satisfy (Uni).

This will show that f , g and fg are quasiisomorphisms; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph G can be obtained with the function `Trygraph` by setting `n := 5`.

The graph K can be obtained with the function `idTrygraph` by setting `n := 5`.

The graph H can be obtained with the function `Doublecyclic` by setting `n := 5`.

The graph morphism f can be obtained with the function `Tryfactorization` by setting `n := 5`.

The graph morphism g can be obtained with the function `idTryacyclic` by setting `n := 5`.

The graph morphism fg can be obtained with the function `Tryacyclic` by setting `n := 5`.

So letting

```
G := Trygraph(5);
K := idTrygraph(5);
H := Doublecyclic(5);
f := Tryfactorization(5);
g := idTryacyclic(5);
fg := Tryacyclic(5);
> G;
<[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>,
  <4, 4, 5>, <5, 5, 4>, <5, 6, 6>, <7, 7, 8>, <8, 8, 7>, <8, 9, 9>, <10, 10, 11>,
  <11, 11, 10>, <3, 12, 4>, <4, 13, 1>, <3, 14, 6>, <6, 15, 7>, <7, 16, 4>,
  <6, 17, 9>, <9, 18, 10>, <10, 19, 7>, <1, 20, 11>, <11, 21, 1>, <9, 22, 1> ]>
> K;
<[ 1, 2, 3, 4, 6, 7, 9, 10 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 3>,
  <3, 5, 4>, <3, 6, 6>, <7, 7, 6>, <6, 8, 7>, <6, 9, 9>, <10, 10, 9>,
  <9, 11, 10>, <4, 13, 1>, <7, 16, 4>, <10, 19, 7>, <1, 20, 9>, <9, 21, 1> ]>
> H;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>, <4, 4, 5>, <5, 5, 4>,
  <4, 6, 3>, <3, 7, 2>, <2, 8, 1>, <1, 9, 5>, <5, 10, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 3>, <6, 6>, <7, 7>, <8, 6>, <9, 9>,
  <10, 10>, <11, 9> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>>,
  <<2, 3, 3>, <2, 3, 3>>, <<4, 4, 5>, <4, 4, 3>>, <<5, 5, 4>, <3, 5, 4>>,
  <<5, 6, 6>, <3, 6, 6>>, <<7, 7, 8>, <7, 7, 6>>, <<8, 8, 7>, <6, 8, 7>>,
  <<8, 9, 9>, <6, 9, 9>>, <<10, 10, 11>, <10, 10, 9>>,
  <<11, 11, 10>, <9, 11, 10>>, <<3, 12, 4>, <3, 5, 4>>,
  <<4, 13, 1>, <4, 13, 1>>, <<3, 14, 6>, <3, 6, 6>>,
  <<6, 15, 7>, <6, 8, 7>>, <<7, 16, 4>, <7, 16, 4>>,
  <<6, 17, 9>, <6, 9, 9>>, <<9, 18, 10>, <9, 11, 10>>,
  <<10, 19, 7>, <10, 19, 7>>, <<1, 20, 11>, <1, 20, 9>>,
  <<11, 21, 1>, <9, 21, 1>>, <<9, 22, 1>, <9, 21, 1>> ]>
```



```

> g;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <6, 4>, <7, 3>, <9, 5>, <10, 4> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 8, 1>>, <<2, 3, 3>, <2, 2, 3>>,
    <<4, 4, 3>, <2, 2, 3>>, <<3, 5, 4>, <3, 7, 2>>, <<3, 6, 6>, <3, 3, 4>>,
    <<7, 7, 6>, <3, 3, 4>>, <<6, 8, 7>, <4, 6, 3>>, <<6, 9, 9>, <4, 4, 5>>,
    <<10, 10, 9>, <4, 4, 5>>, <<9, 11, 10>, <5, 5, 4>>,
    <<4, 13, 1>, <2, 8, 1>>, <<7, 16, 4>, <3, 7, 2>>,
    <<10, 19, 7>, <4, 6, 3>>, <<1, 20, 9>, <1, 9, 5>>,
    <<9, 21, 1>, <5, 10, 1>> ]>
> fg;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3>, <6, 4>, <7, 3>, <8, 4>, <9, 5>,
  <10, 4>, <11, 5> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 8, 1>>,
    <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>, <<5, 5, 4>, <3, 7, 2>>,
    <<5, 6, 6>, <3, 3, 4>>, <<7, 7, 8>, <3, 3, 4>>, <<8, 8, 7>, <4, 6, 3>>,
    <<8, 9, 9>, <4, 4, 5>>, <<10, 10, 11>, <4, 4, 5>>, <<11, 11, 10>, <5, 5, 4>>,
    <<3, 12, 4>, <3, 7, 2>>, <<4, 13, 1>, <2, 8, 1>>,
    <<3, 14, 6>, <3, 3, 4>>, <<6, 15, 7>, <4, 6, 3>>,
    <<7, 16, 4>, <3, 7, 2>>, <<6, 17, 9>, <4, 4, 5>>,
    <<9, 18, 10>, <5, 5, 4>>, <<10, 19, 7>, <4, 6, 3>>,
    <<1, 20, 11>, <1, 9, 5>>, <<11, 21, 1>, <5, 10, 1>>,
    <<9, 22, 1>, <5, 10, 1>> ]>

```

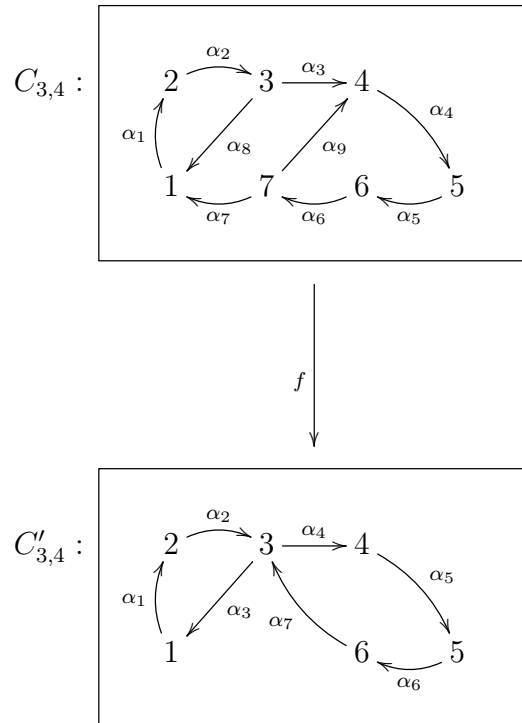
we get

```

> IsGraphMorphism(f,G,K);
true
> SuffCond(f,G,K);
<true, true>
> IsGraphMorphism(g,K,H);
true
> SuffCond(g,K,H);
<true, true>
> IsEqual(fg,ComposeGraphMorphisms(f,g));
true
> IsGraphMorphism(fg,G,H);
true
> SuffCond(fg,G,H);
<true, true>

```

Example 224 We consider the following graph morphism.



Here, the graph morphism $f : C_{3,4} \rightarrow C'_{3,4}$ maps the vertices and the edges in a vertical way, where $1 V_f = 1$, $2 V_f = 2$, $4 V_f = 4$ and $6 V_f = 6$.

We will verify that the graph morphism f is an étale fibration that satisfies (Uni).

This will show that f is a quasiisomorphism; cf. Proposition 210.

Via Magma we may proceed as follows, using the functions given in §10 below.

The graph $C_{3,4}$ can be obtained with the function `CnCm` by setting `n := 3` and `m := 4`.

The graph $C'_{3,4}$ can be obtained with the function `cncm` by setting `n := 3` and `m := 4`.

The graph morphism f can be obtained with the function `cncmqis` by setting `n := 3` and `m := 4`.

So letting

```
C34 := CnCm(3,4);
Cp34 := cncm(3,4); // "C prime" = C'
f := cncmqis(3,4);
> C34;
<[ 1, 2, 3, 4, 5, 6, 7 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>, <4, 4, 5>,
  <5, 5, 6>, <6, 6, 7>, <7, 7, 1>, <3, 8, 1>, <7, 9, 4> ]>
> Cp34;
<[ 1, 2, 3, 4, 5, 6 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 1>, <3, 4, 4>, <4, 5, 5>,
  <5, 6, 6>, <6, 7, 3> ]>
```

```

> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 5>, <6, 6>, <7, 3> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 3>, <2, 2, 3>>, <<3, 3, 4>, <3, 4, 4>>,
    <<4, 4, 5>, <4, 5, 5>>, <<5, 5, 6>, <5, 6, 6>>, <<6, 6, 7>, <6, 7, 3>>,
    <<7, 7, 1>, <3, 3, 1>>, <<3, 8, 1>, <3, 3, 1>>, <<7, 9, 4>, <3, 4, 4>> ]>

```

we get

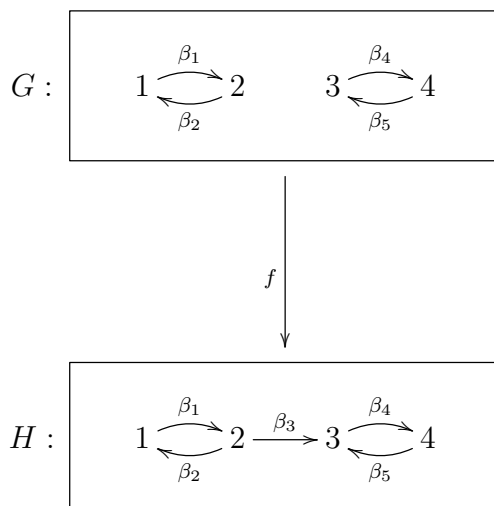
```

> IsGraphMorphism(f,C34,Cp34);
true
> SuffCond(f,C34,Cp34);
<true, true>

```

Varying the input n and m , it is possible to construct further quasiisomorphisms. This seems to hold independently of $n \geq 2$.

Example 225 We consider the following inclusion morphism f of the subgraph G into the graph H .



Then the graph morphism f is a quasiisomorphism.

Via Magma, we calculated that the graph morphism f satisfies (Uni) as follows.

```

G := <[1,2,3,4], [<1,1,2>,<2,2,1>,<3,4,4>,<4,5,3>]>;
H := <[1,2,3,4], [<1,1,2>,<2,2,1>,<2,3,3>,<3,4,4>,<4,5,3>]>;
f := IsSubgraph(G,H)[1];
> IsGraphMorphism(f,G,H);
true
> Uni(f,G,H);
true
> IsEtaleFibration(f,G,H);
false

```

But we can not apply Proposition 210 because f is not an etale fibration.

Proof. Suppose given the following diagram in Gph .

$$\begin{array}{ccc} & & G \\ & & \downarrow f \\ C_n & \xrightarrow{p} & H \end{array}$$

We have to show that there exists a unique graph morphism $q : C_n \rightarrow G$ such that $qf = p$.

Uniqueness. The graph morphism $q : C_n \rightarrow G$ is unique since the graph morphism $f : G \rightarrow H$ is injective, whence $(C_n, f)_{\text{Gph}}$ is injective.

Existence. In order to be able to let $q := p|_G$, we have to show that $C_n p \subseteq G$. I.e. we have to show that $\beta_3 \notin E_{C_n} E_p$.

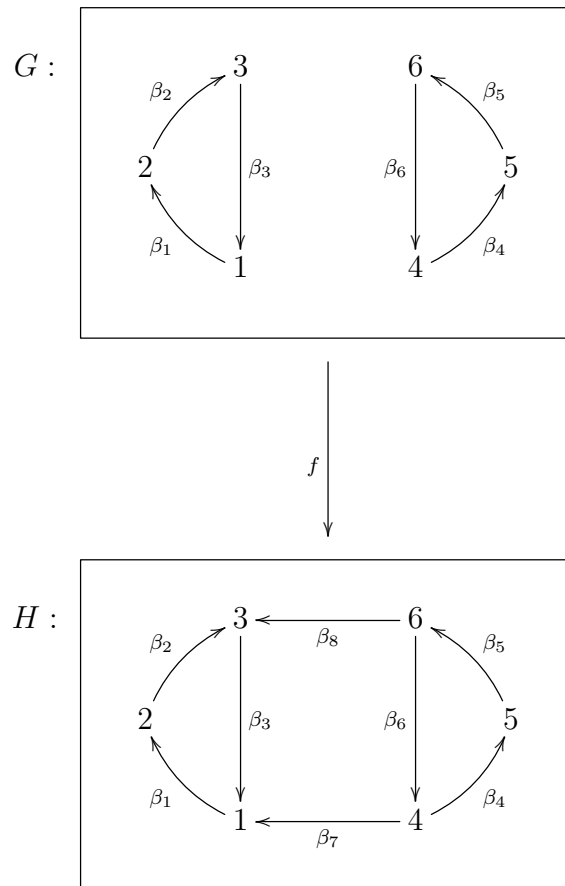
We assume that $\beta_3 \in E_{C_n} E_p$.

So there exists an edge $e_j \in E_{C_n}$ such that $e_j E_p = \beta_3$.

Thus $e_{j+1} E_p = \beta_4$. Thus $e_{j+2} E_p = \beta_5$. Thus $e_{j+3} E_p = \beta_4$. Etc.

We deduce that $e_i E_p \in \{\beta_4, \beta_5\}$ for $i \in \mathbb{Z}/n\mathbb{Z}$, contradicting $e_j E_p = \beta_3$. □

Example 226 We consider the following inclusion morphism f of the subgraph G into the graph H .



Then the graph morphism f is a quasiisomorphism.

Both G and H are fibrant.

The graph morphism f is not a fibration.

Via Magma we calculated that the graph morphism f satisfies (Uni), using the functions given in §10 below.

But we can not apply Proposition 210 because f is not an etale fibration.

With the function CNCN we calculated the graphs G and H and the graph morphism f .

So letting

```
G := CNCN(3)[1];
H := CNCN(3)[2];
f := CNCN(3)[3];
```

we get

```
> IsSubgraph(G,H)[2];
true
> IsGraphMorphism(f,G,H);
true
> Uni(f,G,H);
true
> IsEtaleFibration(f,G,H);
false
```

Proof. Suppose given the following diagram in Gph.

$$\begin{array}{ccc} & & G \\ & & \downarrow f \\ C_n & \xrightarrow{p} & H \end{array}$$

We have to show that there exists a unique graph morphism $q : C_n \rightarrow G$ such that $qf = p$.

Uniqueness. The graph morphism $q : C_n \rightarrow G$ is unique since the graph morphism $f : G \rightarrow H$ is injective, whence $(C_n, f)_{\text{Gph}}$ is injective

Existence. In order to be able to let $q := p|_G$, we have to show that $C_n p \subseteq G$. I.e. we have to show that $\beta_7, \beta_8 \notin E_{C_n} E_p$.

We show that $\beta_7 \notin C_n p$, the proof for β_8 being similar.

We assume that $\beta_7 \in E_{C_n} E_p$.

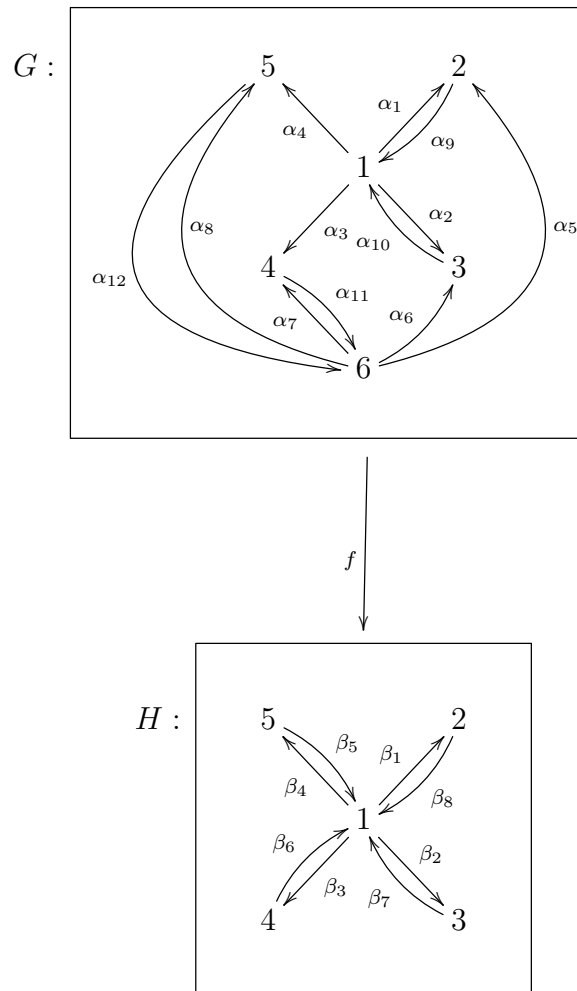
So there exists an edge $e_j \in E_{C_n}$ such that $e_j E_p = \beta_7$.

Thus $e_{j+1} E_p = \beta_1$. Thus $e_{j+2} E_p = \beta_2$. Thus $e_{j+3} E_p = \beta_3$. Thus $e_{j+4} E_p = \beta_1$. Etc.

We deduce that $e_i E_p \in \{\beta_1, \beta_2, \beta_3\}$ for $i \in \mathbb{Z}/n\mathbb{Z}$, contradicting $e_j E_p = \beta_7$. \square

Varying the input n , it is possible to construct further quasiisomorphisms.

Example 227 We consider the following graphs.



We consider the graph morphism $f : G \rightarrow H$ with

$$\begin{aligned} 1 V_f &:= 1, & 2 V_f &:= 2, & 3 V_f &:= 3, \\ 4 V_f &:= 4, & 5 V_f &:= 5, & 6 V_f &:= 1; \end{aligned}$$

and with the corresponding map on the edges.

The graph morphism $f : G \rightarrow H$ is an acyclic fibration.

Proof. Via Magma we will calculate that the graph morphism f is an etale fibration that satisfies (Uni), using the functions given in §10 below.

This will show that f is a quasiisomorphism; cf. Proposition 210.

The graph morphism f together with the graphs G and H can be obtained with Magma with the function `Exflower` by setting `n := 4` and `list := [2,3]`.

So letting

```
G := Exflower(4,[2,3])[1];
H := Exflower(4,[2,3])[2];
f := Exflower(4,[2,3])[3];
> G;
<[ 1, 2, 3, 4, 5, 6 ], [ <1, 1, 2>, <1, 2, 3>, <1, 3, 4>, <1, 4, 5>, <6, 5, 2>,
  <6, 6, 3>, <6, 7, 4>, <6, 8, 5>, <2, 9, 1>, <3, 10, 1>, <4, 11, 6>,
  <5, 12, 6> ]>
> H;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <1, 2, 3>, <1, 3, 4>, <1, 4, 5>, <5, 5, 1>,
  <4, 6, 1>, <3, 7, 1>, <2, 8, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4>, <5, 5>, <6, 1> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<1, 2, 3>, <1, 2, 3>>, <<1, 3, 4>, <1, 3, 4>>, <<1, 4, 5>, <1, 4, 5>>,
  <<6, 5, 2>, <1, 1, 2>>, <<6, 6, 3>, <1, 2, 3>>, <<6, 7, 4>, <1, 3, 4>>,
  <<6, 8, 5>, <1, 4, 5>>, <<2, 9, 1>, <2, 8, 1>>, <<3, 10, 1>, <3, 7, 1>>,
  <<4, 11, 6>, <4, 6, 1>>, <<5, 12, 6>, <5, 5, 1>> ]>
```

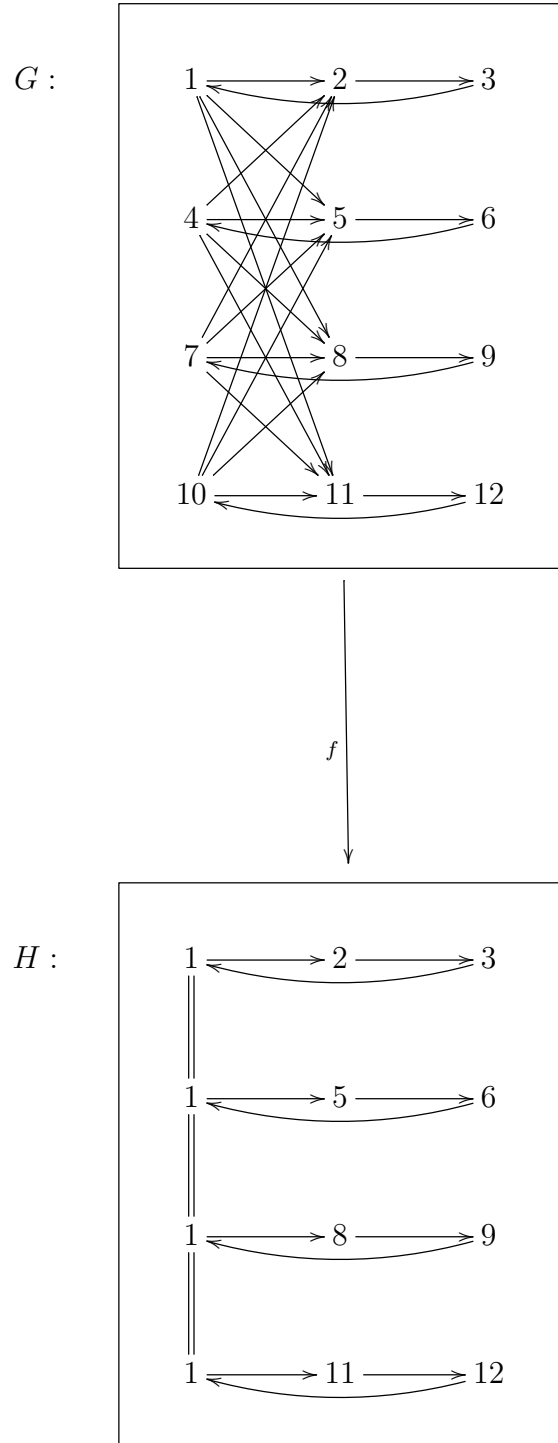
we get

```
> IsGraphMorphism(f,G,H);
true
> SuffCond(f,G,H);
<true, true>
```

By varying the input `n` and `list` it is possible to construct further quasiisomorphisms using `Exflower(n,list)`.

The “lower” graph consists of 4 respectively n cyclic graphs C_2 glued together at one vertex such that the form of a flower is visible.

Example 228 We consider the graph morphism



where we let

$$\begin{aligned}
 1V_f &:= 1 & 2V_f &:= 2 & 3V_f &:= 3 & 4V_f &:= 1 & 5V_f &:= 5 & 6V_f &:= 6 \\
 7V_f &:= 1 & 8V_f &:= 8 & 9V_f &:= 9 & 10V_f &:= 1 & 11V_f &:= 11 & 12V_f &:= 12 .
 \end{aligned}$$

Here, the graph H has only a single vertex named 1, displayed four times for sake of clarity.

The graph morphism $f : G \rightarrow H$ is an acyclic fibration.

Proof. Via Magma we will calculate that the graph morphism f is an étale fibration that satisfies (Uni), using the functions given in §10 below.

This will show that f is a quasiisomorphism; cf. Proposition 210.

The graph morphism f together with the graphs G and H can be obtained with Magma with the function `Exflower2` by setting $n := 4$ and $k := 3$.

So letting

```
G := Exflower2(4,3)[1];
H := Exflower2(4,3)[2];
f := Exflower2(4,3)[3];
> G;
<[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 1>,
  <4, 4, 5>, <5, 5, 6>, <6, 6, 4>, <7, 7, 8>, <8, 8, 9>, <9, 9, 7>,
  <10, 10, 11>, <11, 11, 12>, <12, 12, 10>, <1, 13, 5>, <1, 14, 8>, <1, 15, 11>,
  <4, 16, 2>, <4, 17, 8>, <4, 18, 11>, <7, 19, 2>, <7, 20, 5>, <7, 21, 11>,
  <10, 22, 2>, <10, 23, 5>, <10, 24, 8> ]>
> H;
<[ 1, 2, 3, 4, 5, 6, 7, 8, 9 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 1>, <1, 4, 4>,
  <4, 5, 5>, <5, 6, 1>, <1, 7, 6>, <6, 8, 7>, <7, 9, 1>, <1, 10, 8>, <8, 11, 9>,
  <9, 12, 1> ]>
> f;
<[ <1, 1>, <4, 1>, <7, 1>, <10, 1>, <2, 2>, <3, 3>, <5, 4>, <6, 5>, <8, 6>,
  <9, 7>, <11, 8>, <12, 9> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 3>, <2, 2, 3>>,
  <<3, 3, 1>, <3, 3, 1>>, <<4, 4, 5>, <1, 4, 4>>, <<5, 5, 6>, <4, 5, 5>>,
  <<6, 6, 4>, <5, 6, 1>>, <<7, 7, 8>, <1, 7, 6>>, <<8, 8, 9>, <6, 8, 7>>,
  <<9, 9, 7>, <7, 9, 1>>, <<10, 10, 11>, <1, 10, 8>>, <<11, 11, 12>, <8, 11, 9>>,
  <<12, 12, 10>, <9, 12, 1>>, <<1, 13, 5>, <1, 4, 4>>, <<1, 14, 8>, <1, 7, 6>>,
  <<1, 15, 11>, <1, 10, 8>>, <<4, 16, 2>, <1, 1, 2>>, <<4, 17, 8>, <1, 7, 6>>,
  <<4, 18, 11>, <1, 10, 8>>, <<7, 19, 2>, <1, 1, 2>>, <<7, 20, 5>, <1, 4, 4>>,
  <<7, 21, 11>, <1, 10, 8>>, <<10, 22, 2>, <1, 1, 2>>, <<10, 23, 5>, <1, 4, 4>>,
  <<10, 24, 8>, <1, 7, 6>> ]>
```

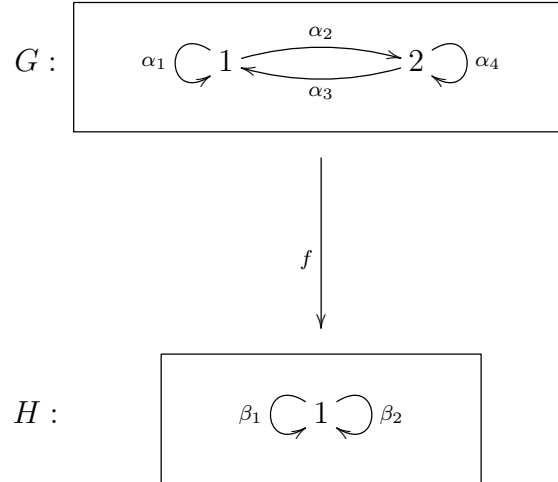
we get

```
> SuffCond(f,G,H);
<true, true>
```

Note that we need to have edges from the vertices 1,4,7 and 10 to the vertices 2,5,8 and 11 to obtain an étale fibration.

Via Magma, we calculated that we obtain quasiisomorphisms with `Exflower2` for input values n in $[1, 30]$ and k in $[2, 30]$.

Example 229 We consider the graph morphism



where

$$\begin{aligned}
 1V_f &= 1, & 2V_f &= 1, \\
 \alpha_1 E_f &= \beta_1, & \alpha_3 E_f &= \beta_1, \\
 \alpha_2 E_f &= \beta_2, & \alpha_4 E_f &= \beta_2.
 \end{aligned}$$

Via Magma we will calculate that the graph morphism f is an etale fibrations that satisfies (Uni), using the functions given in §10 below.

Letting

```

G := <[1,2], [<1,1,1>, <1,2,2>, <2,3,1>, <2,4,2>]>;
H := <[1], [<1,1,1>, <1,2,1>]>;
f := <[<1,1>, <2,1>], [<<1,1,1>, <1,1,1>>, <<1,2,2>, <1,2,1>>, <<2,3,1>, <1,1,1>>,
      <<2,4,2>, <1,2,1>>]>;

```

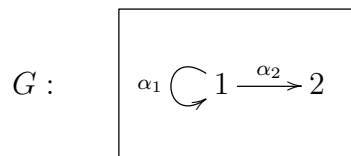
we get

```

> SuffCond(f,G,H);
<true, true>

```

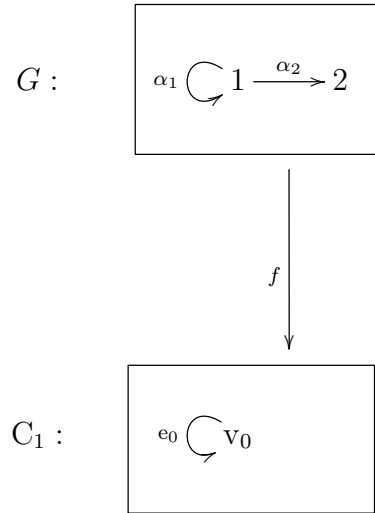
Example 230 We consider the graph



and the cyclic graph C_1 ; cf. Definition 52.

We consider the graph morphism $f : G \rightarrow C_1$ with

$$\begin{aligned}
 1V_f &= v_0, & \alpha_1 E_f &= e_0, \\
 2V_f &= v_0, & \alpha_2 E_f &= e_0.
 \end{aligned}$$



We will calculate that the graph G and the cyclic graph C_1 are thin and that the graph morphism $f : G \rightarrow C_1$ does **not** satisfy (Uni) and f is **not** a fibration; cf. Definitions 52 and 127.(1) and Proposition 210. But the graph morphism $f : G \rightarrow C_1$ is a quasiisomorphism; cf. Definition 115.

Via Magma we may proceed as follows, using the functions given in §10 below.

```

G := <[1,2], [<1,1,1>, <1,2,2>]>;
> IsThin(G);
true
C1 := C(1);
> IsThin(C1);
true
f := ListGraphMorphisms(G,C1)[1];
> f;
<[ <1, 1>, <2, 1> ], [ <<1, 1, 1>, <1, 1, 1>>, <<1, 2, 2>, <1, 1, 1>> ]>
> Uni(f,G,C1);
false
> IsFibration(f,G,C1);
false

```

9.2 Some examples of graph morphisms related to the sufficient condition of Proposition 210

Example 231 We consider the graph morphism $f : \emptyset \rightarrow C_1$.

Via Magma, we calculated that the empty graph \emptyset and the cyclic graph $C_1 =: H$ are thin and that the graph morphism $f : \emptyset \rightarrow C_1$ is an étale fibration; cf. Definitions 70, 52 and 127.(2). But the graph morphism f does **not** satisfy (Uni); cf. Proposition 210. Moreover, f is **not** a quasiisomorphism, since $(C_1, \emptyset)_{\text{Gph}} = \emptyset \neq (C_1, C_1)_{\text{Gph}}$.

```

G := <[], []>;
H := C(1);
f := <[], []>;
> IsThin(G);
true
> IsThin(H);
true
> IsEtaleFibration(f,G,H);
true
> Uni(f,G,H);
false
> IsQis_Bound(f,G,H,1);
false

```

Example 232 We consider the graph morphism $f : D_1 \rightarrow C_1$.

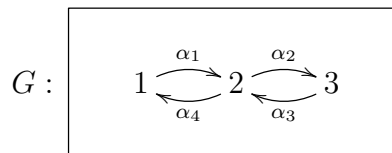
Via Magma, we calculated that the direct graph D_1 and the cyclic graph C_1 are thin and that the graph morphism $f : D_1 \rightarrow C_1$ satisfies (Uni); cf. Definitions 56 and 52 and Proposition 210. But the graph morphism f is **not** a fibration; cf. Definition 127.(1). Moreover, f is **not** a quasiisomorphism, since $(C_1, D_1)_{\text{Gph}} = \emptyset \neq (C_1, C_1)_{\text{Gph}}$.

```

G := D(1);
> IsThin(G);
true
H := C(1);
> IsThin(H);
true
> f := ListGraphMorphisms(G,H)[1];
> f;
<[ <0, 1>, <1, 1> ], [ <<0, 0, 1>, <1, 1, 1>> ]>
> Uni(f,G,H);
true
> IsFibration(f,G,H);
false

```

Example 233 We consider the graphs



and C_2 .

Let $f : G \rightarrow C_2$ be the graph morphism with

$$\begin{aligned}
1V_f &= v_1, & 2V_f &= v_2, & 3V_f &= v_1, \\
\alpha_1 E_f &= e_1, & \alpha_2 E_f &= e_2, & \alpha_3 E_f &= e_1, & \alpha_4 E_f &= e_2.
\end{aligned}$$

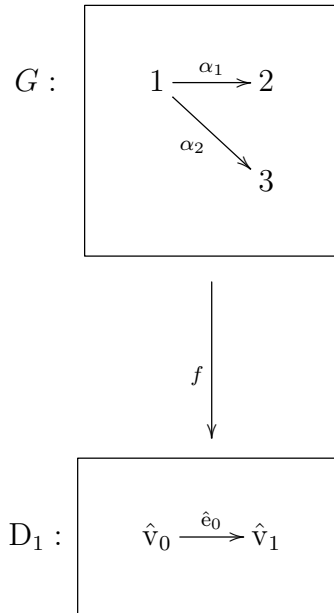
Via Magma, we calculated that the graph G and the cyclic graph C_2 are thin and that the graph morphism $f : G \rightarrow C_2$ satisfies (Uni); cf. Definition 52 and Proposition 210. Moreover, f is a fibration; cf. Definition 127.(1). But the graph morphism f is **not** an etale fibration; cf. Definition 127.(2). And f is **not** a quasiisomorphism since $|(C_2, G)_{\text{Gph}}| = 4 \neq 2 = |(C_2, C_2)_{\text{Gph}}|$; cf. Definition 115.

```

G := c2chain(3);
> G;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>
> IsThin(G);
true
H := C(2);
> H;
<[ 1, 2 ], [ <1, 1, 2>, <2, 2, 1> ]>
> IsThin(H);
true
> f := ListGraphMorphisms(G,H) [2];
> f;
<[ <1, 1>, <2, 2>, <3, 1> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 3>, <2, 2, 1>>,
    <<3, 3, 2>, <1, 1, 2>>, <<2, 4, 1>, <2, 2, 1>> ]>
> Uni(f,G,H);
true
> IsFibration(f,G,H);
true
> IsEtaleFibration(f,G,H);
false
> IsQis_Bound(f,G,H,2);
false

```

Example 234 We consider the following graph morphism, mapping the vertices and the edges in a vertical way.



Via Magma, we calculated that the graph morphism $f : G \rightarrow D_1$ satisfies (Uni) and that f is a fibration; cf. Proposition 210 and Definition 127.(1). But the graph morphism f is not an etale fibration; cf. Definition 127.(2). However, f is a quasiisomorphism, since $(C_n, G)_{\text{Gph}} = \emptyset = (C_n, D_1)_{\text{Gph}}$ for $n \in \mathbb{N}$.

```

G := <[0,1,2], [<0,1,1>, <0,2,2>]>;
H := D(1);
f := ListGraphMorphisms(G,H)[1];
> f;
<[ <0, 0>, <1, 1>, <2, 1> ], [ <<0, 1, 1>, <0, 0, 1>>, <<0, 2, 2>, <0, 0, 1>> ]>
> IsFibration(f,G,H);
true
> Uni(f,G,H);
true
> IsEtaleFibration(f,G,H);
false

```

So the converse to Proposition 210 does not hold, even for fibrations satisfying (Uni).

Example 235 Suppose given $n \in \mathbb{N}$.

We consider the graph morphism $f := \tau_{C_1 \sqcup C_n} : C_1 \sqcup C_n \rightarrow C_1$.

The graph morphism f is an etale fibration. The edge e_0 of the target C_1 is not unitargeting. In fact, the edges mapping to e_0 have $n + 1 \geq 2$ distinct targets.

The graph morphism f can be calculated via Magma with the function `exampleforbadbound2` given in §10.8.

E.g., setting $n := 3$, we get

```
> exampleforbadbound2(3);
<<[ 1, 2, 3, 4 ], [ <1, 1, 1>, <2, 2, 3>, <3, 3, 4>, <4, 4, 2> ]>,
  <[ 1 ], [ <1, 1, 1> ]>, <[ <1, 1>, <2, 1>, <3, 1>, <4, 1> ],
  [ <<1, 1, 1>, <1, 1, 1>>, <<2, 2, 3>, <1, 1, 1>>, <<3, 3, 4>,
    <1, 1, 1>>, <<4, 4, 2>, <1, 1, 1>> ]>>
```

The map $(C_k, f)_{\text{Gph}}$ is bijective for $1 \leq k < n$ since there are unique graph morphisms $C_k \rightarrow C_1 \sqcup C_n$ and $C_k \rightarrow C_1$.

But the map $(C_n, f)_{\text{Gph}}$ is not bijective since we may have $p, q : C_n \rightarrow C_1 \sqcup C_n$ with $C_n p = C_1$ and $C_n q = C_n$.

In particular, f is not a quasiisomorphism.

E.g., we get

```
f := exampleforbadbound2(3)[3];
G := exampleforbadbound2(3)[1];
> IsQis_Bound(f,G,C(1),2);
true
> IsQis_Bound(f,G,C(1),3);
false
```

and

```
f := exampleforbadbound2(10)[3];
G := exampleforbadbound2(10)[1];
> IsQis_Bound(f,G,C(1),9);
true
> IsQis_Bound(f,G,C(1),10);
false
```

9.3 Some inequalities of subsets of $\text{Mor}(\text{Gph})$

We consider subsets of $\text{Mor}(\text{Gph})$.

Remark 236 We have $\text{AcFib} \subsetneq \text{Fib}$.

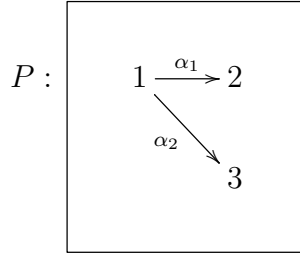
Proof. We consider the graph morphism $\iota_1 : D_0 \rightarrow D_0 \sqcup C_1$.

Note that the map $E_{\iota_1, \hat{v}_0} : \emptyset \rightarrow \emptyset$ is surjective. Thus the graph morphism $\iota_1 : D_0 \rightarrow D_0 \sqcup C_1$ is a fibration; cf. Definition 127.(1). But ι_1 is not a quasiisomorphism; cf. Definitions 115 and 127.(1).

So $\iota_1 \in \text{Fib} \setminus \text{AcFib}$. □

Remark 237 We have $\text{AcFib} \subsetneq \text{Qis}$.

Proof. Let



We consider the acyclic cofibration $g : D_1 \rightarrow P$ mapping \hat{e}_1 to α_2 ; cf. Assertion 254 below.

The graph morphism $g : D_1 \rightarrow P$ is an acyclic cofibration and thus a quasiisomorphism; cf. Lemma 175. But g is not a fibration and thus not an acyclic fibration; cf. Definitions 115, 127.(1), 138.

```
> P := <[1,2,3], [<1,1,2>, <1,2,3>]>;
> g := VtoE(D(1), P, [<0,1>, <1,3>]);
> IsAcCofib(g, D(1), P);
true
```

So $g \in \text{Qis} \setminus \text{AcFib}$. □

Remark 238 We have $\text{AcCofib} \subsetneq \text{Cofib}$.

Proof. We consider the cofibration $\iota_{C_1} : \emptyset \rightarrow C_1$; cf. Remark 151.

The graph morphism $\iota_{C_1} : \emptyset \rightarrow C_1$ is not a quasiisomorphism; cf. Definition 115.

So $\iota_{C_1} \in \text{Cofib} \setminus \text{AcCofib}$. □

Remark 239 We have $\text{AcCofib} \subsetneq \text{Qis}$.

Proof. We consider the acyclic fibration $\iota_{D_0} : \emptyset \rightarrow D_0$; cf. Example 158.(1).

The graph morphism $\iota_{D_0} : \emptyset \rightarrow D_0$ is an acyclic fibration and thus a quasiisomorphism.

But ι_{D_0} is not a cofibration; cf. Example 158.(2). Thus it is not an acyclic cofibration; cf. Lemma 175.

Alternatively, the graph morphism $\iota_{D_0} : \emptyset \rightarrow D_0$ is not an acyclic cofibration because it does not satisfy (AcCofib 3) since $V_{D_0} \setminus V_\emptyset \neq \emptyset$, but $E_{D_0} = \emptyset$.

So $\iota_{D_0} \in \text{Qis} \setminus \text{AcCofib}$. □

Remark 240 We have $\square \text{Qis} \subsetneq \square \text{AcFib} \stackrel{\text{Def. 144}}{=} \text{Cofib}$.

In fact, we have $\text{AcCofib} \setminus \square \text{Qis} \neq \emptyset$.

Proof. We consider the graph morphism $\iota_{0,1} : D_0 \rightarrow D_1$; cf. Definition 56. It is an acyclic cofibration; cf. Remark 166. Thus it is a cofibration and a quasiisomorphism; cf. Lemma 175.

We consider the following commutative quadrangle.

$$\begin{array}{ccc} D_0 & \xrightarrow{\text{id}_{D_0}} & D_0 \\ \iota_{0,1} \downarrow & & \downarrow \cong \iota_{0,1} \\ D_1 & \xrightarrow{\text{id}_{D_1}} & D_1 \end{array}$$

Since $E_{D_1} = \{\hat{e}_0\}$ and $E_{D_0} = \emptyset$, there does not exist a graph morphism $c : D_1 \rightarrow D_0$.

$$\begin{array}{ccc} D_0 & \xrightarrow{\text{id}_{D_0}} & D_0 \\ \iota_{0,1} \downarrow & \nearrow \# & \downarrow \cong \iota_{0,1} \\ D_1 & \xrightarrow{\text{id}_{D_1}} & D_1 \end{array}$$

So the graph morphism $\iota_{0,1} : D_0 \rightarrow D_1$ is in $\text{AcCofib} \setminus \square \text{Qis}$. □

Note that we also have $\square \text{Fib} \subsetneq \square \text{AcFib}$ since

$$\square \text{Fib} \stackrel{\text{Lemma 191}}{=} \text{AcCofib} \stackrel{\text{Remark 238}}{\subsetneq} \text{Cofib} \stackrel{\text{Definition 144}}{=} \square \text{AcFib} .$$

Remark 241 We have $\square \text{Qis} \setminus \text{AcCofib} \stackrel{\text{Lemma 191}}{=} \square \text{Qis} \setminus \square \text{Fib} \neq \emptyset$.

Proof. We consider the graph morphism $f := \iota_{C_2} : \emptyset \rightarrow C_2$.

First, we show that f is in $\square \text{Qis}$.

Suppose given a commutative diagram as follows.

$$\begin{array}{ccc} \emptyset & \xrightarrow{p} & G \\ f \downarrow & & \downarrow \cong g \\ C_2 & \xrightarrow{q} & H \end{array}$$

Since the graph morphism $g : G \rightarrow H$ is a quasiisomorphism, there exists a graph morphism $h : C_2 \rightarrow G$ such that $hg = q$.

Thus we have the following commutative diagram.

$$\begin{array}{ccc} \emptyset & \xrightarrow{p} & G \\ f \downarrow & \nearrow h & \downarrow \cong g \\ C_2 & \xrightarrow{q} & H \end{array}$$

So the graph morphism $f : \emptyset \rightarrow C_2$ is in $\square \text{Qis}$.

In particular, f is a cofibration; cf. Remark 195.(3) and Definition 138, or Remark 151.

Secondly, we show that f is not in $\square \text{Fib}$.

We consider the following commutative diagram.

$$\begin{array}{ccc} \emptyset & \xrightarrow{p} & C_4 \\ f \downarrow & & \downarrow \tau_{C_4} \\ C_2 & \xrightarrow{\tau_{C_2}} & C_1 \end{array}$$

Note that the graph morphism $\tau_{C_4} : C_4 \rightarrow C_1$ is a fibration; cf. Example 137.

But there does not exist a graph morphism $h : C_2 \rightarrow C_4$ at all.

So f is not in $\square \text{Fib}$.

Alternatively, using $\square \text{Fib} \stackrel{\text{Lemma 191}}{=} \text{AcCofib}$, f is not an acyclic cofibration since it does not satisfy (AcCofib 5).

So $f \in \square \text{Qis} \setminus \square \text{Fib}$. □

Remark 242 We have $\square \text{Fib} \setminus \square \text{Qis} \stackrel{\text{Lemma 191}}{=} \text{AcCofib} \setminus \square \text{Qis} \neq \emptyset$.

Proof. We consider the acyclic cofibration $f := \iota_{0,1} : D_0 \rightarrow D_1$; cf. Remark 166.

We consider the following commutative diagram.

$$\begin{array}{ccc} D_0 & \xrightarrow{\text{id}_{D_0}} & D_0 \\ f \downarrow & & \downarrow \cong f \\ D_1 & \xrightarrow{\text{id}_{D_1}} & D_1 \end{array}$$

There does not exist a graph morphism $h : D_1 \rightarrow D_0$.

So f is not in $\square \text{Qis}$.

So $f \in \text{AcCofib} \setminus \square \text{Qis}$. □

Remark 243 We have $\text{Qis}^\square \subsetneq \text{AcCofib}^\square \stackrel{\text{Lemma 192}}{=} \text{Fib}$.

In fact, we have $\text{AcFib} \setminus \text{Qis}^\square \neq \emptyset$.

Proof. We consider the graph morphism $f := \iota_1 : D_0 \rightarrow D_0 \sqcup D_0$, where $\hat{v}_0 V_f = (1, \hat{v}_0)$.

The graph morphism $f : D_0 \rightarrow D_0 \sqcup D_0$ is a quasiisomorphism since $(C_n, D_0)_{\text{Gph}} = \emptyset = (C_n, D_0 \sqcup D_0)_{\text{Gph}}$.

The map $E_{f, \hat{v}_0} : \emptyset \rightarrow \emptyset$ is surjective. So the graph morphism $f : D_0 \rightarrow D_0 \sqcup D_0$ is a fibration.

So the graph morphism $f : D_0 \rightarrow D_0 \sqcup D_0$ is an acyclic fibration.

We consider the following commutative quadrangle.

$$\begin{array}{ccc} D_0 & \xrightarrow{\text{id}_{D_0}} & D_0 \\ f \downarrow & & \downarrow f \\ D_0 \sqcup D_0 & \xrightarrow{\text{id}_{D_0 \sqcup D_0}} & D_0 \sqcup D_0 \end{array}$$

There exists the unique graph morphism $c : D_0 \sqcup D_0 \rightarrow D_0$, having $(1, \hat{v}_0) V_c = (2, \hat{v}_0) V_c = \hat{v}_0$. But we do not have $cf = \text{id}_{D_0 \sqcup D_0}$ since $(2, \hat{v}_0) V_{cf} = (1, \hat{v}_0) \neq (2, \hat{v}_0) = (2, \hat{v}_0) V_{\text{id}_{D_0 \sqcup D_0}}$.

$$\begin{array}{ccc}
 D_0 & \xrightarrow{\text{id}_{D_0}} & D_0 \\
 \downarrow f \wr & \nearrow \# & \downarrow f \\
 D_0 \sqcup D_0 & \xrightarrow{\text{id}_{D_0 \sqcup D_0}} & D_0 \sqcup D_0
 \end{array}$$

So the graph morphism $f : D_0 \rightarrow D_0 \sqcup D_0$ is in $\text{AcFib} \setminus \text{Qis}^\square$. □

Note that we also have $\text{Cofib}^\square \subsetneq \text{AcCofib}^\square$ since

$$\text{Cofib}^\square \stackrel{\text{Lemma 193}}{=} \text{AcFib} \stackrel{\text{Remark 236}}{\subsetneq} \text{Fib} \stackrel{\text{Lemma 192}}{=} \text{AcCofib}^\square.$$

Remark 244 We have $\text{Cofib}^\square \setminus \text{Qis}^\square \stackrel{\text{Lemma 193}}{=} \text{AcFib} \setminus \text{Qis}^\square \neq \emptyset$.

Proof. We consider the acyclic fibration $f : \emptyset \rightarrow D_0$; cf. Example 158.(1).

We consider the following commutative diagram

$$\begin{array}{ccc}
 \emptyset & \longrightarrow & \emptyset \\
 \downarrow f \wr & & \downarrow f \\
 D_0 & \xrightarrow{\text{id}_{D_0}} & D_0
 \end{array}$$

There does not exist a graph morphism $h : D_0 \rightarrow \emptyset$.

So the graph morphism f is not in Qis^\square .

So $f \in \text{Cofib}^\square \setminus \text{Qis}^\square$. □

9.4 Counterexamples for model categories

Recall that Gph is a Quillen closed model category; cf. Proposition 204.

Recall that the label **Assertion** indicates an assertion that we **falsify** by a counterexample.

Some of the following assertions are dual to each other, leading to further possibilities to falsify them.

9.4.1 Elementary counterexamples

Assertion 245 *In a Quillen closed model category, each acyclic fibration is a retraction.*

This assertion is **false**.

Counterexample 1 in Gph.

We consider the acyclic fibration $f : \emptyset \rightarrow D_0$; cf. Example 158.

But f is not a retraction since there does not exist a graph morphism $g : D_0 \rightarrow \emptyset$.

Counterexample 2 in Gph.

Consider the graph morphism $f : G \rightarrow H$ as in Example 215.

The acyclic fibration f is not a retraction.

We assume that f is a retraction.

Let $g : H \rightarrow G$ be a corresponding coretraction, i.e. $g \cdot f = \text{id}_H$.

So $1V_g = 1$. Thus $\beta_1 E_g = \alpha_1$. So $2V_g = 2$. Thus $\beta_2 E_g = \alpha_3$. So $3V_g = 3$. Thus $\beta_3 E_g = \alpha_4$.
So $2V_g = 2'$.

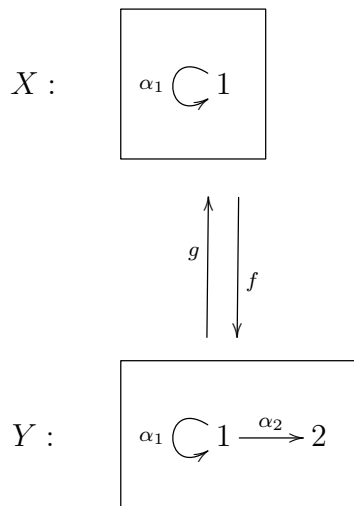
Contradiction. □

Assertion 246 *The subset of fibrations in a Quillen closed model category satisfies (2 of 3).*

This assertion is **false**.

Counterexample in Gph.

Consider the following graphs.



We consider the unique graph morphisms $f : X \rightarrow Y$ and $g := \tau_Y : Y \rightarrow X$, as well as their composite $fg = \text{id}_X : X \rightarrow X$.

The graph morphisms id_X and g are acyclic fibrations.

But f is not a fibration; cf. Definition 127.

```

X := C(1);
Y := <[1,2], [<1,1,1>, <1,2,2>]>;
f := VtoE(X,Y, [<1,1>]);
> f;
<[ <1, 1> ], [ <<1, 1, 1>, <1, 1, 1>> ]>
> IsFibration(f,X,Y);
false

```

□

Assertion 247 *In a Quillen closed model category, each acyclic cofibration is a coretraction.*

This assertion is **false**.

Counterexample in Gph.

We consider the acyclic cofibration $\iota_{0,1} : D_0 \rightarrow D_1$; cf. Remark 166.

But $\iota_{0,1}$ is not a coretraction since there does not exist a graph morphism $g : D_1 \rightarrow D_0$. □

Assertion 248

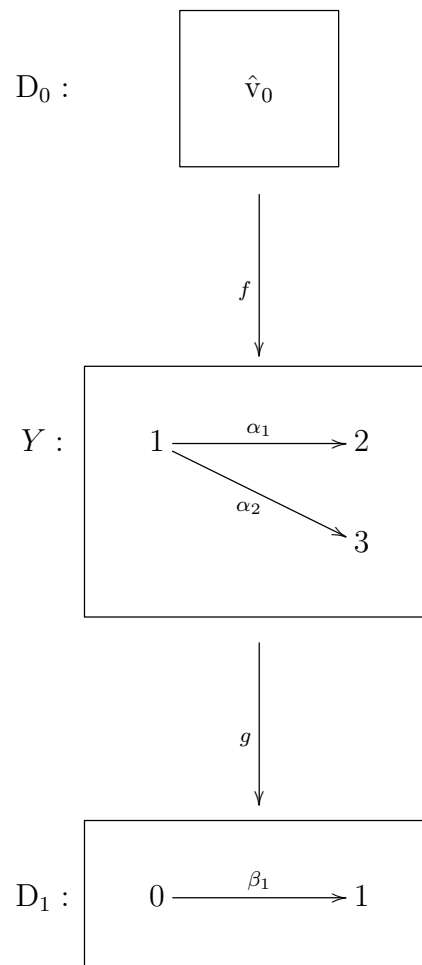
(1) *The subset of acyclic cofibrations in a Quillen closed model category satisfies (2 of 3).*

(2) *The subset of cofibrations in a Quillen closed model category satisfies (2 of 3).*

The assertions (1) and (2) are **false**.

Counterexample in Gph.

We consider the following graph morphisms.



We consider the graph morphisms $f : D_0 \rightarrow Y$, $g : Y \rightarrow D_1$ and $fg = \iota_1 : D_0 \rightarrow D_1$, where $\hat{v}_0 V_f := 1$ and g mapping the vertices and the edges in a vertical way.

The graph morphisms f and fg are acyclic cofibrations; cf. Remark 166.

But g is not an acyclic cofibration since it does not satisfy e.g. (AcCofib 2).

```

Y := <[1,2,3], [<1,1,2>, <1,2,3>]>;
f := VtoE(D(0), Y, [<0,1>]);
g := ListGraphMorphisms(Y, D(1)) [1];
> IsAcCofib(f, D(0), Y);
true
> IsAcCofib(ComposeGraphMorphisms(f, g), D(0), D(1));
true
> IsAcCofib(g, Y, D(1));
false

```

Since g is a quasiisomorphism, we infer that g is not a cofibration; cf. Lemma 185. \square

Assertion 249 Consider the following assertions (1–6) in a Quillen closed model category.

- (1) *Each quasiisomorphism that is a retraction is a fibration.*
- (2) *Each quasiisomorphism that is a coretraction is a cofibration.*
- (3) *Suppose given a commutative diagram as follows.*

$$\begin{array}{ccc}
 & Y & \\
 g \nearrow & & \searrow f \\
 X & \xrightarrow{gf} & Z
 \end{array}$$

Then $f : Y \rightarrow Z$ is a fibration.

- (4) *Suppose given a commutative diagram as follows.*

$$\begin{array}{ccc}
 & Y & \\
 g \nearrow & & \searrow f \\
 X & \xrightarrow{gf} & Z
 \end{array}$$

Then $f : Y \rightarrow Z$ is an acyclic fibration.

- (5) *Suppose given a commutative diagram as follows.*

$$\begin{array}{ccc}
 & Y & \\
 g \nearrow & & \searrow f \\
 X & \xrightarrow{gf} & Z
 \end{array}$$

Then $g : X \rightarrow Y$ is a cofibration.

(6) Suppose given a commutative diagram as follows.

$$\begin{array}{ccc}
 & Y & \\
 g \nearrow & & \searrow f \\
 X & \xrightarrow{gf} & Z
 \end{array}$$

Then $g : X \rightarrow Y$ is an acyclic cofibration.

The assertions (1–6) are **false**.

Counterexample in Gph.

We consider the following graph morphisms.

$$\begin{array}{ccc}
 Y : & \boxed{\begin{array}{c} 1 \\ \\ 2 \xrightarrow{\alpha_1} 3 \end{array}} & \\
 & \begin{array}{c} \uparrow \\ g \\ \downarrow \\ f \end{array} & \\
 X = Z : & \boxed{1 \xrightarrow{\beta_1} 2} &
 \end{array}$$

Here, the graph morphisms $f : Y \rightarrow Z$ and $g : X \rightarrow Y$ map the vertices and the edges in a vertical way, where $1V_g := 2$, $2V_g := 3$ and $\beta_1 E_g := \alpha_1$.

So we have $gf = \text{id}_X$.

So the graph morphism $f : Z \rightarrow Y$ is a retraction and the graph morphism $g : X \rightarrow Z$ is a coretraction.

Since $(C_k, X)_{\text{Gph}} = (C_k, Y)_{\text{Gph}} = \emptyset$ for $k \geq 1$, the graph morphisms f and g are quasiisomorphisms.

$$\begin{aligned}
 X &:= D(1); \\
 Z &:= D(1); \\
 Y &:= \langle [1, 2, 3], [\langle 2, 1, 3 \rangle] \rangle; \\
 g &:= \text{VtoE}(X, Y, [\langle 0, 2 \rangle, \langle 1, 3 \rangle]); \\
 f &:= \text{VtoE}(Y, Z, [\langle 1, 0 \rangle, \langle 2, 0 \rangle, \langle 3, 1 \rangle]);
 \end{aligned}$$

Ad (1).

The graph morphism $f : Y \rightarrow Z$ is a retraction and a quasiisomorphism but not a fibration.

```
> IsFibration(f,Y,Z);
false
```

Ad (2).

The graph morphism $g : X \rightarrow Z$ is a coretraction and a quasiisomorphism but not a cofibration.

```
> IsAcCofib(g,X,Z);
false
```

The graph morphism g is not an acyclic cofibration since g does not satisfy (AcCofib 3, 5).

Since g is a quasiisomorphism and since $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$, g is not a cofibration; cf. Lemma 185.

Alternative proof for (2). We have $\text{Cofib} = \square \text{AcFib}$; cf. Definition 144. The graph morphism g is an acyclic fibration since it is a quasiisomorphism and a fibration; cf. Definition 138.

```
> IsFibration(g,X,Y);
true
```

We consider the following commutative diagram.

$$\begin{array}{ccc} X & \xrightarrow{\text{id}_X} & X \\ g \downarrow & & \downarrow g \\ Y & \xrightarrow{\text{id}_Y} & Y \end{array}$$

But there does not exist a graph morphism $h : Z \rightarrow X$ such that $gh = \text{id}_X$ and $hg = \text{id}_X$ since g is not a graph isomorphism.

So g is not a cofibration.

Ad (3).

We have $gf = \text{id}_X : X \dashrightarrow Z$, but f is not a fibration.

Ad (4).

We have $gf = \text{id}_X : X \dashrightarrow Z$, but f is not an acyclic fibration since it is not a fibration.

Ad (5).

We have $gf = \text{id}_X : X \dashrightarrow Z$, but g is not a cofibration.

Ad (6).

We have $gf = \text{id}_X : X \dashrightarrow Z$, but g is not an acyclic cofibration. □

Remark 250 Using Gph^{op} , we can falsify the assertions dual to Assertions 245 – 249.

9.4.2 Counterexamples for pushouts and pullbacks

The following assertions take place in a Quillen closed model category.

Assertion 251 *In a Quillen closed model category, a pushout of a quasiisomorphism along a cofibration is a quasiisomorphism.*

This assertion is **false**.

Counterexample in Gph.

A counterexample is given in Remark 205. □

Assertion 252

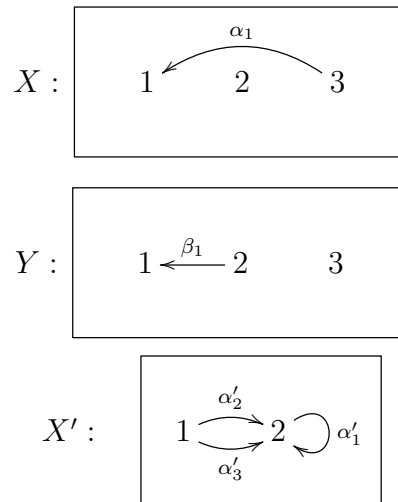
Consider the following assertions (1–4) in a Quillen closed model category.

- (1) *A pushout of a fibration is a fibration.*
- (2) *A pushout of an acyclic fibration is a fibration.*
- (3) *A pushout of an acyclic fibration is a quasiisomorphism.*
- (4) *A pushout of an etale fibration is a fibration.*

The assertions (1), (2), (3) and (4) are **false**.

Counterexample in Gph.

We consider the following graphs.



Let $f = (V_f, E_f) : X \rightarrow Y$ be the graph morphism with

$$1 V_f = 1, \quad 2 V_f = 1, \quad 3 V_f = 2$$

and with

$$\alpha_1 E_f = \beta_1 .$$

Then the graph morphism f is an etale fibration as we calculated with Magma [2].

```

X := <[1,2,3],[<3,1,1>>>;
Y := <[1,2,3],[<2,1,1>>>;
f := <[<1,1>,<2,1>,<3,2>],[<<3,1,1>,<2,1,1>>>>;
> IsFibration(f,X,Y);
true
> IsEtaleFibration(f,X,Y);
true

```

Moreover, f is a quasiisomorphism since $(C_n, X)_{\text{Gph}} = \emptyset = (C_n, Y)_{\text{Gph}}$ for $n \in \mathbb{N}$.

In conclusion, f is an acyclic fibration.

Let $g = (V_g, E_g) : X \rightarrow X'$ be the graph morphism with

$$1 V_g = 2, \quad 2 V_g = 1, \quad 3 V_g = 2$$

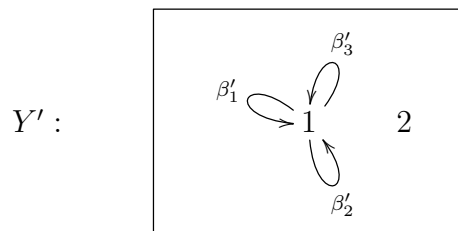
and with

$$\alpha_1 E_g = \alpha'_1.$$

We calculated a pushout with Magma.

$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

We obtained the pushout Y' .



We have

$$\begin{aligned}
 1 V_{f'} &= 1, & 2 V_{f'} &= 1, \\
 \alpha'_1 E_{f'} &= \beta'_1, & \alpha'_2 E_{f'} &= \beta'_2, & \alpha'_3 E_{f'} &= \beta'_3
 \end{aligned}$$

and

$$\begin{aligned}
 1 V_h &= 1, & 2 V_h &= 1, & 3 V_h &= 2, \\
 \beta_1 E_h &= \beta'_1.
 \end{aligned}$$

The graph morphism f is an etale fibration but the graph morphism f' is not.

```

Xp := <[1,2],[<2,1,2>,<1,2,2>,<1,3,2>>>; // "X prime" = X'
g := ListGraphMorphisms(X,Xp)[1];
> g;
<[ <1, 2>, <2, 1>, <3, 2> ], [ <<3, 1, 1>, <2, 1, 2>> ]>
> IsGraphMorphism(g,X,Xp);
true

```

```

PO := PushoutGraphs(X,Y,Xp,f,g);
> PO;
<<[ 1, 2 ], [ <1, 1, 1>, <1, 2, 1>, <1, 3, 1> ]>,
<[ <1, 1>, <2, 1> ], [ <<2, 1, 2>, <1, 1, 1>>, <<1, 2, 2>, <1, 2, 1>>,
      <<1, 3, 2>, <1, 3, 1>> ]>,
<[ <1, 1>, <2, 1>, <3, 2> ], [ <<2, 1, 1>, <1, 1, 1>> ]>>
Yp := PO[1]; // "Y prime" = Y'
fp := PO[2]; // "f prime" = f'
h := PO[3];
> IsFibration(fp,Xp,Yp);
false
> IsQis_Bound(fp,Xp,Yp,1);
false

```

Here, the pushout $f' : X' \rightarrow Y'$ of the acyclic fibration $f : X \rightarrow Y$ is neither a fibration nor a quasiisomorphism, as we calculated with Magma. We can also directly see that

$$|\{e \in E_{X'} : es_{X'} = 2\}| = 1 \neq 3 = |\{e \in E_{Y'} : es_{Y'} = 2 \vee_{f'} = 1\}|$$

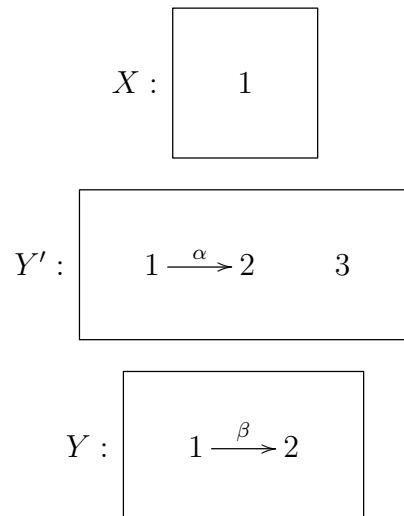
and that $|(C_1, X')_{\text{Gph}}| = 1 \neq 3 = |(C_1, Y')_{\text{Gph}}|$. □

Assertion 253 *In a Quillen closed model category, a pullback of an acyclic cofibration is a cofibration.*

This assertion is **false**.

Counterexample in Gph.

We consider the following graphs.



Let $f = (V_f, E_f) : X \rightarrow Y$ be the graph morphism with

$$1 \vee_f = 1.$$

Then the graph morphism f is an acyclic cofibration, as we calculated with Magma [2].

Let $q = (V_q, E_q) : Y' \rightarrow Y$ be the graph morphism with

$$\begin{aligned} 1 V_q &= 1, \\ 2 V_q &= 2, \\ 3 V_q &= 2, \end{aligned}$$

and with

$$\alpha E_q = \beta.$$

Let $f' : X' \rightarrow Y'$ be the pullback of $f : X \rightarrow Y$ along q .

$$\begin{array}{ccc} X' & \xrightarrow{p} & X \\ f' \downarrow & \lrcorner & \downarrow f \\ Y' & \xrightarrow{q} & Y \end{array}.$$

Via Magma, we obtain $V_{X'} = \{1\}$ and $E_{X'} = \emptyset$ and

$$1 V_{f'} = 1,$$

and

$$1 V_p = 1.$$

$$X' : \begin{array}{|c|} \hline 1 \\ \hline \end{array}$$

The graph morphism f is an acyclic cofibration, but the graph morphism f' is not, as we calculated with Magma:

```
X := <[1], []>;
Y := <[1,2], [<1,1,2>]>;
Yp := <[1,2,3], [<1,1,2>]>; // "Y prime" = Y'
f := <[<1,1>], []>;
q := <[<1,1>, <2,2>, <3,2>], [<<1,1,2>, <1,1,2>>]>;
> IsGraphMorphism(f,X,Y);
true
> IsAcCofib(f,X,Y);
true
> IsGraphMorphism(q,Yp,Y);
true
PB := PullbackGraphs(X,Y,Yp,f,q);
> PB;
<<[ 1 ], []>, [<1, 1> ], []>, [<1, 1> ], []>>;
Xp := PB[1]; // "X prime" = X'
fp := PB[2]; // "f prime" = f'
q := PB[3];
```

```

> IsAcCofib(fp,Xp,Yp);
false
> AcCofib1to4(fp,Xp,Yp);
false
> AcCofib5(fp,Xp,Yp);
false

```

We can see that the vertex $3 \in V_{Y'} \setminus V_{X'f'}$ does not have an edge $e \in E_{Y'}$ such that $e t_{Y'} = 3$. So the graph morphism $f' : X' \rightarrow Y'$ does not satisfy (AcCofib 3). Hence it is not an acyclic cofibration.

The pullback $f' : X' \rightarrow Y'$ of the acyclic cofibration $f : X \rightarrow Y$ is **not** a cofibration since first, it is not an acyclic cofibration, and second, it is a quasiisomorphism as $(C_n, Y')_{\text{Gph}} = \emptyset = (C_n, X')_{\text{Gph}}$, for $n \in \mathbb{N}$, or by Remark 125; cf. Lemma 185.

Alternatively, to show that f' is not a cofibration, we can consider the commutative diagram

$$\begin{array}{ccc}
 X' & \xrightarrow{p} & C_1 \\
 f' \downarrow & & \downarrow \iota_1 \\
 Y' & \xrightarrow{r} & C_1 \sqcup D_0,
 \end{array}$$

where $1V_p := v_0$, $1V_r = (1, v_0)$, $2V_r = (1, v_0)$ and $3V_r := (2, \hat{v}_0)$.

The graph morphism ι_1 is in fact a quasiisomorphism since $(C_k, \iota_1)_{\text{Gph}}$ is injective and $|(C_k, C_1)_{\text{Gph}}| = 1 = |(C_k, C_1 \sqcup D_0)_{\text{Gph}}|$.

Moreover, ι_1 is a fibration, as we will verify with Magma. So ι_1 is an acyclic fibration.

The unique graph morphism $h : Y' \rightarrow C_1$ maps the vertex 3 to v_0 .

So we have $3V_r = (2, \hat{v}_0) \neq (1, v_0) = v_0 V_{\iota_1} = 3V_{h\iota_1}$.

Thus $h\iota_1 \neq r$.

So f' does not have the left lifting property with respect to the acyclic fibration ι_1 .

So $f' : X' \rightarrow Y'$ is not a cofibration; cf. Definition 144. □

```

Xp := <[1], []>; // "X prime" = X'
p := VtoE(Xp,C(1), [<1,1>]);
> p;
<[ <1, 1> ], []>
Yp := <[1,2,3], [<1,1,2>]>; // "Y prime" = Y'
fp := <[<1,1>], []>; // "f prime" = f'
C1D0 := <[0,1], [<0,1,0>]>;
r := <[<1,0>, <2,0>, <3,1>], [<<1,1,2>, <0,1,0>>]>;
iota_1 := VtoE(C(1),C1D0, [<1,0>]);
> iota_1;
<[ <1, 0> ], [ <<1, 1, 1>, <0, 1, 0>> ]>
> IsFibration(iota_1,C(1),C1D0);
true

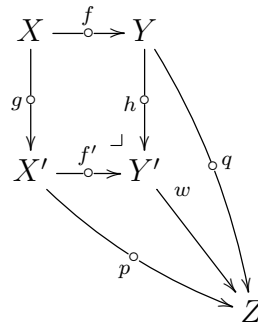
```

```
> Lift(Xp,C(1),Yp,C1D0,p,r,fp,iota_1);
false
```

Note that in Gph , a pullback of an acyclic cofibration is a quasiisomorphism; cf. Remark 176.

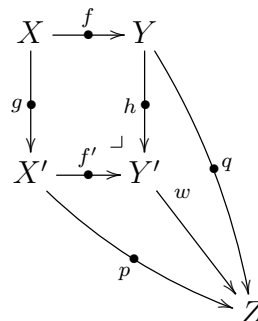
Assertion 254 Consider the following assertions (1–2) in a Quillen closed model category.

(1) Suppose given a commutative diagram as follows.



Then the graph morphism $w : Y' \rightarrow Z$ is an acyclic cofibration.

(2) Suppose given a commutative diagram as follows.

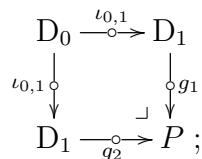


Then the graph morphism $w : Y' \rightarrow Z$ is a cofibration.

The assertions (1) and (2) are **false**.

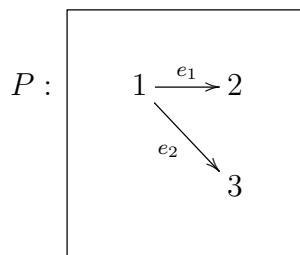
Ad (1). Counterexample in Gph .

We consider

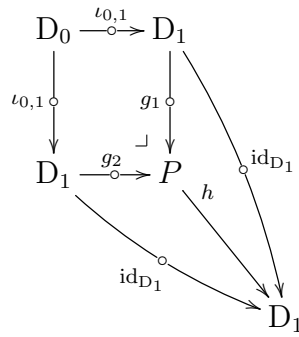


where $\hat{e}_0 E_{g_1} := e_1$ and $\hat{e}_0 E_{g_2} := e_2$; cf. Remarks 166 and 171.

Magma gives



We now consider



We have

$$\begin{aligned}
 h &: P \rightarrow D_1 \\
 V_h &: 1 \mapsto \hat{v}_0 \\
 &: 2 \mapsto \hat{v}_1 \\
 &: 3 \mapsto \hat{v}_1 \\
 E_h &: e_1 \mapsto \hat{e}_0 \\
 &: e_2 \mapsto \hat{e}_0
 \end{aligned}$$

Note that the graph morphism $\text{id}_{D_1} : D_1 \rightarrow D_1$ is an acyclic cofibration; cf. Remark 165.

But the graph morphism $h : P \rightarrow D_1$ is **not** an acyclic cofibration, since neither V_h nor E_h are injective; cf. Definition 162.

```

D0 := D(0);
D1 := D(1);
f := VtoE(D0,D1,[<0,0>]); // iota_{0,1}
> D0;
<[ 0 ], []>
> D1;
<[ 0, 1 ], [ <0, 0, 1> ]>
> f;
<[ <0, 0> ], []>
P0 := PushoutGraphs(D0,D1,D1,f,f);
> P0[1];
<[ 1, 2, 3 ], [ <1, 1, 2>, <1, 2, 3> ]>
> P0[2];
<[ <0, 1>, <1, 2> ], [ <<0, 0, 1>, <1, 1, 2>> ]>
> P0[3];
<[ <0, 1>, <1, 3> ], [ <<0, 0, 1>, <1, 2, 3>> ]>
> IsAcCofib(Identity(D1),D1,D1);
true
h := VtoE(P0[1],D1,[<1,0>,<2,1>,<3,1>]);
> h;
<[ <1, 0>, <2, 1>, <3, 1> ], [ <<1, 1, 2>, <0, 0, 1>>, <<1, 2, 3>, <0, 0, 1>> ]>
> IsEqual(ComposeGraphMorphisms(P0[2],h),Identity(D1));
true

```

```

> IsEqual(ComposeGraphMorphisms(P0[3],h),Identity(D1));
true
> IsAcCofib(h,P0[1],D1);
false
> IsFibration(h,P0[1],D1);
true

```

Note that h is a quasiisomorphism by (2 of 3).

Thus h is not a cofibration.

By the way, h is an acyclic fibration as it is a quasiisomorphism and a fibration.

Ad (2). Counterexample in Gph.

We consider

$$\begin{array}{ccc}
 D_0 & \xrightarrow{\iota_{0,1}} & D_1 \\
 \iota_{0,1} \downarrow & & \downarrow g_1 \\
 D_1 & \xrightarrow{g_2} & P
 \end{array}$$

where $\hat{e}_0 E_{g_1} := e_1$ and $\hat{e}_0 E_{g_2} := e_2$ as above; cf. Lemma 185 and Remark 148.

We now consider

$$\begin{array}{ccc}
 D_0 & \xrightarrow{\iota_{0,1}} & D_1 \\
 \iota_{0,1} \downarrow & & \downarrow g_1 \\
 D_1 & \xrightarrow{g_2} & P \\
 & & \downarrow h \\
 & & D_1
 \end{array}$$

$\text{id}_{D_1} : D_1 \rightarrow D_1$ (curved arrow from D_1 to D_1)

as above.

Note that the graph morphism $\text{id}_{D_1} : D_1 \rightarrow D_1$ is a cofibration; cf. Remark 145.

But we will show that the graph morphism $h : P \rightarrow D_1$ is not a cofibration.

Since $(C_n, P) = \emptyset = (C_n, D_1)$ the graph morphism $h : P \rightarrow D_1$ is a quasiisomorphism; cf. Definition 115.

But the graph morphism h is not an acyclic cofibration since it does not satisfy (AcCofib 1) since $2V_h = \hat{v}_1 = 3V_h$.

Since we have $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$ the quasiisomorphism h is not a cofibration; cf. Lemma 175. \square

Assertion 255 *Suppose given a pushout as follows.*

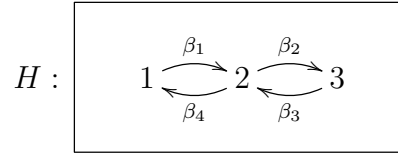
$$\begin{array}{ccc}
 X & \xrightarrow{f} & Y \\
 g \downarrow & & \downarrow h \\
 X' & \xrightarrow{f'} & Y'
 \end{array}$$

If Y and X' are cofibrant, then Y' is cofibrant.

This assertion is **false**.

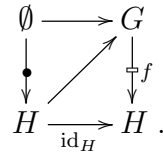
Counterexample in Gph.

We consider the following graph.



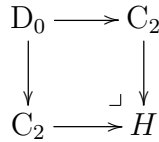
The graph H is not cofibrant; cf. Definition 150.

Assume that H is cofibrant. Consider $f : G \twoheadrightarrow H$ from Example 215. Then we obtain a commutative diagram as follows



But $f : G \rightarrow H$ is not a retraction; cf. Counterexample 2 to Assertion 245.

However, Magma yields a pushout as follows.



□

Note that C_2 is cofibrant; cf. Remark 151.

```

D0 := D(0);
C := C(2);
f := VtoE(D0,C,[<0,1>]);
> D0;
<[ 0 ], []>
> C;
<[ 1, 2 ], [ <1, 1, 2>, <2, 2, 1> ]>
> f;
<[ <0, 1> ], []>
PO := PushoutGraphs(D0,C,C,f,f);
> PO[1];
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 1>, <1, 3, 3>, <3, 4, 1> ]>
> PO[2];
<[ <1, 1>, <2, 2> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>> ]>
> PO[3];
<[ <1, 1>, <2, 3> ], [ <<1, 1, 2>, <1, 3, 3>>, <<2, 2, 1>, <3, 4, 1>> ]>

```

We obtain that $PO[1] \simeq H$.

Remark 256 Using Gph^{op} , we can falsify the assertions dual to Assertions 251 – 255.

9.5 Counterexamples in Gph

Assertion 257

(1) *Suppose given an acyclic fibration $f : G \twoheadrightarrow H$.*

Suppose given $n \in \mathbb{Z}_{\geq 1}$. Suppose given graph morphisms $C_n \xrightarrow{p} G$ and $C_n \xrightarrow{q} G$.

If $C_n p \cap C_n q = \emptyset$, then $C_n pf \cap C_n qf = \emptyset$.

(2) *Suppose given an acyclic fibration $f : G \twoheadrightarrow H$.*

Suppose given $n \in \mathbb{Z}_{\geq 1}$. Suppose given graph morphisms $C_n \xrightarrow{p} G$, $C_n \xrightarrow{q} G$ and $C_n \xrightarrow{r} G$.

If $C_n p$, $C_n q$ and $C_n r$ are pairwise disjoint, so are $C_n pf$, $C_n qf$ and $C_n rf$.

The assertions (1, 2) are **false**.

Counterexamples.

Ad (1). We consider the graph morphism $g : K \rightarrow H$ from Example 216.

We let $n := 2$

We consider the graph morphisms $p, q : C_2 \rightarrow K$, where $v_0 V_p := 1$, $v_1 V_p := 2$, $v_0 V_q := 2'$ and $v_1 V_q := 3$.

Then we have $C_2 p \cap C_2 q = \emptyset$, but $C_2 pf \cap C_2 qf$ consists the vertex 2.

Ad (2). We consider the graph morphism $g : K \rightarrow H$ from Example 218.

We let $n := 2$

We consider the graph morphisms $p, q, r : C_2 \rightarrow K$ where $v_0 V_p := 1$, $v_1 V_p := 2$, $v_0 V_q := 2'$, $v_1 V_q := 3$, $v_0 V_r := 3'$ and $v_1 V_r := 4$.

Then we have $C_2 p \cap C_2 q = \emptyset$, but $C_2 pf \cap C_2 qf$ consists the vertex 2.

And we have $C_2 q \cap C_2 r = \emptyset$, but $C_2 qf \cap C_2 rf$ consists the vertex 3. □

Assertion 258

(1) *Suppose given a graph morphism $f : G \rightarrow H$.*

If for each $n \geq 1$ and each graph morphism $u : C_n \rightarrow H$ such that E_u is injective there exists a unique graph morphism $\hat{u} : C_n \rightarrow G$ with $\hat{u}f = u$, then f is a quasiisomorphism.

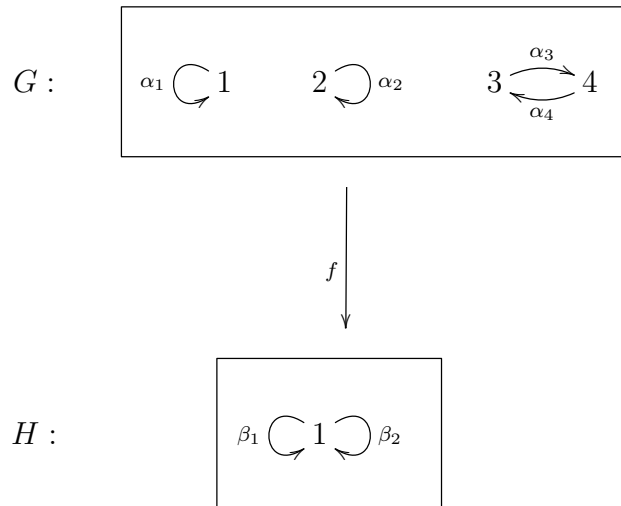
(2) *Suppose given $m \in \mathbb{N}$ such that there does not exist an injective graph morphism $g : C_n \rightarrow G$ for $n > m$ and such that there does not exist an injective graph morphism $h : C_n \rightarrow H$ for $n > m$ and such that the map $(C_n, f)_{\text{Gph}}$ is bijective for $n \in [1, m]$.*

Then the graph morphism $f : G \rightarrow H$ is a quasiisomorphism.

The assertions (1, 2) are **false**.

Counterexample.

We consider the following graph morphism.



Here, $f : G \rightarrow H$ is the graph morphism with

$$\begin{array}{cccc}
 1 V_f = 1 & 2 V_f = 1 & 3 V_f = 1 & 4 V_f = 1 \\
 \alpha_1 E_f = \beta_1 & \alpha_2 E_f = \beta_2 & \alpha_3 E_f = \beta_1 & \alpha_4 E_f = \beta_2
 \end{array}$$

Ad (1).

There are two graph morphisms $p, q : C_1 \rightarrow H$ with E_p, E_q injective, having $e_0 E_p = \beta_1$ and $e_0 E_q = \beta_2$.

For each of these injective graph morphisms there exists a unique graph morphism $\tilde{p}, \tilde{q} : C_1 \rightarrow G$ such that $\tilde{p}f = p$ and $\tilde{q}f = q$.

There are two graph morphisms $p, q : C_2 \rightarrow H$ with E_p and E_q injective. In particular, we have $e_0 E_p = \beta_1$ and $e_0 E_q = \beta_2$.

For each of these injective graph morphisms exists a unique graph morphism $\tilde{p}, \tilde{q} : C_2 \rightarrow G$ such that $\tilde{p}f = p$ and $\tilde{q}f = q$.

But the graph morphism $f : G \rightarrow H$ is not a quasiisomorphism.

Let $g : C_3 \rightarrow G$ be the graph morphism with

$$\begin{array}{ccc}
 v_0 V_g = 1, & v_1 V_g = 1, & v_2 V_g = 1, \\
 e_0 E_g = \beta_1, & e_1 E_g = \beta_2, & e_2 E_g = \beta_1.
 \end{array}$$

There does not exist a graph morphism $\tilde{g} : C_3 \rightarrow G$ such that $\tilde{g}f = g$.

So the graph morphism $f : G \rightarrow H$ is not a quasiisomorphism.

Ad (2).

We have $m = 2$ since there does not exist an injective graph morphism $g : C_n \rightarrow G$ for $n > 2$ and there does not exist an injective graph morphism $h : C_n \rightarrow H$ for $n > 2$. Furthermore, $(C_n, f)_{\text{Gph}}$ is bijective for $n \in [1, 2]$, but $(C_3, f)_{\text{Gph}}$ is not bijective since it is not surjective. \square

Assertion 259 *Suppose given*

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

in Gph.

- (1) *If $(C_n, f)_{\text{Gph}}$ is surjective for $n \geq 1$, then $(C_n, f')_{\text{Gph}}$ is surjective for $n \geq 1$.*
- (2) *If f is a retraction, then f' is a retraction.*

The assertions (1) and (2) are **false**.

Counterexample.

Let $X := C_2 \sqcup C_2$. Let $Y := C_2$. Let $f := d_{C_2} : C_2 \sqcup C_2 \rightarrow C_2$.

Then f is a retraction. In particular, $(C_n, f)_{\text{Gph}}$ is surjective for $n \geq 1$.

Let

$$X' : \boxed{\begin{array}{ccccccc} & & \alpha_1 & & & & \alpha_5 & & \\ & & \curvearrowright & & \alpha_3 & \rightarrow & \alpha_4 & & \curvearrowleft & & \\ 1 & & & 2 & & 3 & & 4 & & 5 & \\ & & \alpha_2 & & & & & & \alpha_6 & & \end{array}}$$

Let $g : C_2 \sqcup C_2 \rightarrow X$ be defined by

$$\begin{aligned} (1, v_0) V_g &:= 1, & (1, v_1) V_g &:= 2, \\ (2, v_0) V_g &:= 5, & (2, v_1) V_g &:= 4. \end{aligned}$$

This defines g since $C_2 \sqcup C_2$ is thin; cf. Remark 77.

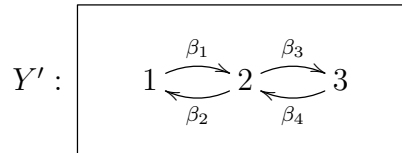
We form the pushout

$$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ g \downarrow & & \downarrow h \\ X' & \xrightarrow{f'} & Y' \end{array}$$

via Magma.

```
X := DUC([2,2]);
Y := C(2);
Xp := <[1,2,3,4,5], [<1,1,2>, <2,2,1>, <2,3,3>, <3,4,4>, <4,5,5>, <5,6,4>]>;
      // "X prime" = X'
f := VtoE(X,Y, [<1,1>, <2,2>, <3,1>, <4,2>]);
g := VtoE(X,Xp, [<1,1>, <2,2>, <3,5>, <4,4>]);
PO := PushoutGraphs(X,Y,Xp,f,g);
Yp := PO[1]; // "Y prime" = Y'
fp := PO[2]; // "f prime" = f'
h := PO[3];
```

We have obtained



We verify that $(C_n, f')_{\text{Gph}}$ is not surjective.

```
listXp := ListGraphMorphisms(C(2),Xp);
listYp := Sort(ListGraphMorphisms(C(2),Yp));
listXpfp := RedSeq([ComposeGraphMorphisms(k,fp) : k in listXp]);
// set of images
> #listYp;
4
> #listXpfp;
2
```

In particular, f' is not a retraction.

We verify this again independently as follows.

We search for a coretraction to f' ; i.e. a graph morphism $c : Y' \rightarrow X'$ such that $cf \cdot f' = \text{id}_{Y'}$.

```
> Identity(Yp);
<[ <1, 1>, <2, 2>, <3, 3> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2, 1>>,
<<2, 3, 3>, <2, 3, 3>>, <<3, 4, 2>, <3, 4, 2>> ]>
list := ListGraphMorphisms(Yp,Xp);
> #list;
4
Cf := [ComposeGraphMorphisms(l,fp) : l in list];
> Cf;
[
  <[ <1, 1>, <2, 2>, <3, 1> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2,
  1>>, <<2, 3, 3>, <2, 2, 1>>, <<3, 4, 2>, <1, 1, 2>> ]>,
  <[ <1, 2>, <2, 1>, <3, 2> ], [ <<1, 1, 2>, <2, 2, 1>>, <<2, 2, 1>, <1, 1,
  2>>, <<2, 3, 3>, <1, 1, 2>>, <<3, 4, 2>, <2, 2, 1>> ]>,
  <[ <1, 2>, <2, 1>, <3, 2> ], [ <<1, 1, 2>, <2, 2, 1>>, <<2, 2, 1>, <1, 1,
  2>>, <<2, 3, 3>, <1, 1, 2>>, <<3, 4, 2>, <2, 2, 1>> ]>,
  <[ <1, 1>, <2, 2>, <3, 1> ], [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 1>, <2, 2,
  1>>, <<2, 3, 3>, <2, 2, 1>>, <<3, 4, 2>, <1, 1, 2>> ]>
]
Cfid := [l : l in list | IsEqual(ComposeGraphMorphisms(l,fp),Identity(Yp))];
> #Cfid;
0
```

So $f' : X' \rightarrow Y'$ is not a retraction since there does not exist a coretraction. \square

Assertion 260 *Suppose given an acyclic fibration $f : G \dashrightarrow H$.*

Suppose given $n \in \mathbb{N}$ and an injective graph morphism $\iota : C_n \rightarrow G$.

Then the graph morphism $\iota \cdot f$ is injective.

This assertion is **false**.

Counterexample.

We consider the graphs G , H and the acyclic fibration f as in Example 215. Consider the injective graph morphism $\iota : C_4 \rightarrow G$ given by

$$\begin{aligned} v_1 V_\iota &= 1, & v_2 V_\iota &= 2, \\ v_3 V_\iota &= 3, & v_4 V_\iota &= 2', \end{aligned}$$

and by

$$\begin{aligned} e_1 E_\iota &= \alpha_1, & e_2 E_\iota &= \alpha_3, \\ e_3 E_\iota &= \alpha_4, & e_4 E_\iota &= \alpha_7. \end{aligned}$$

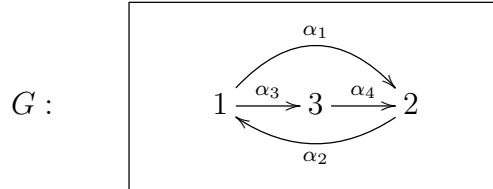
Then $(v_2) V_{\iota f} = 2 = (v_4) V_{\iota f}$. So the graph morphism $\iota \cdot f$ is not injective. \square

Assertion 261 *Suppose given $n \in \mathbb{N}$ and a graph morphism $f : C_n \rightarrow G$. Then there exists $k \in \mathbb{N}$ and a graph morphism $g : C_k \rightarrow G$ such that E_g is injective and $C_n f = C_k g$.*

This assertion is **false**.

Counterexample.

We consider the following graph.



We consider the graph morphism $f : C_5 \rightarrow G$ with

$$\begin{aligned} v_0 V_f &:= 1, & v_1 V_f &:= 2, & v_2 V_f &:= 1, & v_3 V_f &:= 3, & v_4 V_f &:= 2, \\ e_0 E_f &:= \alpha_1, & e_1 E_f &:= \alpha_2, & e_2 E_f &:= \alpha_3, & e_3 E_f &:= \alpha_4, & e_4 E_f &:= \alpha_2. \end{aligned}$$

Assume that there exists $k \in \mathbb{N}$ and an injective graph morphism $g : C_k \rightarrow G$ such that $C_n f = C_k g$.

Since $C_n f = G$, we have $k = |E_G| = 4$.

Without loss of generality, we have $e_0 E_g = \alpha_1$.

Hence $e_1 E_g = \alpha_2$.

Hence $e_2 E_g = \alpha_3$, using injectivity of E_g .

Hence $e_3 E_g = \alpha_4$.

Hence $e_0 E_g = e_4 E_g = \alpha_2$.

Contradiction. \square

Chapter 10

Algorithmic treatment of graphs

We use Magma [2] to codify finite graphs and graph morphisms in order to perform calculations.

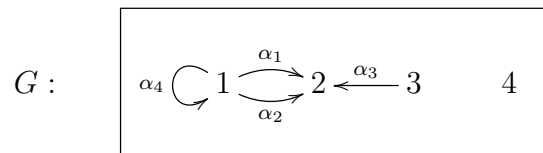
10.1 Implementation of graphs

In the following functions we calculate graphs and graph morphisms as introduced in Definition 45 and 54.

In our implementation, a graph is a tuple consisting of its list of vertices as first entry and its list of edges as second entry.

An edge is a triple consisting of its source vertex as first entry, its name as second entry and its target vertex as third entry.

For instance, in Example 49 we consider the graph



It has the following codification.

```
G := <[1,2,3,4], [<1,1,2>, <1,2,2>, <3,3,2>, <1,4,1>]>;  
> G;  
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <1, 2, 2>, <3, 3, 2>, <1, 4, 1> ]>
```

Note that the edges are codified by their indices, so e.g. $1 \xrightarrow{\alpha_2} 2$ is codified as $\langle 1, 2, 3 \rangle$, and $1 \xrightarrow{\alpha_4} 1$ is codified as $\langle 1, 4, 1 \rangle$.

Here,

```
> G[1];  
[ 1, 2, 3, 4 ]
```

is the list of the four vertices in the graph G and

```
> G[2];
[ <1, 1, 2>, <1, 2, 2>, <3, 3, 2>, <1, 4, 1> ]
```

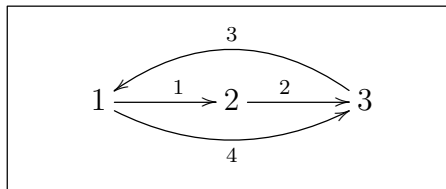
is the list of the four edges in the graph G .

The following function `SetGraphExample` gives us a random graph with exactly e edges and v vertices.

```
SetGraphExample := function(v,e) // v: number of vertices, e: number of edges
  edges := [];
  for j in [1..e] do
    edges cat:= [<Random([1..v]),j,Random([1..v])>];
  end for;
  return <[i : i in [1..v]],edges>;
end function;
```

For example the following graph has been obtained by this function.

```
> SetGraphExample(3,4);
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 1>, <1, 4, 3> ]>
```

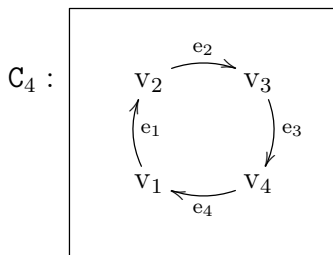


Given n in \mathbb{N} , with the function `CyclicGraph` we calculate the cyclic graph C_n .

```
CyclicGraph := function(n) // returns cyclic graph with n edges
  return <[i : i in [1..n]], [<i,i,i+1> : i in [1..n-1]] cat [<n,n,1>]>;
end function;
```

For example, we get

```
> CyclicGraph(1);
<[ 1 ], [ <1, 1, 1> ]>
> CyclicGraph(2);
<[ 1, 2 ], [ <1, 1, 2>, <2, 2, 1> ]>
> CyclicGraph(3);
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 1> ]>
> CyclicGraph(4);
<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 4>, <4, 4, 1> ]>
```

Note that we have $v_4 = v_0$ and $e_4 = e_0$.

For further use, we abbreviate

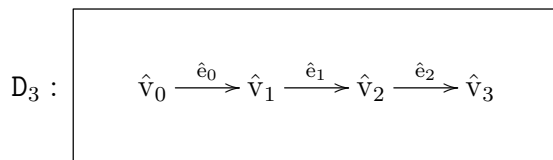
```
C := CyclicGraph; // %%
```

Given n in \mathbb{N} , with the function `DirectGraph` we calculate the direct graph D_n .

```
DirectGraph := function(n)
  return <[i : i in [0..n]], [<i, i, i+1> : i in [0..n-1]]>;
end function;
```

For example, we get

```
> DirectGraph(0);
<[ 0 ], []>
> DirectGraph(1);
<[ 0, 1 ], [ <0, 0, 1> ]>
> DirectGraph(2);
<[ 0, 1, 2 ], [ <0, 0, 1>, <1, 1, 2> ]>
> DirectGraph(3);
<[ 0, 1, 2, 3 ], [ <0, 0, 1>, <1, 1, 2>, <2, 2, 3> ]>
```



For further use, we abbreviate

```
D := DirectGraph; // %%
```

With the function `IsThin` we can test if a given graph G is thin.

```
IsThin := function(G)
  E := {<e[1], e[3]> : e in G[2]};
  return #E eq #G[2];
end function;
```

For example, given the graph G from above we get

```
> IsThin(G);
false
> IsThin(C(4));
true
```

10.2 Implementation of graph morphisms

In our implementation, a graph morphism $f : G \rightarrow H$ is a tuple consisting of the map V_f as first entry and the map E_f as second entry.

Such a map on vertices is a list of tuples with each vertex $v_G \in V_G$ of the graph G as first entry and its image $v_G V_f \in V_H$ as second entry.

Such a map on edges is a list of tuples with each edge $e_G \in E_G$ of the graph G as first entry and its image $e_G E_f \in E_H$ as second entry.

Given graphs G and H as shown below, for example f is a graph morphism from G to H .

```
G := <[ 1, 2, 3, 4 ], [ <1, 1, 1>, <1, 2, 2>, <3, 3, 2> ]>;
H := <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 2>, <3, 3, 2> ]>;
f := <[ <1, 2>, <2, 2>, <3, 1>, <4, 1> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>;
```

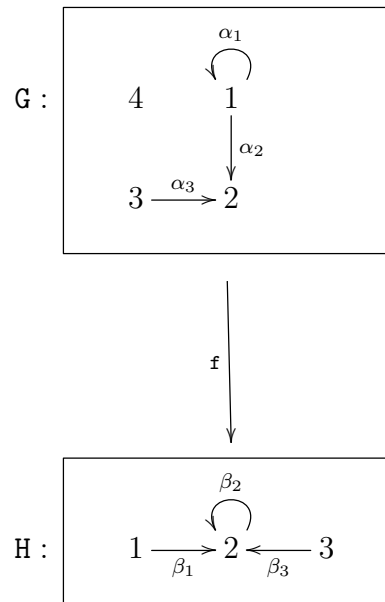
We have the map on the vertices

```
> f[1];
[ <1, 2>, <2, 2>, <3, 1>, <4, 1> ]
```

and the map on the edges

```
> f[2];
[ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]
```

We have



and the graph morphism $f : G \rightarrow H$ with

$$\begin{aligned}
 1 V_f &= 2 & 2 V_f &= 2 & 3 V_f &= 1 & 4 V_f &= 1 \\
 \alpha_1 E_f &= \beta_2 & \alpha_2 E_f &= \beta_2 & \alpha_3 E_f &= \beta_1 .
 \end{aligned}$$

With the function `IsGraphMorphism` we can test if a given pair of maps `m` from a given graph `G` to a given graph `H` is a graph morphism.

```

IsGraphMorphism := function(m,G,H) // m = <v,e>
  v := m[1];
  e := m[2];
  // test, if v is a map:
  if not SequenceToMultiset([x[1] : x in v]) eq SequenceToMultiset(G[1]) then
    return false;
  end if;
  if not SequenceToSet([x[2] : x in v]) subset SequenceToSet(H[1]) then
    return false;
  end if;
  // test, if e is a map:
  if not SequenceToMultiset([x[1] : x in e]) eq SequenceToMultiset(G[2]) then
    return false;
  end if;
  if not SequenceToSet([x[2] : x in e]) subset SequenceToSet(H[2]) then
    return false;
  end if;
  // test, if v and e are compatible concerning source and target:
  for x in e do
    if not &and[<x[1][1],x[2][1]> in v, <x[1][3],x[2][3]> in v] then
      return false;
    end if;
  end for;
  return true;
end function;

```

We can confirm that the graph morphism `f` above actually is a graph morphism.

```

> IsGraphMorphism(f,G,H);
true

```

Moreover, we give an example of a pair of maps `g`

```

g := <[<1,2>,<2,2>,<3,2>,<4,1>],
      [<<1,1,1>,<2,2,2>>,<<1,2,2>,<2,2,2>>,<<3,3,2>,<1,1,2>>>>;

```

for which

```

> IsGraphMorphism(g,G,H);
false

```

Using the following functions we calculate the list of all graph morphisms from a given graph `G` to a given graph `H`.

```

IsRightUnique := function(u) // u: relation, e.g. u := [<1,3>,<1,4>,<2,3>],
                        // or u := {<1,3>,<1,4>,<2,3>}

right_unique := true;
left_elements := {x[1] : x in u};
for y in left_elements do
  if #{x[2] : x in u | x[1] eq y} ge 2 then
    right_unique := false;
    break y;
  end if;
end for;
return right_unique;
end function;

```

```

CompletionsToMaps := function(D,C,u)
// D/C: list of elements in the domain/codomain,
// u: right unique relation
// e.g. D := [1,2,3,4]; C := [1,2,3,4,5]; u := {<1,3>, <3,5>};
to_be_mapped := [x : x in D | not x in {y[1] : y in u}];
list := [u];
for i in to_be_mapped do
  list_new := [];
  for v in list do
    for j in C do
      list_new cat:= [v join {<i,j>}];
    end for;
  end for;
  list := list_new;
end for;
return list;
end function;

```

```

RelationOnVerticesFromPartialMapOnEdges := function(x);
// x: partial map on edges from graph G to graph H (G, H not required as data)
return {<z[1][1],z[2][1]> : z in x} join {<z[1][3],z[2][3]> : z in x};
end function;

```

For further use, we abbreviate

```
RVPME := RelationOnVerticesFromPartialMapOnEdges; // %%
```

```

ListGraphMorphisms := function(G,H)
list := [[]];
for z in G[2] do
  list_new := [];
  for w in H[2] do
    for x in list do
      x_test := x cat [<z,w>];

```

```

    if IsRightUnique(RVPME(x_test)) then // %%
      list_new cat:= [x_test];
    end if;
  end for;
end for;
list := list_new;
end for;
list_mor := [];
for y in list do
  list_completions_to_maps_on_vertices := [Sort(SetToSequence(x)) : x in
    CompletionsToMaps(G[1],H[1],RVPME(y))]; // %%
  list_mor cat:= [<x,y> : x in list_completions_to_maps_on_vertices];
end for;
return list_mor;
end function;

```

For example, we can list all graph morphisms between the graphs G and H given above.

```

> ListGraphMorphisms(G,H);
[
  <[ <1, 2>, <2, 2>, <3, 1>, <4, 1> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 1>, <4, 2> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 1>, <4, 3> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 2>, <4, 1> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <2, 2, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 2>, <4, 2> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <2, 2, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 2>, <4, 3> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <2, 2, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 3>, <4, 1> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 3>, <4, 2> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>,
  <[ <1, 2>, <2, 2>, <3, 3>, <4, 3> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>
]
> ListGraphMorphisms(H,G);
[
  <[ <1, 1>, <2, 1>, <3, 1> ], [ <<1, 1, 2>, <1, 1, 1>>,
    <<2, 2, 2>, <1, 1, 1>>, <<3, 3, 2>, <1, 1, 1>> ]>
]

```

Note that the graph morphism $f: G \rightarrow H$ from above

```
> f;
<[ <1, 2>, <2, 2>, <3, 1>, <4, 1> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>
```

is the first entry in `ListGraphMorphisms(G,H)`.

Given a graph G , we can get the identity id_G with the following function `Identity`.

```
Identity := function(G);
  return <[<x,x> : x in G[1]], [<x,x> : x in G[2]]>;
end function;
```

For example, for G as above we get

```
> G;
<[ 1, 2, 3, 4 ], [ <1, 1, 1>, <1, 2, 2>, <3, 3, 2> ]>
> Identity(G);
<[ <1, 1>, <2, 2>, <3, 3>, <4, 4> ],
  [ <<1, 1, 1>, <1, 1, 1>>, <<1, 2, 2>, <1, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>
```

With the function `Is_Injective` we can test if a given graph morphism f from a given graph G to a given graph H is injective.

With the function `Is_Surjective` we can test if it is surjective, with the function `Is_Bijective` we can test if it is bijective.

```
Is_Injective := function(m,G,H)
  // m: G -> H: <morphism_on_vertices,morphism_on_edges>
  return #SequenceToSet([u[2] : u in m[1]]) eq #m[1] and
         #SequenceToSet([u[2] : u in m[2]]) eq #m[2];
end function;
```

```
Is_Surjective := function(m,G,H)
  // m: G -> H: <morphism_on_vertices,morphism_on_edges>
  return #SequenceToSet([u[2] : u in m[1]]) eq #H[1] and
         #SequenceToSet([u[2] : u in m[2]]) eq #H[2];
end function;
```

```
Is_Bijective := function(f,G,H)
  return Is_Injective(f,G,H) and Is_Surjective(f,G,H); // %%
end function;
```

For example, given G , H and f as above, we get

```
> Is_Injective(f,G,H);
false
> Is_Surjective(f,G,H);
false
> Is_Bijective(f,G,H);
false
```

With the function `ComposeGraphMorphisms` we can compose given graph morphisms $G \xrightarrow{p} H \xrightarrow{q} I$.

```
ComposeGraphMorphisms := function(p,q)
  v := []; // v for vertices
  for x in p[1] do
    v cat:= [<x[1],y[2]> : y in q[1] | x[2] eq y[1]];
  end for;
  e := []; // e for edges
  for x in p[2] do
    e cat:= [<x[1],y[2]> : y in q[2] | x[2] eq y[1]];
  end for;
  return <v,e>;
end function;
```

For example, let

```
p := ListGraphMorphisms(G,H) [8];
q := ListGraphMorphisms(H,H) [1];
```

Then we get

```
> ComposeGraphMorphisms(p,q);
<[ <1, 2>, <2, 2>, <3, 1>, <4, 2> ],
  [ <<1, 1, 1>, <2, 2, 2>>, <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>
```

The result equals `ListGraphMorphisms(G,H) [2]`.

With the function `IsIsomorphic` we can test if given graphs G and H are isomorphic.

In case of isomorphic graphs, we additionally give a graph isomorphism as output.

```
IsIsomorphic := function(G,H) // G, H: graphs
  if #G[1] ne #H[1] or #G[2] ne #H[2] then
    return <0,false>;
  end if;
  list := ListGraphMorphisms(G,H); // %%
  for m in list do
    if Is_Bijective(m,G,H) then // %%
      return <m,true>;
    end if;
  end for;
  return <0,false>;
end function;
```

For example:

```
> IsIsomorphic(G,H);
<0, false>
```

With the function `IsSubgraph` we can test if a given graph `G` is a subgraph of a given graph `H`.
If so, we additionally give the inclusion morphism as output.

```
IsSubgraph := function(G,H) // returns true if G is a subgraph of H
  for v in G[1] do
    if not v in H[1] then
      return <0,false>;
    end if;
  end for;
  for e in G[2] do
    if not e in H[2] then
      return <0,false>;
    end if;
  end for;
  m := <[<v,v> : v in G[1]], [<e,e> : e in G[2]]>;
  return <m,true>;
end function;
```

A test whether `G` is a full subgraph of `H`:

```
IsFullSubgraph := function(G,H);
  if not IsSubgraph(G,H)[2] then // %%
    return false;
  end if;
  return not &or[e[1] in G[1] and e[3] in G[1] and not e in G[2] : e in H[2]];
end function;
```

For example, for

```
G1 := <[ 1, 2 ], [ <1, 1, 1>, <1, 2, 2> ]>;
H1 := <[ 1, 2, 3 ], [ <1, 1, 1>, <1, 2, 2>, <2, 3, 1>, <2, 4, 2>, <3, 5, 2> ]>;
```

we get

```
> IsSubgraph(G1,H1);
<<[ <1, 1>, <2, 2> ], [ <<1, 1, 1>, <1, 1, 1>>, <<1, 2, 2>, <1, 2, 2>> ]>, true>
> IsFullSubgraph(G1,H1);
false
```

With the function `VtoE` we can complete a given map on vertices from a given graph `G` to a given thin graph `H` to a graph morphism.

If such a graph morphism does not exist, the function prints this out and returns `<0,0>`.

```
VtoE := function(G,H,Vf) // H thin, Vf = [<1,2>,<2,5>,<3,1>,<4,2>] map on
      // vertices
```



```

for e in G[2] do
  if #[h : h in H[2] | h[1] eq [v[2] : v in Vf | v[1] eq e[1]][1] and
      h[3] eq [v[2] : v in Vf | v[1] eq e[3]][1]] eq 0 then
    print "graph morphism does not exist";
    return <0,0>;
  end if;
end for;
Ef := [<e,[h : h in H[2] | h[1] eq [v[2] : v in Vf | v[1] eq e[1]][1] and
      h[3] eq [v[2] : v in Vf | v[1] eq e[3]][1]][1]> : e in G[2]];
return <Vf,Ef>;
end function;

```

E.g. the graph morphism f from above can be obtained as

```
> VtoE(G,H,f[1]);
```

With `ListGraphMorphisms_partial` we get the list of graph morphisms $f: G \rightarrow H$ that obey given partial mapping rules, without having to calculate the whole list of graph morphisms from G to H .

This is useful for the search for quasiisomorphisms.

```

ListGraphMorphisms_partial := function(f,G,H)
// returns all graph morphisms that obey given partial mapping rule f
vertices_partial := f[1];
v1 := [n[1] : n in f[1]];
edges_partial := f[2];
e1 := [n[1] : n in f[2]];
list := [];
for z in G[2] do
  list_new := [];
  if z in e1 then
    H_edges := [n[2] : n in f[2] | n[1] eq z];
  else
    H_edges := H[2];
  end if;
  source := [v : v in H[1] | #[e : e in H[2] | e[1] eq v] ge 1];
  target := [v : v in H[1] | #[e : e in H[2] | e[3] eq v] ge 1];
  if z[1] in v1 then
    source := [n[2] : n in f[1] | n[1] eq z[1]];
  end if;
  if z[3] in v1 then
    target := [n[2] : n in f[1] | n[1] eq z[3]];
  end if;
  H_edges := [h : h in H_edges | h[1] in source and h[3] in target];
  for w in H_edges do
    for x in list do

```

```

    x_test := x cat [<z,w>];
    if IsRightUnique(RVPME(x_test)) then // %%
        list_new cat:= [x_test];
    end if;
end for;
end for;
list := list_new;
end for;
list_mor := [];
for y in list do
    list_completions_to_maps_on_vertices := [Sort(SetToSequence(x)) : x in
        CompletionsToMaps(G[1],H[1],RVPME(y)
        join SequenceToSet(f[1]))]; // %%
    list_mor cat:= [<x,y> : x in list_completions_to_maps_on_vertices];
end for;
return list_mor;
end function;

```

For example, for

```

G := <[ 1, 2, 3, 4 ], [ <1, 1, 1>, <1, 2, 2>, <3, 3, 2> ]>;
H := <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 2>, <3, 3, 2> ]>;

```

as above, we get

```

> ListGraphMorphisms_partial(<[<4,1>], []>,G,H);
[
    <[ <1, 2>, <2, 2>, <3, 1>, <4, 1> ], [ <<1, 1, 1>, <2, 2, 2>>,
    <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <1, 1, 2>> ]>,
    <[ <1, 2>, <2, 2>, <3, 2>, <4, 1> ], [ <<1, 1, 1>, <2, 2, 2>>,
    <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <2, 2, 2>> ]>,
    <[ <1, 2>, <2, 2>, <3, 3>, <4, 1> ], [ <<1, 1, 1>, <2, 2, 2>>,
    <<1, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>
]

```

The function `DisjointUnionCycles` produces a graph isomorphic to a disjoint union of cycles. E.g. for `list := [2,4,5]`, it returns a graph isomorphic to $C_2 \sqcup C_4 \sqcup C_5$.

```

DisjointUnionCycles := function(list) // e.g. list := [2,4,5]
G1 := [i : i in [1..#list]]; // vertices
lists := [[u : u in [1..list[1]]] cat [[u : u in
    [&+[list[k] : k in [1..i-1]]+1..#list[k] : k in [1..i]]]] : i in [2..#list]];
edges := &cat[[<t[i],t[i+1]> : i in [1..#t-1]] cat [<t[#t],t[1]>] : t in lists];
edges_numbered := [<edges[i][1],i,edges[i][2]> : i in [1..#edges]];
G := <G1,edges_numbered>;
return G;
end function;

```

We abbreviate as follows.

```
DUC := DisjointUnionCycles; // %%
```

E.g. we get

```
> DUC([2,4,5]);
<[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ], [ <1, 1, 2>, <2, 2, 1>, <3, 3, 4>,
  <4, 4, 5>, <5, 5, 6>, <6, 6, 3>, <7, 7, 8>,
  <8, 8, 9>, <9, 9, 10>, <10, 10, 11>, <11, 11, 7> ]>
```

With the function DCN we calculate the diagonal graph morphism $d_{C_n} : C_n \sqcup C_n \rightarrow C_n$ for a given n .

```
DCN := function(n)
  G := DUC([n,n]); // %%
  H := C(n); // %%
  f := VtoE(G,H,Sort([<i,i mod n> : i in [1..2*n-1] | not i eq n] cat
    [<n,n>,<2*n,n>]));
  return <G,H,f>;
end function;
```

E.g. we get

```
> DCN(3) [3];
<[ <1, 1>, <2, 2>, <3, 3>, <4, 1>, <5, 2>, <6, 3> ],
  [ <<1, 1, 2>, <1, 1, 2>>, <<2, 2, 3>, <2, 2, 3>>, <<3, 3, 1>, <3, 3, 1>>,
  <<4, 4, 5>, <1, 1, 2>>, <<5, 5, 6>, <2, 2, 3>>, <<6, 6, 4>, <3, 3, 1>> ]>
```

10.3 Calculating a pushout and a pullback of graphs

10.3.1 Calculating a pushout of graphs

The following functions return the equivalence relation that is generated by a given relation R on M . Here a relation on M is a list of pairs with both entries in M .

```
RedSeq := function(S); // S: sequence, to be reduced and sorted
  return Sort(SetToSequence(SequenceToSet(S)));
end function;
```

```
Equivclasses := function(R,M) // R: relation on set M
  Rinv := [<r[2],r[1]> : r in R];
  Diag := [<m,m> : m in M];
  RR := [r : r in R cat Rinv | not r[1] eq r[2]];
  equivclasses := [];
```

```

Mtodo := [m : m in M];
while not #Mtodo eq 0 do
  k := Mtodo[1];
  kclassold := [];
  kclassnew := [k];
  while not #kclassnew eq #kclassold do
    kclassold := kclassnew;
    kclassnew cat:= [j[2] : j in RR | j[1] in kclassold];
    kclassnew := RedSeq(kclassnew); // %%
  end while;
  equivclasses cat:= [kclassnew];
  Mtodo := [u : u in Mtodo | not u in kclassnew];
end while;
return equivclasses;
end function;

```

```

Equivrelation := function(R,M) // R: relation on set M
  equivclasses := Equivclasses(R,M); // %%
  return Sort(&cat[ [<k,l> : k, l in x] : x in equivclasses]);
end function;

```

With the following functions we calculate a pushout in Set.

```

DisjointUnionSets := function(X,Y); // X, Y lists
  return [<1,x> : x in X] cat [<2,y> : y in Y];
end function;

```

```

PushoutSets := function(X,Y,X2,f,g); // f : X -> Y, g : X -> X2 maps
  M := DisjointUnionSets(X2,Y); // %%
  R := [ [<<1, g_elt[2]>, <2, f_elt[2]>> : g_elt in g, f_elt in f |
    g_elt[1] eq x and f_elt[1] eq x] [1] : x in X];
  equiv := Equivclasses(R,M); // %%
  u := [ [<x2,t> : t in equiv | <1,x2> in t] [1] : x2 in X2];
  v := [ [<y,t> : t in equiv | <2, y> in t] [1] : y in Y];
  return <equiv, u, v>; // pushout, u : X2 -> pushout, v : Y -> pushout
end function;

```

With the following function we calculate a pushout in Gph.

```

PushoutGraphs := function(X,Y,X2,f,g); // f : X -> Y, g : X -> X2
  // graph morphisms, returns the pushout
  vertices := PushoutSets(X[1],Y[1],X2[1],f[1],g[1]); // %%
  edges := PushoutSets(X[2],Y[2],X2[2],f[2],g[2]); // %%
  N := [i : i in [1..#vertices[1]]];
  E := [i : i in [1..#edges[1]]]; // edges without source and target
  EE := [ <Index(vertices[1], [n : n in vertices[1] | <edges[1][e][1][1],

```

```

    edges[1][e][1][2][1]> in n[1]),e, Index(vertices[1], [n :
n in vertices[1] | <edges[1][e][1][1],edges[1][e][1][2][3]> in n[1]])> :
e in E];
    // edges with source and target
PP := <N,EE>;
uN := [<x2, Index(vertices[1], [n[2] : n in vertices[2] | x2 eq n[1]][1])> :
x2 in X2[1]]; // <p,p@v>
uE := [<x2, Index(edges[1], [e[2] : e in edges[2] | x2 eq e[1]][1])> :
x2 in X2[2]]; // <p,p@v>, second entry is number of edge
uEE := [ <x[1], <[ee[1] : ee in EE | ee[2] eq x[2]][1],x[2],[ee[3] : ee in EE |
ee[2] eq x[2]][1]>> : x in uE]; // <p,p@u>
    // second entry with source and target
u := <uN,uEE>;
vN := [<y, Index(vertices[1], [n[2] : n in vertices[3] | y eq n[1]][1])> :
y in Y[1]]; // <p,p@v>
vE := [<y, Index(edges[1], [e[2] : e in edges[3] | y eq e[1]][1])> :
y in Y[2]]; // <p,p@v>, , second entry is number of edge
vEE := [ <x[1], <[ee[1] : ee in EE | ee[2] eq x[2]][1],x[2],[ee[3] : ee in EE |
ee[2] eq x[2]][1]>> : x in vE]; // <p,p@u>
    // second entry with source and target
v := <vN,vEE>;
return <PP,u,v>; // pushout, u : X2 -> pushout, v : Y -> pushout
end function;

```

E.g. we get

```

> PushoutGraphs(G,H,H,f,f);
<<[ 1, 2, 3, 4 ], [ <1, 1, 2>, <2, 2, 2>, <3, 3, 2>, <4, 4, 2> ]>,
  <[ <1, 1>, <2, 2>, <3, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <3, 3, 2>> ]>,
  <[ <1, 1>, <2, 2>, <3, 4> ], [ <<1, 1, 2>, <1, 1, 2>>,
  <<2, 2, 2>, <2, 2, 2>>, <<3, 3, 2>, <4, 4, 2>> ]>>

```

10.3.2 Calculating a pullback of graphs

With the following function we calculate a pullback in Set.

```

PullbackSets := function(X,Y,Y2,f,g) // f: X -> Y, g: Y2 -> Y, maps between sets
// given as a list of tuples, e.g. f = [<x_1,y_1>,...,<x_n,y_n>]
P := Sort([<x,y2> : x in X, y2 in Y2 | #[<s,t> : s in f, t in g | x eq s[1] and
y2 eq t[1] and s[2] eq t[2]] eq 1]);
return P;
end function;

```

With the following function we calculate a pullback in Gph.

```

PullbackGraphs := function(X,Y,Y2,f,g)
  // f: X -> Y, g: Y2 -> Y graph morphisms
  vertices := PullbackSets(X[1],Y[1],Y2[1],f[1],g[1]); // %%
  edges := PullbackSets(X[2],Y[2],Y2[2],f[2],g[2]); // %%
  N := [i : i in [1..#vertices]];
  E := [i : i in [1..#edges]]; // edges without source and target
  EE := [<Index(vertices,<edges[e][1][1],edges[e][2][1]>),e,
        Index(vertices,<edges[e][1][3],edges[e][2][3]>)> : e in E];
        // edges with source and target
  PP := <N, EE>;
  vE := [<ee,edges[ee[2]][1]> : ee in EE]; // <p,p@v>
  uE := [<ee,edges[ee[2]][2]> : ee in EE]; // <p,p@u>
  vN := [<n,vertices[n][1]> : n in N]; // <p,p@v>
  uN := [<n,vertices[n][2]> : n in N]; // <p,p@u>
  v := <vN,vE>;
  u := <uN,uE>;
  return <PP,u,v>; // pullback, u: pullback -> Y2, v: pullback -> X
end function;

```

E.g. we get

```

> PullbackGraphs(G,H,G,f,f);
<<[ 1, 2, 3, 4, 5, 6, 7, 8 ], [ <1, 1, 1>, <1, 2, 2>, <1, 3, 3>, <1, 4, 4>,
  <5, 5, 4> ]>,
  <[ <1, 1>, <2, 2>, <3, 1>, <4, 2>, <5, 3>, <6, 4>, <7, 3>, <8, 4> ],
  [ <<1, 1, 1>, <1, 1, 1>>, <<1, 2, 2>, <1, 2, 2>>, <<1, 3, 3>, <1, 1, 1>>,
  <<1, 4, 4>, <1, 2, 2>>, <<5, 5, 4>, <3, 3, 2>> ]>,
  <[ <1, 1>, <2, 1>, <3, 2>, <4, 2>, <5, 3>, <6, 3>, <7, 4>, <8, 4> ],
  [ <<1, 1, 1>, <1, 1, 1>>, <<1, 2, 2>, <1, 1, 1>>, <<1, 3, 3>, <1, 2, 2>>,
  <<1, 4, 4>, <1, 2, 2>>, <<5, 5, 4>, <3, 3, 2>> ]>>

```

10.3.3 Calculating induced morphisms of pushouts and pullbacks of graphs

With the following functions we calculate the induced morphism of a given pushout respectively pullback.

```

InducedMorphismSetsPO := function(X,X2,Y,Y2,u,u2,v,v2)
  // u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
  T := <Y2,u2,v2>;
  P := PushoutSets(X,Y,X2,v,u); // %%
  x2 := [<i,[t[2] : t in T[2] | t[1] in [r[1] : r in P[2] | r[2] eq i]]> :
        i in P[1]];
  y := [<i,[t[2] : t in T[3] | t[1] in [r[1] : r in P[3] | r[2] eq i]]> :

```

```

    i in P[1]];
c := [<r[1],r[2][1]> : r in x2 | not #r[2] eq 0] cat [<r[1],r[2][1]> :
    r in y | not #r[2] eq 0];
return RedSeq(c); // %%
end function;

```

```

InducedMorphismGraphsP0 := function(X,X2,Y,Y2,u,u2,v,v2)
// u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
T := <Y2,u2,v2>;
P := PushoutGraphs(X,Y,X2,v,u); // %%
x2_vertices := [<i,[t[2] : t in T[2][1] | t[1] in [r[1] : r in P[2][1] |
    r[2] eq i]]> : i in P[1][1]];
x2_edges := [<i,[t[2] : t in T[2][2] | t[1] in [r[1] : r in P[2][2] |
    r[2] eq i]]> : i in P[1][2]];
y_vertices := [<i,[t[2] : t in T[3][1] | t[1] in [r[1] : r in P[3][1] |
    r[2] eq i]]> : i in P[1][1]];
y_edges := [<i,[t[2] : t in T[3][2] | t[1] in [r[1] : r in P[3][2] |
    r[2] eq i]]> : i in P[1][2]];
c_vertices := [<r[1],r[2][1]> : r in x2_vertices | not #r[2] eq 0]
    cat [<r[1],r[2][1]> : r in y_vertices | not #r[2] eq 0];
c_edges := [<r[1],r[2][1]> : r in x2_edges | not #r[2] eq 0]
    cat [<r[1],r[2][1]> : r in y_edges | not #r[2] eq 0];
c := <RedSeq(c_vertices),RedSeq(c_edges)>; // %%
return c;
end function;

```

This allows us to decide whether a given commutative quadrangle is a pushout:

```

IsPushoutGraphs := function(X,X2,Y,Y2,u,u2,v,v2)
// u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
p := InducedMorphismGraphsP0(X,X2,Y,Y2,u,u2,v,v2); // %%
return Is_Bijective(p,PushoutGraphs(X,Y,X2,v,u)[1],Y2); // %%
end function;

```

With the following functions we calculate the induced morphism of a given pullback.

```

PullbackSets_num := function(X,Y,Y2,f,g) // f: X -> Y, g: Y2 -> Y
// maps between sets
// given as a list of tuples, e.g. f = [<x_1,y_1>, ..., <x_n,y_n>]
P := Sort([<x,y2> : x in X, y2 in Y2 | #<s,t> : s in f, t in g |
    x eq s[1] and y2 eq t[1] and s[2] eq t[2]] eq 1]);
PP := [i : i in [1..#P]];
v := [<i,P[i][1]> : i in PP]; // <p,p@v>
u := [<i,P[i][2]> : i in PP]; // <p,p@u>
num := [<i,P[i]> : i in [1..#P]];
return <PP,u,v,num>; // pullback PP, u: PP -> Y2, v: PP -> X
end function;

```

```

PullbackGraphs_num := function(X,Y,Y2,f,g) // f: X -> Y, g: Y2 -> Y
    // graph morphisms
    vertices := PullbackSets(X[1],Y[1],Y2[1],f[1],g[1]); // %%
    edges := PullbackSets(X[2],Y[2],Y2[2],f[2],g[2]); // %%
    N := [i : i in [1..#vertices]];
    E := [i : i in [1..#edges]]; // edges without source and target
    EE := [<Index(vertices,<edges[e][1][1],edges[e][2][1]>),e,
        Index(vertices,<edges[e][1][3],edges[e][2][3]>)> : e in E];
    // edges with source and target
    PP := <N, EE>;
    vE := [<ee,edges[ee[2]][1]> : ee in EE]; // <p,p@v>
    uE := [<ee,edges[ee[2]][2]> : ee in EE]; // <p,p@u>
    vN := [<n,vertices[n][1]> : n in N]; // <p,p@v>
    uN := [<n,vertices[n][2]> : n in N]; // <p,p@u>
    v := <vN,vE>;
    u := <uN,uE>;
    numvertices := [<i,vertices[i]> : i in [1..#vertices]];
    numedges := [<i,edges[i]> : i in [1..#edges]];
    num := <numvertices,numedges>;
    return <PP,u,v,num>; // pullback PP, u: PP -> Y2, v: PP -> X
end function;

InducedMorphismSetsPB := function(X,X2,Y,Y2,u,u2,v,v2)
    // u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
    T := <X,v,u>;
    P := PullbackSets_num(Y,Y2,X2,v2,u2); // %%
    c_underscore := [<t,<[r[2] : r in T[2] | r[1] eq t][1],[r[2] : r in T[3] |
        r[1] eq t][1]>> : t in T[1]];
    c := [<r[1],Index([p[2] : p in P[4]],r[2])> : r in c_underscore];
    return c;
end function;

InducedMorphismGraphsPB := function(X,X2,Y,Y2,u,u2,v,v2)
    // u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
    T := <X,v,u>;
    P := PullbackGraphs_num(Y,Y2,X2,v2,u2); // %%
    c_underscore_vertices := [<t,<[r[2] : r in T[2][1] | r[1] eq t][1],[r[2] :
        r in T[3][1] | r[1] eq t][1]>> : t in T[1][1]];
    c_underscore_edges := [<t,<[r[2] : r in T[2][2] | r[1] eq t][1],[r[2] :
        r in T[3][2] | r[1] eq t][1]>> : t in T[1][2]];
    c_vertices := [<r[1],Index([p[2] : p in P[4][1]],r[2])> :
        r in c_underscore_vertices];
    c_edges := [<r[1],Index([p[2] : p in P[4][2]],r[2])> : r in c_underscore_edges];
    c := <c_vertices,c_edges>;
    return c;
end function;

```


This allows us to decide whether a given commutative quadrangle is a pullback:

```
IsPullbackGraphs := function(X,X2,Y,Y2,u,u2,v,v2)
  // u: X -> X2, u2: X2 -> Y2, v: X -> Y, v2: Y -> Y2
  p := InducedMorphismGraphsPB(X,X2,Y,Y2,u,u2,v,v2); // %%
  return Is_Bijective(p,X,PullbackGraphs_num(Y,Y2,X2,v2,u2)[1]); // %%
end function;
```

10.4 Calculating tree graphs

With the following functions we calculate the graph $\text{Tree}(x, X)$ at a given vertex $x \in V_X$ of a given graph X if $\text{Tree}(x, X)$ is finite.

The function `CyclesFromVertex` returns true if cycles in a given graph G exist that contain a given vertex x .

```
CyclesFromVertex := function(x,G) // G graph, x vertex in G
  if #[a : a in G[2] | a[1] eq x] eq 0 then
    return false;
  end if;
  S := {}; // S: set of vertices to achieve
  Snew := {x};
  while not #Snew eq #S do
    S := Snew;
    Snew join:= {a[3] : a in G[2] | a[1] in S};
    if x in {a[3] : a in G[2] | a[1] in S} then
      return true;
    end if;
  end while;
  return false;
end function;
```

For example, for the graph

```
G2 := <[ 1, 2, 3 ], [ <2, 1, 2>, <3, 2, 2>, <2, 3, 3>, <1, 4, 2> ]>;
```

the function yields

```
> CyclesFromVertex(1,G2);
false
> CyclesFromVertex(2,G2);
true
> CyclesFromVertex(3,G2);
true
```

Given a graph G and a vertex r of G , the function `VerticesToAchieve` returns the list of vertices v in G for which there exists a path from r to v .

```
VerticesToAchieve := function(G,r) // G: graph, r vertex in G
  if #[a : a in G[2] | a[1] eq r] eq 0 then
    return [];
  end if;
  S := {}; // S: set of vertices to achieve
  Snew := {r};
  while not #Snew eq #S do
    S := Snew;
    Snew join:= {a[3] : a in G[2] | a[1] in S};
  end while;
  return Sort(SetToSequence(Snew));
end function;
```

E.g. we get

```
> VerticesToAchieve(G2,1);
[ 1, 2, 3 ]
> VerticesToAchieve(G2,2);
[ 2, 3 ]
> VerticesToAchieve(G2,3);
[ 2, 3 ]
```

The following function `CyclesInPathFromx` returns `true` if some path from a given vertex x in a given graph G can be restricted to a graph morphism to a cyclic graph C_n for some n .

```
CyclesInPathFromx := function(x,G)
  return &or[CyclesFromVertex(v,G) : v in VerticesToAchieve(G,x)];
end function;
```

E.g. we get

```
> CyclesInPathFromx(1,G2);
true
```

The function `Paths` returns the list of paths in G starting in x if there are finitely many. Such a path is given as a list of edges.

```
Paths := function(x,G) // G graph, x vertex in G[1],
  if not CyclesInPathFromx(x,G) then // %%
    Listofpaths := [[]];
    if not #[[[e] : e in G[2] | e[1] eq x]] eq 0 then
      Listofpaths cat:= [[[e] : e in G[2] | e[1] eq x]];
    else
```

```

    return &cat(Listofpaths);
end if;
newpaths := [<p,[e : e in G[2] | e[1] eq p[#p][3]]> :
             p in Listofpaths[#Listofpaths] | not #[e : e in G[2] |
             e[1] eq p[#p][3]] eq 0];
while not #newpaths eq 0 do
  Listofpaths cat:= [&cat[&cat[[n[1] cat [n[2][i]]] : i in [1..#n[2]]] :
                    n in newpaths]];
  newpaths := [<p,[e : e in G[2] | e[1] eq p[#p][3]]> :
              p in Listofpaths[#Listofpaths] | not #[e : e in G[2] |
              e[1] eq p[#p][3]] eq 0];
end while;
return Listofpaths;
end if;
return "infinite";
end function;

```

The function `TreeOfPaths` returns the graph $\text{Tree}(x, G)$ in case it is finite, for a given graph G and one of its vertices x .

```

TreeOfPaths := function(x,G) // G graph, x vertex in G
if not CyclesInPathFromx(x,G) then // %%
  P := Paths(x,G); // %%
  V := &cat(P);
  E := [<[],<[],p[1],p>,p> : p in P[2]];
  E cat:= [<[p[i] : i in [1..#p-1]],<[p[i] : i in [1..#p-1]],p[#p],p>,p> :
          p in V | #p ge 2];
  return <V,E>;
end if;
return "infinite";
end function;

```

For example, for

```

G3 := [<[ 1, 2, 3 ], [ <1, "a", 2>, <2, "b", 3>, <1, "c", 3> ]>;
C(3);

```

we get

```

> Paths(1,G3);
[
  [
    []
  ],
  [
    [ <1, "a", 2> ],
    [ <1, "c", 3> ]
  ]
]

```

```

    ],
    [
      [ <1, "a", 2>, <2, "b", 3> ]
    ]
  ]
> Paths(1,C(3));
infinite
> TreeOfPaths(1,G3);
<[
  [],
  [ <1, "a", 2> ],
  [ <1, "c", 3> ],
  [ <1, "a", 2>, <2, "b", 3> ]
], [
  <[], <[], <1, "a", 2>, [ <1, "a", 2> ]>, [ <1, "a", 2> ]>,
  <[], <[], <1, "c", 3>, [ <1, "c", 3> ]>, [ <1, "c", 3> ]>,
  <[ <1, "a", 2> ],
  <[ <1, "a", 2> ], <2, "b", 3>, [ <1, "a", 2>, <2, "b", 3> ]>,
  [ <1, "a", 2>, <2, "b", 3> ]>]
> TreeOfPaths(1,C(3));
infinite

```

With the function `IsTreeDef` we can calculate whether a given graph G is a tree using Definition 108.

```

IsTreeDef := function(G)
  for r in G[1] do // searching root
    if #[v : v in [g : g in G[1] | not g eq r] | not #[e : e in G[2] |
      e[3] eq v] eq 1] eq 0 then // (Tree 1)
      if #[e : e in G[2] | e[3] eq r] eq 0 then // (Tree 2)
        if &and[v in VerticesToAchieve(G,r) : v in G[1]] then // (Tree 3) // %%
          return true;
        end if;
      end if;
    end if;
  end for;
  return false;
end function;

```

Cf. the function `IsTree` in §10.5 below.

10.5 Testing properties of graph morphisms

With the following functions we test properties of graph morphisms as discussed in §3.

```
ListOfnCycles := function(G,n) // G: graph
  return ListGraphMorphisms(C(n),G); // %%
end function;
```

The function `Cnf_Bij` returns `true` if $(C_n, f)_{\text{Gph}}$ is bijective for a given graph morphism $f: G \rightarrow H$ and for a given number n .

```
Cnf_Bij := function(f,G,H,n) // G, H: graphs, f: G -> H: graph morphism
  if not #ListOfnCycles(G,n) eq #ListOfnCycles(H,n) then // %%
    return false;
  end if;
  if SequenceToSet([ComposeGraphMorphisms(ListOfnCycles(G,n)[i],f) : i in
    [1..#ListOfnCycles(G,n)]]) eq SequenceToSet(ListOfnCycles(H,n)) then
    return true; // %%
  end if;
  return false;
end function;
```

Given a graph morphism $f: G \rightarrow H$ and an upper bound `ub`, the function `IsQis_Bound` returns `true` if the map $(C_k, f)_{\text{Gph}}$ is bijective for $k \leq \text{ub}$.

```
IsQis_Bound := function(f,G,H,ub) // ub: upper bound
  i := 1;
  while Cnf_Bij(f,G,H,i) and i le ub do // %%
    i := i+1;
  end while;
  return i eq ub+1;
end function;
```

With the function `IsFibration` we can calculate whether a given graph morphism $f: G \rightarrow H$ is a fibration or not; cf. Definition 127.(1).

```
IsFibration := function(f,G,H) // G, H: graphs, f: G -> H graph morphism
  for x in G[1] do
    y := [a[2] : a in f[1] | a[1] eq x][1];
    for b in [h : h in H[2] | h[1] eq y] do
      if #[0 : a in G[2] | <a,b> in f[2] and a[1] eq x] eq 0 then
        return false;
      end if;
    end for;
  end for;
  return true;
end function;
```

With the function `IsEtaleFibration` we can calculate whether a given graph morphism $f: G \rightarrow H$ is an étale fibration or not; cf. Definition 127.(2).

```
IsEtaleFibration := function(f,G,H) // G, H: graphs, f: G -> H graph morphism
  for x in G[1] do
    y := [a[2] : a in f[1] | a[1] eq x][1];
    if not #[e : e in G[2] | e[1] eq x] eq #[h : h in H[2] | h[1] eq y] then
      return false;
    elif not Sort([a[2] : a in f[2] | a[1][1] eq x])
      eq Sort([h : h in H[2] | h[1] eq y]) then
      return false;
    end if;
  end for;
  return true;
end function;
```

For example, given

```
G := <[1,2,3],[<1,1,2>,<1,2,3>]>;
f := VtoE(G,D(1),[<1,0>,<2,1>,<3,1>]);
```

we get

```
> IsFibration(f,G,D(1));
true
> IsEtaleFibration(f,G,D(1));
false
```

With the function `IsFibrant` we can calculate whether a given graph X is fibrant or not; cf. Definition 135.

```
IsFibrant := function(X)
  return &and[not #[e : e in X[2] | e[1] eq v] eq 0: v in X[1]];
end function;
```

For example, we get

```
> IsFibrant(C(2));
true
> IsFibrant(D(2));
false
```

With the functions `AcCofib1to4` and `AcCofib5` we can check if a given graph morphism $f: G \rightarrow H$ satisfies the properties (`AcCofib 1–4`) respectively (`AcCofib 5`).

The function `IsAcCofib` checks all properties (`AcCofib 1–5`), i.e. it decides whether f is an acyclic cofibration.

```

AcCofib1to4 := function(f,G,H) // f: G -> H graph morphism
  if #[0 : x in H[1] | #[0 : a in G[1] | <a,x> in f[1]] ge 2] ge 1 then
    // (AcCofib 1)
    return false;
  end if;
  if #[0 : x in H[2] | #[0 : a in G[2] | <a,x> in f[2]] ge 2] ge 1 then
    // (AcCofib 2)
    return false;
  end if;
  HH1 := [x : x in H[1] | #[0 : a in G[1] | <a,x> in f[1]] eq 0];
  HH2 := [x : x in H[2] | #[0 : a in G[2] | <a,x> in f[2]] eq 0];
  if #[0 : x in HH1 | not #[a : a in H[2] | a[3] eq x] eq 1] ge 1 then
    // (AcCofib 3)
    return false;
  end if;
  if #[0 : x in HH2 | not x[3] in HH1] ge 1 then
    // (AcCofib 4)
    return false;
  end if;
  return true;
end function;

AcCofib5 := function(f,G,H)
  HH1 := [x : x in H[1] | #[0 : a in G[1] | <a,x> in f[1]] eq 0];
  max := #H[2];
  L := [];
  for i in [1..max] do
    L cat:= [ListGraphMorphisms(D(i),H)]; // %%
  end for;
  for v in HH1 do
    list := [];
    for i in [1..max] do
      list cat:= [l : l in L[i] | not l[1][1][2] in HH1 and l[1][#[1][1]][2] eq v];
    end for;
    if #list eq 0 then
      return false;
    end if;
  end for;
  return true;
end function;

IsAcCofib := function(f,G,H)
  return AcCofib1to4(f,G,H) and AcCofib5(f,G,H);
end function;

```

For example, for $f := \text{VtoE}(D(0), D(2), [<0, 0 >])$;

we get

```
> IsAcCofib(f,D(0),D(2));
true
```

For example, for

```
G := <[], []>;
f := <[], []>;
```

we get

```
> AcCofib1to4(f,G,C(1));
true
> AcCofib5(f,G,C(1)); // %%
false
```

With the function `IsTree` we can calculate whether a given graph G is a tree using Remark 178.

```
IsTree := function(G)
  for x in G[1] do
    if IsAcCofib(VtoE(D(0),G,[<0,x>]),D(0),G) then // %%
      return true;
    end if;
  end for;
  return false;
end function;
```

10.6 Testing the sufficient condition of Proposition 210 for graph morphisms

The function `Unitargeting` returns the unitargeting edges in H with respect to f .

```
Unitargeting := function(f,G,H)
  return [e : e in H[2] | #RedSeq([ee[1][3] : ee in f[2] | ee[2] eq e]) eq 1];
  // %%
end function;
```

For example, using functions from §10.7, for

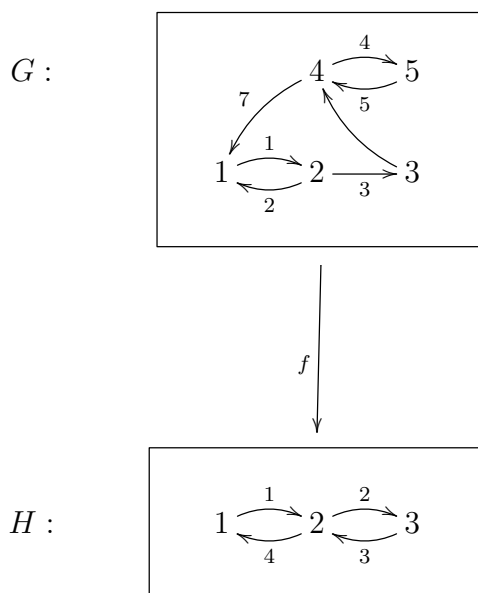
```
G := trygraph(3);
H := c2chain(3);
f := tryacyclic(3);
```


we get

```

> G;
<[ 1, 2, 3, 4, 5 ], [ <1, 1, 2>, <2, 2, 1>, <2, 3, 3>, <4, 4, 5>, <5, 5, 4>,
      <3, 6, 4>, <4, 7, 1> ]>
> H;
<[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>
> f;
<[ <1, 1>, <2, 2>, <3, 3>, <4, 2>, <5, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
      <<2, 2, 1>, <2, 4, 1>>, <<2, 3, 3>, <2, 2, 3>>, <<4, 4, 5>, <2, 2, 3>>,
      <<5, 5, 4>, <3, 3, 2>>, <<3, 6, 4>, <3, 3, 2>>, <<4, 7, 1>, <2, 4, 1>> ]>
> Unitargeting(f,G,H);
[ <1, 1, 2>, <3, 3, 2>, <2, 4, 1> ]

```



The function `Uni` tests if the property (Uni) holds for the given graph morphism $f: G \rightarrow H$.

```

Uni := function(f,G,H)
  U := Unitargeting(f,G,H); // %%
  HH := <H[1],[e : e in H[2] | not e in U]>; // H without unitargeting edges
  n := Minimum([#HH[1],#HH[2]]);
  return &and[#ListGraphMorphisms(C(i),HH) eq 0 : i in [1..n]]; // %%
end function;

```

With the function `SuffCond` we test if our sufficient condition for a morphism to be a quasi-isomorphism holds; cf. Proposition 210.

```

SuffCond := function(f,G,H)
  return <IsEtaleFibration(f,G,H),Uni(f,G,H)>; // %%
end function;

```

For example, we get

```
> Uni(f,G,H);
true
> SuffCond(f,G,H);
<true, true>
```

For

```
G := <[ 1, 2, 3 ], [ <2, 1, 2>, <3, 2, 1>, <3, 3, 1> ]>;
H := <[ 1, 2, 3 ], [ <2, 1, 2>, <3, 2, 1>, <1, 3, 3> ]>;
L := ListGraphMorphisms(G,H);
M := ListGraphMorphisms(H,G);
```

we get

```
> M;
[
  <[ <1, 2>, <2, 2>, <3, 2> ], [ <<2, 1, 2>, <2, 1, 2>>,
    <<3, 2, 1>, <2, 1, 2>>, <<1, 3, 3>, <2, 1, 2>> ]>
]
> L;
[
  <[ <1, 2>, <2, 2>, <3, 2> ], [ <<2, 1, 2>, <2, 1, 2>>,
    <<3, 2, 1>, <2, 1, 2>>, <<3, 3, 1>, <2, 1, 2>> ]>,
  <[ <1, 1>, <2, 2>, <3, 3> ], [ <<2, 1, 2>, <2, 1, 2>>,
    <<3, 2, 1>, <3, 2, 1>>, <<3, 3, 1>, <3, 2, 1>> ]>,
  <[ <1, 3>, <2, 2>, <3, 1> ], [ <<2, 1, 2>, <2, 1, 2>>,
    <<3, 2, 1>, <1, 3, 3>>, <<3, 3, 1>, <1, 3, 3>> ]>
]
> SuffCond(M[1],H,G);
<true, false>
> SuffCond(L[1],G,H);
<false, false>
> SuffCond(L[2],G,H);
<false, true>
> SuffCond(L[3],G,H);
<false, true>
```

10.7 Functions to calculate examples in §9.1

With the following functions we calculated the examples mentioned in §9.1.

```

c2chain := function(n)
  edges := [<i,i,i+1> : i in [1..n-1]] cat [<i+1,2*n-1-i,i> : i in [1..n-1]];
  edges_tosort := Sort([<e[2],e[1],e[3]> : e in edges]);
  edges := [<e[2],e[1],e[3]> : e in edges_tosort];
  return [<i : i in [1..n]>],edges>;
end function;

trygraph := function(n) // n geq 3
  edges := [<3*n-5,3*n-5,3*n-4>,<3*n-4,3*n-4,3*n-5>,<3*n-5,6*n-11,3*n-8>,<3*n-6,6*n-12,3*n-5>];
  edges cat:= &cat[ [<3*k-2,3*k-2,3*k-1>,<3*k-1,3*k-1,3*k-2>,<3*k-1,3*k,3*k>] :
    k in [1..n-2]]; // innerhalb der Stufen
  edges cat:= &cat[ [<3*k+1,3*n-5+3*k,3*k-2>,<3*k,3*n-6+3*k,3*k+1>,<3*k,3*n-4+3*k,3*k+3>] : k in [1..n-3]];
  edges_tosort := Sort([<e[2],e[1],e[3]> : e in edges]);
  edges := [<e[2],e[1],e[3]> : e in edges_tosort];
  G := [<i : i in [1..3*n-4]>],edges>;
  return G;
end function;

tryacyclic := function(n) // trygraph -> c2chain
  vertices := [<1,1>,<2,2>,<4,2>,<3*n-6,n>,<3*n-4,n>];
  vertices cat:= &cat[ [<3*k-6,k>,<3*k-4,k>,<3*k-2,k>] : k in [3..n-1]];
  if n eq 3 then
    edges := [<<1,1,2>,<1,1,2>>,<<2,2,1>,<2,2*n-2,1>>,<<2,3,3>,<2,2,3>>,<<4,4,5>,<2,2,3>>,<<5,5,4>,<3,2*n-3,2>>,<<3,3*n-3,4>,<3,2*n-3,2>>,<<4,3*n-2,1>,<2,2*n-2,1>>];
  else
    edges := [<<1,1,2>,<1,1,2>>,<<2,2,1>,<2,2*n-2,1>>,<<2,3,3>,<2,2,3>>,<<4,4,5>,<2,2,3>>,<<5,5,4>,<3,2*n-3,2>>,<<3,3*n-3,4>,<3,2*n-3,2>>,<<4,3*n-2,1>,<2,2*n-2,1>>,<<7,3*n+1,4>,<3,2*n-3,2>>,<<3*n-7,3*n-6,3*n-6>,<n-1,n-1,n>>,<<3*n-5,3*n-5,3*n-4>,<n-1,n-1,n>>,<<3*n-4,3*n-4,3*n-5>,<n,n,n-1>>,<<3*n-9,6*n-13,3*n-6>,<n-1,n-1,n>>,<<3*n-6,6*n-12,3*n-5>,<n,n,n-1>>];
  edges cat:= &cat[ [<<3*k-4,3*k-3,3*k-3>,<k,k,k+1>>,<<3*k-2,3*k-2,3*k-1>,<k,k,k+1>>,<<3*k-1,3*k-1,3*k-2>,<k+1,2*n-1-k,k>>,<<3*k-6,3*n-10+3*k,3*k-3>,<k,k,k+1>>,<<3*k-3,3*n-9+3*k,3*k-2>,<k+1,2*n-1-k,k>>,<<3*k+1,3*n-5+3*k,3*k-2>,<k+1,2*n-1-k,k>>] : k in [3..n-2]];
  end if;
  edges_tosort := Sort([<<e[1][2],e[1][1],e[1][3]>,e[2]> : e in edges]);
  edges := [<<e[1][2],e[1][1],e[1][3]>,e[2]> : e in edges_tosort];
  return <Sort(vertices),edges>;
end function;

```

```

glue_vertices := function(G,list) // e.g. list := [[2,3],[1,4,5]] list of
// sublists of vertices to glue
list := [RedSeq(1) : 1 in list | #RedSeq(1) ge 2]; // %%
vertices_to_glue := RedSeq(&cat(list)); // %%
vertices_left_over := [n : n in G[1] | not n in vertices_to_glue];
vertices := Sort([l[1] : 1 in list] cat vertices_left_over);
edges_1 := [e : e in G[2] | not e[1] in vertices_to_glue
            and not e[3] in vertices_to_glue];
edges := [];
for e in G[2] do
  if not e[1] in vertices_to_glue and not e[3] in vertices_to_glue then
    edges cat:= [e];
  else if not e[1] in vertices_to_glue then
    edges cat:= [<e[1],e[2],[l[1] : 1 in list | e[3] in l][1]>];
  else
    edges cat:= [<[l[1] : 1 in list | e[1] in l][1],e[2],e[3]>];
  end if;
end if;
end for;
edges_named := [];
for e in edges do
  if e[1] in &cat(list) then
    e1_new := [l : 1 in list | e[1] in l][1][1];
  else
    e1_new := e[1];
  end if;
  if e[3] in &cat(list) then
    e3_new := [l : 1 in list | e[1] in l][1][1];
  else
    e3_new := e[3];
  end if;
  edges_named cat:= [<e1_new,e[2],e3_new>];
end for;
return <vertices,edges_named>;
end function;

```

```

glue_vertices_including_edges := function(G,list)
// e.g. list := [[2,3],[1,4,5]] list of sublists of vertices to glue
G := glue_vertices(G,list); // %%
edges := [];
for g in G[2] do
  if #[e : e in edges | e[1] eq g[1] and e[3] eq g[3]] eq 0 then
    edges cat:= [g];
  end if;
end for;
return <G[1],edges>;
end function;

```

For example, for

```
G := <[1,2,3],[<1,1,2>]>;
H := <[1,2,3],[<1,1,2>,<1,2,2>,<2,3,3>,<2,4,3>]>;
```

we get

```
> glue_vertices(G,[[1,2]]);
<[ 1, 3 ], [ <1, 1, 1> ]>
> glue_vertices(H,[[1,2]]);
<[ 1, 3 ], [ <1, 1, 1>, <1, 2, 1>, <1, 3, 3>, <1, 4, 3> ]>
> glue_vertices_including_edges(H,[[1,2]]);
<[ 1, 3 ], [ <1, 1, 1>, <1, 3, 3> ]>
```

```
try_id_vertices := function(n) // n ge 3
  return [[3*i,3*i+2] : i in [1..n-2]];
end function;

idtrygraph := function(n)
  return glue_vertices_including_edges(trygraph(n),try_id_vertices(n));
  // %%
end function;

tryfactorization := function(n) // trygraph --> idtrygraph
  T := try_id_vertices(n); // %%
  G := trygraph(n); // %%
  vertices := Sort(<[t[2],t[1]> : t in T] cat [<i,i> : i in [1..3*n-4] |
    not i in [t[2] : t in T]]);
  edges_to_map := trygraph(n)[2];
  edges_images := idtrygraph(n)[2];
  edges := [];
  for e in edges_to_map do
    if e in edges_images then
      im_e := e;
    else
      im_e1 := [n[2] : n in vertices | n[1] eq e[1]][1];
      im_e3 := [n[2] : n in vertices | n[1] eq e[3]][1];
      im_e := [edge : edge in edges_images |
        edge[1] eq im_e1 and edge[3] eq im_e3][1];
    end if;
    edges cat:= [<e,im_e>];
  end for;
  return <vertices,edges>;
end function;
```

```

idtryacyclic := function(n) // idtrygraph -> c2chain
  T := try_id_vertices(n); // %%
  T2 := [t[2] : t in try_id_vertices(n)]; // %%
  G := idtrygraph(n); // %%
  f := tryacyclic(n); // %%
  vertices := [n : n in f[1] | n[1] in G[1]];
  edges := [];
  for e in f[2] do
    if e[1] in G[2] then
      edges cat:= [e];
    else
      e1 := e[1][1];
      e3 := e[1][3];
      if e[1][1] in T2 then
        e1 := [t[1] : t in T | t[2] eq e[1][1]][1];
      end if;
      if e[1][3] in T2 then
        e3 := [t[1] : t in T | t[2] eq e[1][3]][1];
      end if;
      edges cat:= [<<e1,e[1][2],e3>,e[2]>];
    end if;
  end for;
  edges2 := [];
  for e in edges do
    if not <e[1][1],e[1][3]> in [<e[1][1],e[1][3]> : e in edges2] then
      edges2 cat:= [e];
    end if;
  end for;
  return <vertices,edges2>;
end function;

Doublecyclic := function(n)
  C := c2chain(n); // %%
  return <C[1],C[2] cat [<1,#C[2]+1,#C[1]>] cat [<#C[1],#C[2]+2,1>]>;
end function;

Trygraph := function(n)
  T := trygraph(n); // %%
  return <T[1],T[2] cat [<1,#T[2]+1,#T[1]>, <#T[1],#T[2]+2,1>, <#T[1]-2,#T[2]+3,1>]>;
end function;

Tryacyclic := function(n) // Trygraph -> Doublecyclic
  T := Trygraph(n); // %%
  t := #T[2];
  D := Doublecyclic(n); // %%
  d := #D[2];
  f := tryacyclic(n); // %%

```

```

return <f[1],f[2] cat [<T[2][t-2],D[2][d-1]>,
<T[2][t-1],D[2][d]>, <T[2][t],D[2][d]>]>;
end function;

```

```

idTrygraph := function(n)
T := idtrygraph(n); // %%
t := [r[2] : r in T[2]];
return <T[1],T[2] cat [<1,t[#t]+1,T[1][#T[1]-1]>]
cat [<T[1][#T[1]-1],t[#t]+2,1>]>;
end function;

```

```

Tryfactorization := function(n) // Trygraph -> idTrygraph
T := trygraph(n); // %%
TT := idtrygraph(n); // %%
t := [r[2] : r in TT[2]];
f := tryfactorization(n); // %%
return <f[1],f[2] cat [<<1,#T[2]+1,#T[1]>,<1,t[#t]+1,TT[1][#TT[1]-1]>>]
cat [<<#T[1],#T[2]+2,1>,<TT[1][#TT[1]-1],t[#t]+2,1>>]
cat [<<#T[1]-2,#T[2]+3,1>,<TT[1][#TT[1]-1],t[#t]+2,1>>]>;
end function;

```

```

idTryacyclic := function(n) // idTrygraph -> Doublecyclic
f := idtryacyclic(n); // %%
T := idtrygraph(n); // %%
t := [r[2] : r in T[2]];
C := c2chain(n); // %%
return <f[1],f[2] cat [<<1,t[#t]+1,T[1][#T[1]-1]>,<1,#C[2]+1,#C[1]>>]
cat [<<T[1][#T[1]-1],t[#t]+2,1>,<#C[1],#C[2]+2,1>>]>;
end function;

```

```

cncm := function(n,m) // cn glued to cm at vertex n
V := [i : i in [1..n+m-1]];
E := [<i,i,i+1> : i in [1..n-1]] cat [<n,n,1>] cat [<n+i,n+i+1,n+i+1> :
i in [0..m-2]] cat [<n+m-1,n+m,n>];
return <V,E>;
end function;

```

```

CnCm := function(n,m)
V := [i : i in [1..n+m]];
E := [<i,i,i+1> : i in [1..n+m-1]] cat [<n+m,n+m,1>]
cat [<n,n+m+1,1>,<n+m,n+m+2,n+1>];
return <V,E>;
end function;

```

```

cncmqis := function(n,m)
  return VtoE(CnCm(n,m),cncm(n,m),[<i,i> : i in [1..n+m-1]] cat [<n+m,n>]);
  // %%
end function;

```

```

CNCN := function(n)
  G := DUC([n,n]); // %%
  H := <G[1],G[2] cat [<n+1,2*n+1,1>,<2*n,2*n+2,n>]>;
  f := VtoE(G,H,[<i,i> : i in [1..2*n]]); // %%
  return <G,H,f>;
end function;

```

```

c2graph := function(list) // list e.g. [<1,3>,<2,4>,<6,6>] list of tuples
  // of vertices
  n := Maximum([x[1] : x in list] cat [x[2] : x in list]);
  edges := [];
  for x in list do
    i := Index(list,x);
    edges cat:= [<x[1],i,x[2]>,<x[2],2*#list-i+1,x[1]>];
  end for;
  edges_tosort := Sort([<e[2],e[1],e[3]> : e in edges]);
  edges := [<e[2],e[1],e[3]> : e in edges_tosort];
  return <[i : i in [1..n]],edges>;
end function;

```

```

Exflower := function(n,list) // list := [2,3,5] contains vertices
  // that are connected with "upper" vertex 1 in G
  edges_G := Sort(&cat[ [<1,i>,<n+2,i>] : i in [2..n+1]]);
  if #list eq 0 then // alle unten
    edges_G cat:= [<i,n+2> : i in [2..n+1]];
    edges_G := [<edges_G[i][1],i,edges_G[i][2]> : i in [1..#edges_G]];
  else
    edges_G cat:= [<i,1> : i in list];
    edges_G cat:= [<i,n+2> : i in [2..n+1] | not i in list];
    edges_G := [<edges_G[i][1],i,edges_G[i][2]> : i in [1..#edges_G]];
  end if;
  flowerG := <[i : i in [1..n+2]],edges_G>;
  flowerH := c2graph([<1,i> : i in [2..n+1]]); // %%
  flowerf := VtoE(flowerG,flowerH,[<i,i> : i in [1..n+1]] cat [<n+2,1>]);
  // %%
  return <flowerG,flowerH,flowerf>;
end function;

```

```

Exflower2 := function(n,k)
  // n: number of "petals", k: size of "petals"
  G := DUC([k : i in [1..n]]); // %%
  VG := G[1];

```



```

EG := G[2];
V := [1+i*k : i in [0..n-1]];
EG2 := &cat[[<v,vv> : vv in [vv+1 : vv in [vv : vv in V | not vv eq v]]] :
      v in V];
EG2 := [<e[1],#EG+Index(EG2,e),e[2]> : e in EG2];
flowerG := <VG,EG cat EG2>;
VH := [i : i in [1..n*(k-1)+1]];
EH := [<1,1+i*k,2+i*(k-1)> : i in [0..n-1]];
if not k eq 2 then
  for i in [1..n] do
    EH cat:= [<1+(i-1)*(k-1)+1,1+(i-1)*k+1,2+(i-1)*(k-1)+1> : l in [1..k-2]];
  end for;
end if;
EH cat:= [<k+i*(k-1),k*(i+1),1> : i in [0..n-1]];
EH_sort := [<e[2],e[1],e[3]> : e in Sort([<e[2],e[1],e[3]> : e in EH])];
flowerH := <VH,EH_sort>; // %%
VtoE_vertices := [<v,1> : v in V] cat &cat[[<1+(i-1)*k+1,2+(i-1)*k+1-i> :
      l in [1..k-1]] : i in [1..n]];
flowerf := VtoE(flowerG,flowerH,VtoE_vertices); // %%
return <flowerG,flowerH,flowerf>;
end function;

```

10.8 Functions to calculate more examples

```

exampleforbadbound := function(n) // (c_2n,f) first not to be bijective
  V := [i : i in [1..2*n+1]];
  E := [<i,i+1,i+1> : i in [3..2*n]];
  E cat:= [<1,1,2>, <2,2,1>, <1,3,3>, <2*n+1,2*n+2,1>];
  X := <V,E>;
  f := ListGraphMorphisms(X,C(2))[2]; // %%
  return <X,f>;
end function;

```

```

exampleforbadbound2 := function(n) // (c_n,f) first not to be bijective
  G := DUC([1,n]); // %%
  f := [<v,1> : v in G[1]], [<e,<1,1,1>> : e in G[2]];
  return <G,C(1),f>;
end function;

```

```

exampleforbadbound3 := function(n) // n even, n ge 4
  G := DUC([2,n-2,n]); // %%
  H := [<i : i in [1..n-1]], [<i,i,i+1> : i in [1..n-2]]
      cat [<n-1,n-1,n-2>, <n-2,n,1>];
  u_vertices := [<1,n-1>, <2,n-2>] cat [<i,i-2> : i in [3..n]]
      cat [<i,i-n> : i in [n+1..2*n-1]] cat [<2*n,n-2>];

```

```

u_edges := [<c,[h : h in H[2] | h[1] eq [u[2] : u in u_vertices |
  u[1] eq c[1]][1] and h[3] eq [u[2] : u in u_vertices |
  u[1] eq c[3]][1]][1]> : c in G[2]];
u := <u_vertices,u_edges>;
return <G,H,u>;
end function;

```

```

DTB := function(n) // decimal to binary
  A := [];
  if n eq 0 then
    return [0];
  end if;
  while n gt 0 do
    A cat:= [n mod 2];
    n := Integers()!((n-(n mod 2))/2);
  end while;
  return A;
end function;

```

```

DTO := function(list) // decimal to other, e.g. list := [2,4,3]
  W := [[i : i in [1..list[1]]]];
  for i in [2..#list] do
    W cat:= [[u : u in [W[#W][#W[#W]]+1..W[#W][#W[#W]]+list[i]]]];
  end for;
  n := #list;
  A := [];
  for i in [0..&*list-1] do
    j := i;
    AA := [];
    for l in [list[#list-i] : i in [0..#list-1]] do
      AA cat:= [j mod l];
      j := Integers()!((j-(j mod l))/l);
    end for;
    A cat:= [[AA[#AA-i] : i in [0..#AA-1]]];
  end for;
  return A;
end function;

```

With the function `Thins` we list all thin graphs that have n vertices.

```

Thins := function(n)
  PV := [<i,j> : i,j in [1..n]]; // pairs of vertices for one edge
  LG := []; // list of thin graphs
  for i in [0..2^(n^2)-1] do
    A := DTB(i); // %%
    E := [PV[i] : i in [1..#A] | A[i] eq 1];
    E := [<e[1],Index(E,e),e[2]> : e in E];
  end for;
end function;

```

```

    LG cat:= [<[i : i in [1..n]],E>];
end for;
return LG;
end function;

```

With the function `Thins2` we list all thin graphs that have n vertices and that do not have an edge that has the same vertex as source as target.

```

Thins2 := function(n)
  PV := [<i,j> : i,j in [1..n]]; // pairs of vertices for one edge
  LG := []; // list of thin graphs
  for i in [0..2^(n^2)-1] do
    A := DTB(i); // %%
    E := [PV[i] : i in [1..#A] | A[i] eq 1];
    E := [e : e in E | not e[1] eq e[2]];
    E := [<e[1],Index(E,e),e[2]> : e in E];
    LG cat:= [<[i : i in [1..n]],E>];
  end for;
  return RedSeq(LG); // %%
end function;

```

For a graph H and a list F of prescribed cardinalities, the function `EFU` gives the list of etale fibrations $f: G \rightarrow H$ satisfying (Uni) such that the cardinalities of the fibres of the vertices of H under V_f are listed in F .

```

EFU := function(H,F) // returns all etale fibrations f: G -> H that
  // satisfy (Uni)
  // F: sizes of fibres in list, e.g. F := [1,2,1]
  n := #H[1];
  VG := [i : i in [1..&+F]];
  W := [[i : i in [1..F[1]]]];
  for i in [2..#F] do
    W cat:= [[u : u in [W[#W][#W[#W]]+1..W[#W][#W[#W]]+F[i]]]];
  end for;
  Vf := [<i,Index(W,[a : a in W | i in a][1])> : i in [1..&+F]];
  EG_all := [[<i,j> : i in W[e[1]], j in W[e[3]]] : e in H[2]];
  Ef_all_sort := &cat([[<[e : e in EG_all[i] | e[1] eq v],H[2][i]> :
    i in [1..#EG_all]]: v in VG]);
  Ef_all_sort := [e : e in Ef_all_sort | not #e[1] eq 0];
  Ef_poss := [[<Ef_all_sort[i][1][d[i]+1],Ef_all_sort[i][2]> : i in [1..#d]] :
    d in DTO([#e[1] : e in Ef_all_sort])]; // %%
  Ef_poss_numbered := [[<Ef_poss[i][j][1][1],j,Ef_poss[i][j][1][2]>,
    Ef_poss[i][j][2]> : j in [1..#Ef_poss[i]]] : i in [1..#Ef_poss]];
  EF := [<<Vf,Ef_poss_numbered[i]>,<VG,[Ef_poss_numbered[i][j][1] : j in
    [1..#Ef_poss_numbered[i]]>,H> : i in [1..#Ef_poss_numbered]];
    // <f,G,H> all possibilities that f is an etale fibration
  return [f : f in EF | &and[SuffCond(f[1],f[2],f[3])[i] : i in [1,2]]];
end function;

```

For example, given

```
H := c2chain(3);
F := [1,2,1];
```

we get

```
> EFU(H,F);
[
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 2>, <3, 3, 2>> ]>, <[ 1, 2, 3, 4 ],
    [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>, <4, 6, 2> ]>,
    <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 2>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 3>, <3, 3, 2>> ]>, <[ 1, 2, 3, 4 ],
    [ <1, 1, 2>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>, <4, 6, 3> ]>,
    <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 3>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 2>, <3, 3, 2>> ]>, <[ 1, 2, 3, 4 ],
    [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>, <4, 6, 2> ]>,
    <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>,
  <<[ <1, 1>, <2, 2>, <3, 2>, <4, 3> ], [ <<1, 1, 3>, <1, 1, 2>>,
    <<2, 2, 4>, <2, 2, 3>>, <<2, 3, 1>, <2, 4, 1>>, <<3, 4, 4>, <2, 2, 3>>,
    <<3, 5, 1>, <2, 4, 1>>, <<4, 6, 3>, <3, 3, 2>> ]>, <[ 1, 2, 3, 4 ],
    [ <1, 1, 3>, <2, 2, 4>, <2, 3, 1>, <3, 4, 4>, <3, 5, 1>, <4, 6, 3> ]>,
    <[ 1, 2, 3 ], [ <1, 1, 2>, <2, 2, 3>, <3, 3, 2>, <2, 4, 1> ]>>
]
```

10.9 More useful functions

```
IsEqual := function(f,g) // returns true if graph morphisms
  // f, g: G -> H are equal
  // returns true if graphs f, g are equal
  return (SequenceToSet(f[1]) eq SequenceToSet(g[1]))
    and (SequenceToSet(f[2]) eq SequenceToSet(g[2]));
end function;
```

```

IsoRepresentatives := function(T,n); // returns representatives of isoclasses
TT := [T[1]];
for i in [1..n] do
  if not &or[IsIsomorphic(T[i],t)[2] : t in TT] then // %%
    TT cat:= [T[i]];
  end if;
end for;
return TT;
end function;

```

```

Is_Comm_Quad_Graphs := function(X,Y,X2,Y2,f,f2,g,h)
  // f: X -> Y, f2: X2 -> Y2, g: X -> X2, h: Y -> Y2
  if IsEqual(ComposeGraphMorphisms(f,h),ComposeGraphMorphisms(g,f2)) then // %%
    return true;
  end if;
  return false;
end function;

```

The function `Lift` returns a lift of a given commutative quadrangle if existent.

```

Lift := function(X,Y,X2,Y2,f,f2,g,h) // f: X -> Y, f2: X2 -> Y2,
  // g: X -> X2, h: Y -> Y2
  L := ListGraphMorphisms(X2,Y); // %%
  L_comm := [l : l in L | IsEqual(ComposeGraphMorphisms(g,l),f) and
    IsEqual(ComposeGraphMorphisms(l,h),f2)]; // %%
  if not #L_comm eq 0 then
    return L_comm[1];
  end if;
  return false;
end function;

```

```

remove_edges := function(G,list)
  return <G[1],[e : e in G[2] | not e in list]>;
end function;

```

```

remove_vertices := function(G,list)
  return <[n : n in G[1] | not n in list],[e : e in G[2] |
    not e[1] in list and not e[2] in list]>;
end function;

```

```

graph_op := function(G); // functor, reverses the direction of arrows in graph
  // exchanges source and target of edges in graph
  return <G[1],[<e[3],e[2],e[1]> : e in G[2]]>;
end function;

```

```
connection := function(G) // returns the list of connected components
list := [];
G_done := {};
while not #G_done eq #G[1] do
  S := {}; // S: set of vertices to achieve
  r := [g : g in G[1] | not g in G_done][1];
  Snew := {r};
  while not #Snew eq #S do
    S := Snew;
    Snew join:= {a[1] : a in G[2] | a[3] in S} join {a[3] : a in G[2] | a[1] in S};
  end while;
  G_done join:= Snew;
  Snew := Sort(SetToSequence(Snew));
  edges := Sort([a : a in G[2] | a[1] in Snew or a[3] in Snew]);
  list cat:= [<Snew,edges>];
end while;
return list;
end function;
```

Appendix A

Explanation for electronic appendix

On the memory stick attached there is a file called `electronic_appendix.txt`. It contains all functions in Magma code that are mentioned in this master thesis. It can be loaded into Magma with

```
load "electronic_appendix.txt";
```


References

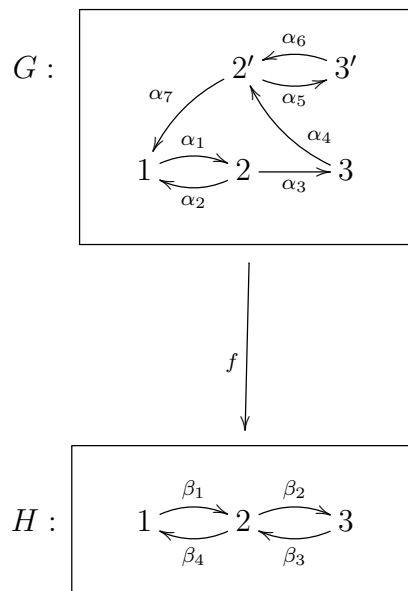
- [1] Nicolas Bourbaki, *Univers*, appendix to Exposé I of M. Artin, A. Grothendieck, J.-L. Verdier: *Théorie des Topos et Cohomologie Étale des Schémas* (SGA 4), Springer Lecture Notes 269, 1972.
- [2] Wieb Bosma, John Cannon and Catherine Playoust, *The Magma algebra system. I. The user language*, J. Symbolic Comput., 24, pp. 235–265, 1997.
- [3] Terrence Bisson and Aristide Tsemo, *A homotopical algebra of graphs related to zeta series*, Homology, Homotopy and Applications, vol. 11(1), pp. 171–184, 2009.
- [4] Daniel G. Quillen, *Homotopical algebra*, Springer Lecture Notes 43, 1967.
- [5] A. K. Bousfield and E. M. Friedlander, *Homotopy theory of Γ -spaces, spectra and bisimplicial sets*, Springer Lecture Notes 568, pp. 81–130, 2006.
- [6] Mathias Ritter, *Quasi-model-categories*, Master’s Thesis, Stuttgart, 2018.
- [7] Deborah A. Vicinsky, *The homotopy calculus of categories and graphs*, Thesis, University of Oregon, 2015.

Zusammenfassung

Graphen. Ein Graph G besteht aus einer Menge von Knoten V_G und einer Menge von Kanten E_G , zusammen mit einer Startabbildung $s_G : E_G \rightarrow V_G$ und einer Zielabbildung $t_G : E_G \rightarrow V_G$, welche einer Kante ihren jeweiligen Start- bzw. Zielknoten zuordnen.

Graphmorphismen. Ein Graphomorphismus $f : G \rightarrow H$ zwischen Graphen G und H besteht aus einer Abbildung $V_f : V_G \rightarrow V_H$ auf den Knoten und einer Abbildung $E_f : E_G \rightarrow E_H$ auf den Kanten derart, dass $E_f s_H = s_G V_f$ und $E_f t_H = t_G V_f$ ist.

Zum Beispiel bildet der folgende Graphomorphismus $f : G \rightarrow H$ die Knoten und Kanten vertikal ab.



Die Kategorie der Graphen bezeichnen wir mit Gph . Die Menge der Graphomorphismen von G nach H bezeichnen wir mit $(G, H)_{\text{Gph}}$. Zu einem Graphomorphismus $f : G \rightarrow H$ und einem Graphen K haben wir die Abbildung

$$(K, f)_{\text{Gph}} : (K, G)_{\text{Gph}} \rightarrow (K, H)_{\text{Gph}} : g \mapsto gf .$$

Eine Modellkategorienstruktur auf Gph. BISSON und TSEMO definieren Daten für eine Modellkategorienstruktur auf Gph wie folgt.

Ein Graphomorphismus $f : G \rightarrow H$ ist ein Quasiisomorphismus, wenn

$$(C_k, f)_{\text{Gph}} : (C_k, G)_{\text{Gph}} \rightarrow (C_k, H)_{\text{Gph}}$$

bijektiv ist für $k \geq 1$. Wir bezeichnen die Menge der Quasiisomorphismen mit $\text{Qis} \subseteq \text{Mor}(\text{Gph})$. Für einen Knoten $v \in V_G$ bezeichnen wir mit $G(v, *) := \{e \in E_G : e s_G = v\}$ die Menge der Kanten mit Start v . Wir haben die Abbildung

$$\begin{aligned} E_{f,v} := E_f \upharpoonright_{G(v,*)}^{H(v V_f, *)} : G(v, *) &\rightarrow H(v V_f, *) \\ e &\mapsto e E_f . \end{aligned}$$

Ein Graphmorphismus $f : G \rightarrow H$ heißt eine Faserung, wenn die Abbildung

$$E_{f,v} : G(v, *) \rightarrow H(v V_f, *)$$

surjektiv ist für $v \in V_G$. Wir bezeichnen die Menge der Faserungen mit $\text{Fib} \subseteq \text{Mor}(\text{Gph})$.

Ein Graphmorphismus $f : G \rightarrow H$ heißt eine etale Faserung, wenn die Abbildung

$$E_{f,v} : G(v, *) \rightarrow H(v V_f, *)$$

bijektiv ist für $v \in V_G$. Dies wird für ein hinreichendes Kriterium für Quasiisomorphismen benötigt.

Ein Graphmorphismus $f : G \rightarrow H$ heißt eine azyklische Kofaserung, wenn die Eigenschaften (AcCofib 1–5) erfüllt sind; cf. Definition 162. Anschaulich erhalten wir eine azyklische Kofaserung $f : G \rightarrow H$, wenn der Graph H durch Ankleben von Bäumen an den Graphen G entsteht. Wir bezeichnen die Menge der azyklischen Kofaserungen mit $\text{AcCofib} \subseteq \text{Mor}(\text{Gph})$.

Die Menge der azyklischen Faserungen bezeichnen wir mit $\text{AcFib} := \text{Qis} \cap \text{Fib} \subseteq \text{Mor}(\text{Gph})$. Wir schreiben $G \dashrightarrow H$ für eine azyklische Faserung.

Ein Graphmorphismus $f : G \rightarrow H$ heißt eine Kofaserung, wenn wir zu einem kommutativen Viereck

$$\begin{array}{ccc} G & \xrightarrow{a} & X \\ f \downarrow & & \downarrow g \\ H & \xrightarrow{b} & Y \end{array},$$

immer einen Lift $h : H \rightarrow X$ so finden, dass zwei kommutative Dreiecke entstehen. Die Menge der Kofaserungen bezeichnen wir mit $\text{Cofib} \subseteq \text{Mor}(\text{Gph})$.

Nun wird Gph zusammen mit Qis , Fib und Cofib zu einer geschlossenen Quillen-Modellkategorie; cf. [3, Cor. 4.8].

Zusätzlich gilt $\text{AcCofib} = \text{Cofib} \cap \text{Qis}$.

Beweis, dass Gph eine Modellkategorie ist. Zum Beweis folgen wir BISSON und TSEMO [3]. Beim Nachweis der Faktorisierung eines gegebenen Morphismus in eine Kofaserung, gefolgt von einer azyklischen Faserung verwenden wir eine iterierte Pushout-Konstruktion.

Ein hinreichendes Kriterium für Quasiisomorphismen. Sei $f : G \rightarrow H$ ein Graphmorphismus. Eine Kante $e \in E_H$ heißt einzielig, wenn das Urbild der Kante e unter E_f einen eindeutigen Zielknoten besitzt.

Wir betrachten folgende Eigenschaft.

(Uni) Für $n \geq 1$ und jeden Graphmorphismus $u : C_n \rightarrow H$ gibt es ein $i \in \mathbb{Z}/n\mathbb{Z}$ so, dass $e_i E_u \in E_H$ einzielig ist.

Nach Entfernung der einzielligen Kanten aus H darf es also keinen Zykel mehr darin geben, damit (Uni) erfüllt ist.

Wenn eine etale Faserung $f : G \rightarrow H$ die Eigenschaft (Uni) erfüllt, ist sie ein Quasiisomorphismus.

Z.B. ist der Graphmorphismus $f : G \rightarrow H$ von oben ein Quasiisomorphismus, wie damit überprüft werden kann.

Beispiele und Gegenbeispiele. Wir geben eine Reihe von Beispielen für Quasiisomorphismen, berechnet mit Magma [2] mithilfe unseres hinreichenden Kriteriums.

Wir zeigen, dass Quasiisomorphismen nicht stabil sind unter Pushouts entlang Kofaserungen.

Wir zeigen, dass Kofaserungen nicht stabil sind unter Pullbacks.

Wir geben ein Beispiel $f : G \rightarrow H$ so, dass $(C_1, f)_{\text{Gph}}$ und $(C_2, f)_{\text{Gph}}$ bijektiv sind und es keinen injektiven Graphmorphismus $C_k \rightarrow G$ oder $C_k \rightarrow H$ für $k \geq 3$ gibt, bei welchem jedoch f kein Quasiisomorphismus ist.

Versicherung

Hiermit versichere ich,

1. dass ich meine Arbeit selbstständig verfasst habe,
2. dass ich keine anderen als die angegeben Quellen benutzt habe und alle wörtlich oder sinngemäß aus anderen Werken übernommenen Aussagen als solche gekennzeichnet habe,
3. dass die eingereichte Arbeit weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen ist und
4. dass das elektronische Exemplar mit den anderen Exemplaren übereinstimmt.

Stuttgart, 12.01.2022

Jannik Hess