2008

Topos-like Properties in Two Categories of Graphs and Graph-like Features in an Abstract Category

Demitri Joel Plessas
The University of Montana

TOPOS-LIKE PROPERTIES IN TWO CATEGORIES OF GRAPHS AND GRAPH-LIKE FEATURES IN AN ABSTRACT CATEGORY

By

Demitri Joel Plessas

B.S. Montana Tech of the University of Montana, USA, 2006

Thesis

presented in partial fulfillment of the requirements
for the degree of

Master of Arts
in Mathematics

The University of Montana
Missoula, MT

Spring 2008

Approved by:

Dr. David A. Strobel, Dean
Graduate School

Dr. George McRae, Chair
Mathematical Sciences

Dr. Adam Nyman
Mathematical Sciences

Dr. Joel Henry
Computer Science

Plessas, Demitri J. Master of Arts, May 2008 Mathematics

Topos-like Properties in Two Categories of Graphs and Graph-like Features in an Abstract Category

Committee Chair: George McRae, Ph.D.

In the study of the Category of Graphs, the usual notion of a graph is that of a simple graph with at most one loop on any vertex, and the usual notion of a graph homomorphism is a mapping of graphs that sends vertices to vertices, edges to edges, and preserves incidence of the mapped vertices and edges. A more general view is to create a category of graphs that allows graphs to have multiple edges between two vertices and multiple loops at a vertex, coupled with a more general graph homomorphism that allows edges to be mapped to vertices as long as that map still preserves incidence. This more general category of graphs is named the Category of Conceptual Graphs.

We investigate topos and topos-like properties of two subcategories of the Category of Conceptual Graphs. The first subcategory is the Category of Simple Loopless Graphs with Strict Morphisms in which the graphs are simple and loopless and the incidence preserving morphisms are restricted to sending edges to edges, and the second subcategory is the Category of Simple Graphs with Strict Morphisms where at most one loop is allowed on a vertex. We also define graph objects that are their graph equivalents when viewed in any of the graph categories, and mimic their graph equivalents when they are in other categories. We conclude by investigating the possible reflective and coreflective aspects of our two subcategories of graphs.

Acknowledgements

I would like to thank my advisor, Dr. George McRae, for all of his help and guidance from the conception of this paper to its completion, the members of the Thesis Committee for their participation in my Thesis process, and the Department of Mathematical Sciences at the University of Montana for giving me this opportunity. I would also like to thank my wife, Ashley Plessas, for all of her love, help and support.

Notations

$E(G)$	the set of edges of a graph G	pg.2
$V(G)$	the set of vertices of a graph G	pg.3
$A \underline{\times} A$	unordered product of set A with itself	pg.3
∂_G	the incidence function of a graph G	pg.3
ι_G	the inclusion map of vertices into edges of a graph G	pg.3
$\underline{\Delta}$	the unordered diagonal map	pg.3
(u_v)	the unordered pair of u and v	pg.3
1_B	the local identity morphism on an object B	pg.7
$\mathcal{G}rph\mathcal{s}$	the Category of Conceptual Graphs	pg.8
$\mathcal{SiStGraphs}$	the Category of Simple Graphs with Strict Morphisms	pg.10
$\mathcal{SiLlStGraphs}$	the Category of Simple Loopless Graphs with Strict Morphisms	pg.10
K_2	the complete graph on 2 vertices	pg.17
K_1	the complete graph on 1 vertex	pg.23
$A \dot{\cup} B$	the disjoint union of sets A and B	pg.24
K_n^c	the empty edge graph on n vertices	pg.26
$\lvert - \rvert$	the underlying set functor	pg.26
$A \dashv B$	functor A is left adjoint to functor B	pg.26
$\sharp(X)$	the cardinality of a set X	pg.30
K_2^ℓ	the complete graph on 2 vertices with a loop at each vertex	pg.37
K_n^ℓ	the complete graph on n vertices with a loop at each vertex	pg.39
$\mathcal{A}b$	the Category of Abelian Groups and Group Homomorphisms	pg.45
tw	the twist automorphism of the edge object	pg.48
$\hookrightarrow$	an inclusion morphism	pg.2
$\rightarrow$	a morphism between objects	pg.7
$\rightsquigarrow$	a functor between categories	pg.16
$\twoheadrightarrow$	an epimorphism between objects	pg.16
$\rightarrowtail$	a monomorphism between objects	pg.18
$\mapsto$	function assignment	pg.70

Contents

Abstract **ii**

Acknowledgements **iii**

Notations **iv**

List of Figures **vii**

1 Introduction to the Categories of Graphs **1**

1.1 Introduction 1

1.2 Graphs and Graph Homomorphisms 2

1.3 Categories of Graphs 7

1.4 Categorial Constructions 11

1.5 Functors, Concrete Categories, and Special Morphisms 16

2 Categorial Comparisons of Simple Loopless Graphs with Strict Morphisms and Simple Graphs with Strict Morphisms **20**

2.1 Lack of Topos-like Properties in $\mathcal{SiLlStGraphs}$ 20

2.2 Other Categorial Constructions in $\mathcal{SiLlStGraphs}$ 28

2.3 Existence of Topos-like Properties in $\mathcal{SiStGraphs}$ 33

2.4 Other Categorial Constructions in $\mathcal{SiStGraphs}$ 39

2.5 $\mathcal{SiLlStGraphs}$ is a Topos Impoverished Category 43

3 Investigation of Graph-like Objects in an Abstract Category 45

3.1 Abstract Categorial Definitions of Graph-like Objects 45

3.2 Graph-like Objects in the Categories of Graphs 49

3.3 Graph-like Objects in Other Categories 57

4 Reflective and Coreflective Subcategories of Graph Categories 60

4.1 The Theory of Reflective and Coreflective Subcategories 60

4.2 Relationships in the Categories of Graphs 62

5 Conclusion 65

A Primer of Category Theory 66

Bibliography 78

List of Figures

1.1 Incidence Mappings for Vertices 3

1.2 An Example of a (non-simple) Conceptual Graph 4

1.3 An Example of a Simple Graph 4

1.4 An Example of a (non-simple) Loopless Graph 5

1.5 The Graph Morphism 6

1.6 The Associative Law 7

1.7 The Identity Law 8

1.8 The Product 11

1.9 The Coproduct 13

1.10 The Cone for a Diagram 14

1.11 The Limit of a Diagram 14

1.12 The Co-cone for a Diagram 15

1.13 The Colimit of a Diagram 15

2.1 Example of $K_2 \times K_2$ in *SiLlStGraphs* 22

2.2 Example of a Quotient Graph in $\mathcal{SiStGraphs}$ 34

2.3 Example of a $K_2^{K_2}$ in $\mathcal{SiStGraphs}$ 36

3.1 The Twist Automorphism, tw 48

3.2 The "Graph" of $(\mathbb{Z},+)$ 59

4.1 The Categories of Graphs 64

A.1 The Equalizer 67

A.2 The Coequalizer 67

A.3 The Product 68

A.4 The Coproduct 69

A.5 Exponentiation and Evaluation 69

A.6 The Pullback 70

A.7 The Pullback Square 70

A.8 The Free Object 75

A.9 The Cofree Object 75

A.10 A Projective Object 76

A.11 An Injective Object 76

Chapter 1

Introduction to the Categories of Graphs

1.1 Introduction

In [11], F. W. Lawvere defines the Category of Sets and Functions, $\mathcal{S}ets$, axiomatically, and in [15], D. Schlomiuk defines the Category of Topological Spaces and Continuous Maps, $\mathcal{T}op$, axiomatically. Berry Mitchell's [13] embedding theorem says that abstract abelian categories are quite concrete categories of modules. This has set a precedent for other mathematical fields to find an axiomatization of their categories. With recent advances in vertex coloring problems in graph theory, graph homomorphisms have been studied. This naturally leads to studying the categories of graphs and in 1977 P. Hell in [7] makes a case as to why graph theorists should do so.

To help in the long term goal of finding an axiomatic characterization of the Categories of Graphs we investigate two graph categories. We investigate the Category of Simple Graphs with Strict Morphisms, where the graphs have at most one edge between any two distinct vertices, and at most one loop at any vertex. The strict morphisms refer to graph homomorphisms that send vertices to vertices and edges to edges (strictly) while preserving the incidence of the mapped edges. A more general morphism allows edges to be mapped to vertices. Then we restrict ourselves to Simple Loopless Graphs with Strict Morphisms, where, in addition to being simple, the graphs cannot have loops.

There is a much more general category of graphs in which our chosen two categories live. This is the Category of Conceptual Graphs, where the morphisms allow edges to be sent to vertices, as long as incidence is still preserved, and the objects are graphs with multiple edges between any two vertices, and multiple loops at a vertex. In the view of this category, we are able to give, for the first time, an abstract categorial definition to graph-like objects as well as an abstract categorial definition of a strict morphism. This allows an investigation of graph-like objects in an abstract category. We also view our chosen categories as subcategories of the Category of Conceptual Graphs and investigate, for the first time, their categorial reflective and co-reflective properties.

We follow the notation of [1] for topics related to graphs and graph results. We follow the notation of [12] for topics related to categories and categorial results, with the exception that we use capital letters to stand for objects, and lower case letters to stand for morphisms inside our categories.

1.2 Graphs and Graph Homomorphisms

In our graphs, we want to start out with as great a generality as possible and add restrictions later. This means we want to allow graphs to have multiple edges between any two vertices and multiple loops at any vertex. We will define our graphs in the style of Bondy and Murty [1].

Definition 1.2.1. *A* conceptual graph G *consists of*

$G = \langle E(G), V(G); \partial_G : E(G) \to V(G)\underline{\times}V(G), \iota_G : V(G) \hookrightarrow E(G)\rangle$ *where* $E(G)$ *is the set* *of edges of G, $V(G)$ is the set of vertices of G, $V(G)\underline{\times}V(G)$ is the set of unordered pairs of vertices of G, ∂_G is the incidence map from the set of edges to the unordered pairs of vertices, ι_G is the inclusion map of the vertex set into the edge set, and for $\underline{\Delta} : V(G) \to V(G)\underline{\times}V(G)$ the unordered diagonal map, $\partial_G \circ \iota_G = \underline{\Delta}$.*

$$\begin{tikzcd} V(G) \arrow[r, hook, "\iota_G"] \arrow[dr, "\underline{\Delta}"'] & E(G) \arrow[d, "\partial_G"] \arrow[dl, phantom, "="] \\ & V(G)\underline{\times}V(G) \end{tikzcd}$$

Figure 1.1: Incidence Mappings for Vertices

Henceforth, we will frequently abbreviate conceptual graph to graph. Furthermore, in our study here, we have no need to restrict our edge sets and vertex sets of our graphs to be finite sets.

We note the following. First, we naturally use the topologist's "boundary" symbol for incidence. Second, an unordered pair in $V(G)\underline{\times}V(G)$ is denoted u_v or (u_v), for vertices $u, v \in V(G)$. Thus the natural unordered diagonal map $\underline{\Delta} : V(G) \to V(G)\underline{\times}V(G)$ is given by $\underline{\Delta}(v) = v\_v$ or (v_v). Finally, we have chosen to consider our vertex set to be a subset of the edge set (i.e. we consider the vertices to be "trivial edges"). Thus as an abstract data structure our graphs are a pair of sets: a set (of edges) and a distinguished subset (called vertices). This is done to make the description of morphisms more natural, i.e. functions between the over sets that takes the distinguished subset to the other distinguished subset. This is what topologists do in the Category of Topological Pairs of Spaces.

Often in graph theory the set of graphs is restricted to allow only one edge between any two vertices (see [8]), and at most one edge between a vertex and itself (a loop). We call these graphs *simple graphs* and define them in terms of conceptual graphs.

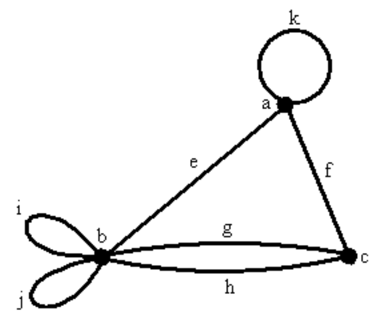


Figure 1.2: An Example of a (non-simple) Conceptual Graph

Definition 1.2.2. *A* simple *graph G is a conceptual graph such that for all $u, v \in V(G)$ with $u \neq v$, there is at most one $e \in E(G)$ such that $\partial_G(e) = (u\_v)$, and for all $w \in V(G)$ there is at most one $f \in E(G) \backslash image(\iota_G)$ such that $\partial_G(f) = (w\_w)$ (where (u_v) is the unordered pair of vertices u and v).*

Thus, a graph is simple if and only if the incidence map is injective (i.e. one-to-one).

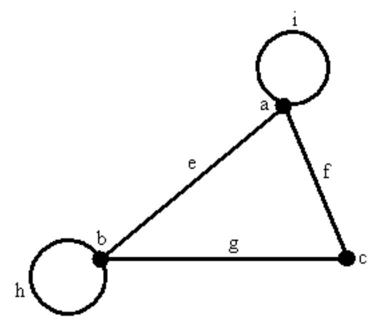


Figure 1.3: An Example of a Simple Graph

Another common restriction is to not allow loops at all (see [5]). This restriction is often required when discussing vertex coloring. We call these graphs *loopless graphs*.

Definition 1.2.3. *A* loopless *graph G is a conceptual graph such that for all $u \in V(G)$ there is no $e \in E(G) \backslash image(\iota_G)$ such that $\partial_G(e) = (u\_u)$.*

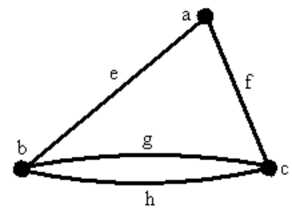


Figure 1.4: An Example of a (non-simple) Loopless Graph

In [1] a graph does not have the inclusion map, ι, but such a map will be critical when defining a graph homomorphism. In this way, we can think of the vertex "part" of the graph as a special type of edge "part" of the graph. When we refer to an edge it will be our convention to refer to an element of $E(G)\backslash image(\iota_G)$, and we do allow $G = \emptyset$, the empty graph, to be considered a graph. However, since ∂_G is required to be a function, if $V(G) = \emptyset$ then $E(G) = \emptyset$.

Almost every textbook on Graph Theory defines a graph isomorphism early in their discussion of Graph Theory (see [1] and [2]) but few define a graph homomorphism. Vertex colorings of a graph have led into research of graph homomorphisms (see [14]) and in 2004 Hell and Nešetřil published the first graph homomorphism textbook [8]. The following definition is a modified form of the definition presented in [8] to apply to conceptual graphs.

Definition 1.2.4. *Let G and H be conceptual graphs. A* strict graph homomorphism *(or* strict morphism*) $f : G \to H$ is a function $f_E : E(G) \to E(H)$ such that $f_V : V(G) \to V(H)$, where f_V is the restriction of f_E to $V(G)$, i.e. $f_V = f_E|_{V(G)}$; incidence is preserved: $\partial_H(f_E(e)) = (f_V(x)\_f_V(y))$ whenever $\partial_G(e) = (x\_y)$, for some $x, y \in V(G)$; and, in addition the strict edge condition is satisfied: for all $e \in E(G)\backslash image(\iota_G)$, $f_E(e) \in E(H)\backslash image(\iota_H)$.*

The condition, $\partial_H(f_E(e)) = (f_V(x)\_f_V(y))$ whenever $\partial_G(e) = (x\_y)$, assures that the incidence of the edges in G is preserved in H under f. Note that the above definition also requires that vertices be mapped to vertices and edges be mapped (strictly) to edges. However, sometimes it may be beneficial to allow edges to be mapped to vertices. Such a morphism would allow a graph to naturally map to the contraction or quotient graph obtained by the contraction of an edge, but this could not be a strict morphism. As such, we call the above definition of a graph homomorphism a *strict graph morphism* and now define a more general graph (homo)morphism.

Definition 1.2.5. *$f : G \to H$ is a* graph (homo)morphism *of conceptual graphs from G to H if f is a function $f_E : E(G) \to E(H)$ and $f_V = f_E|_{V(G)} : V(G) \to V(H)$ that preserves incidence, i.e. $\partial_H(f_E(e)) = (f_V(x)\_f_V(y))$ whenever $\partial_G(e) = (x\_y)$, for all $e \in E(G)$ and some $x, y \in V(G)$.*

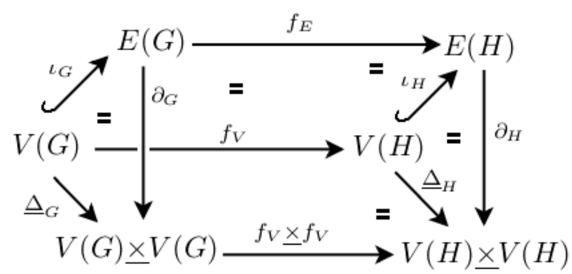


Figure 1.5: The Graph Morphism

This definition allows a graph homomorphism to map an edge to a vertex as long as the incidence of the edges are preserved. As an edge, $e \in E(G)$, can be mapped to the edge set of the codomain graph, H, so that it is the image of a vertex, i.e. $f(e) = \iota_H(v)$ for some $v \in V(H)$.

Now that we have defined our graphs and graph homomorphisms, we are ready to discuss categories of graphs.

1.3 Categories of Graphs

We begin this section by defining a category axiomatically (as in [4]).

Definition 1.3.1. *A* category, $\mathcal{C}$, *comprises*

(1) a class of objects (e.g. dots • or capital letters A, B, *and* C*);*

(2) a class of morphisms (e.g. arrows $\rightarrow$ *or lower case letters* f, g, *and* h*);*

(3) operations assigning each morphism, f, *an object Dom(*f*) (the "domain" of* f*) and an object Cod(*f*) (the "codomain" of* f*). If A=Dom(*f*) and B=Cod(*f*) we display this as* $f : A \rightarrow B$ *or* $A \xrightarrow{f} B$*;*

(4) an operation assigning each pair $\langle g, f\rangle$ *of morphisms with Dom(g)=Cod(*f*), a morphism* $g \circ f$, *the composite of* f *and* g, *with Dom(*$g \circ f$*)=Dom(*f*) and Cod(*$g \circ f$*)=Cod(g), i.e.* $g \circ f$ *:Dom(*f*)* $\rightarrow$*Cod(g), such that the associative law holds, i.e. given* $A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$ *then* $(h \circ g) \circ f = h \circ (g \circ f)$*;*

(5) an assignment to each object B *a morphism* $1_B : B \rightarrow B$, *called the local identity of* B, *such that the identity law holds, i.e. for all morphisms* $f : A \rightarrow B$ *and* $g : B \rightarrow C$, $1_B \circ f = f$ *and* $g \circ 1_B = g$.

The associative law asserts that the following diagram commutes:

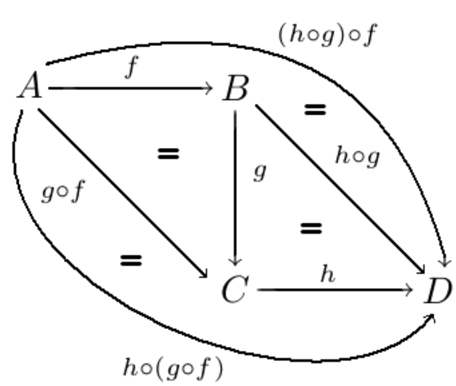


Figure 1.6: The Associative Law

While the identity law asserts that the following diagram commutes:

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[dr, "1_B \circ f"'] & B \arrow[d, "1_B"] \arrow[dr, "g \circ 1_B"] & \\
& B \arrow[r, "g"] & C
\end{tikzcd}$$

Figure 1.7: The Identity Law

We now define an isomorphism in a category. We think of isomorphisms as morphisms that preserve the complete structure of an object.

Definition 1.3.2. *A morphism* $f : A \to B$ *is an* isomorphism *if there exists a morphism* $g : B \to A$ *such that* $f \circ g = 1_B$ *and* $g \circ f = 1_A$

We also think of isomorphisms as (two-sided) invertible morphisms. Now that a conceptual graph, graph morphism, and a category are defined, we can define the categories of graphs.

Definition 1.3.3. *The* Category of Conceptual Graphs*, Grphs, is a category where the objects are conceptual graphs and the morphisms are graph morphisms.*

We now must show the axioms of a category are satisfied by this definition.

Proposition 1.3.4. *Grphs is a category.*

Proof. Since our objects and morphisms are defined, and in our definition of a graph homomorphism, a domain and codomain are defined, *Grphs* satisfies axioms (1),(2), and (3). Now we naturally define the compositions of graph homomorphisms and show that these compositions are graph homomorphisms. Let A, B, and C be objects in *Grphs*, and let f and g be morphisms in *Grphs* such that $A \xrightarrow{f} B \xrightarrow{g} C$. We then define $g \circ f$ to be a pair of compositions of set functions $g_V \circ f_V : V(A) \to V(C)$ and $g_E \circ f_E : E(A) \to E(C)$.

Now let $e \in E(A)$, then there are vertices $u, v \in V(A)$ such that $\partial_A(e) = (u\_v)$. Now consider $\partial_C(g_E \circ f_E(e))$. Since f is a graph homomorphism, $\partial_B(f_E(e)) = (f_V(u)\_f_V(v))$. Since g is a graph homomorphism, $\partial_C(g_E \circ f_E(e)) = \partial_C(g_E(f_E(e))) = (g_V(f_V(u))\_g_V(f_V(v))) = ((g_V \circ f_V(u))\_(g_V \circ f_V(v)))$. Hence incidence is preserved, and $g \circ f$ is a graph homomorphism.

Now let A, B, C, and D be objects in $\mathcal{Grphs}$, and let f, g, and h be morphisms in $\mathcal{Grphs}$ such that $A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$. Consider $(h \circ g) \circ f$. Since composition of graph homomorphisms are graph homomorphisms, $(h \circ g) \circ f$ is a pair of set functions $(h_V \circ g_V) \circ f_V : V(A) \to V(D)$ and $(h_E \circ g_E) \circ f_E : E(A) \to E(D)$ that preserve incidence. Since set functions are associative, $(h_V \circ g_V) \circ f_V = h_V \circ (g_V \circ f_V)$ and $(h_E \circ g_E) \circ f_E = h_E \circ (g_E \circ f_E)$ and hence $(h \circ g) \circ f = h \circ (g \circ f)$. Therefore the associative law is satisfied and, as such, so is axiom (4).

We now show there are local identities and that the identity morphism satisfies the identity law. Let B be an object in $\mathcal{Grphs}$. Define $1_B : B \to B$ as the pair of set functions $1_{V(B)} : V(B) \to V(B)$ and $1_{E(B)} : E(B) \to E(B)$ where $1_{V(B)}$ is the identity function on the set $V(B)$ and $1_{E(B)}$ is the identity function on $E(B)$. Let $e \in E(B)$ such that $\partial_B(e) = (u\_v)$ for some $u, v \in V(B)$, then $\partial_B(1_{E(B)}(e)) = \partial_B(e) = (u\_v) = (1_{V(B)}(u)\_1_{V(B)}(v))$. Thus 1_B preserves incidence and is a graph homomorphism.

Let A be an object in $\mathcal{Grphs}$ with a morphism $f : A \to B$, and C be an object in $\mathcal{Grphs}$ with a morphism $g : B \to C$. Consider $1_B \circ f$. Since the composition $1_B \circ f$ is a pair of set functions $1_{V(B)} \circ f_V$ and $1_{E(B)} \circ f_E$, and since $1_{V(B)} \circ f_V = f_V$ and $1_{E(B)} \circ f_E = f_E$, $1_B \circ f = f$. Now consider $g \circ 1_B$. $g \circ 1_B$ is a pair of set functions $g_V \circ 1_{V(B)}$ and $g_E \circ 1_{V(B)}$. Since $g_V \circ 1_{V(B)} = g_V$ and $g_E \circ 1_{E(B)} = g_E$, $g \circ 1_B = g$. Hence 1_B satisfies the identity law and therefore axiom (5).

Thus $\mathcal{Grphs}$ is a category. □

The next graph category is the only undirected graph category in the literature. It is used in [16], [3], and unofficially in [8], although one of the authors, Pavol Hell, officially uses this category in [7].

Definition 1.3.5. *The* Category of Simple Graphs with Strict Morphisms*, $\mathcal{SiStGraphs}$, is a category where the objects are simple graphs and the morphisms are strict graph homomorphisms.*

Proposition 1.3.6. *$\mathcal{SiStGraphs}$ is a category.*

Proof. The proof follows similarly to the proof of Proposition 1.3.3., given that we prove that the composition of strict graph homomorphisms are strict graph homomorphisms. So let A, B, and C be graphs (not necessarily simple) with strict graph homomorphisms f and g where $A \xrightarrow{f} B \xrightarrow{g} C$. Since f and g are graph homomorphisms, by the proof of Proposition 1.3.3., $g \circ f$ is a graph homomorphism. Let $e \in E(A) \backslash image(\iota_A)$, and consider $g \circ f(e)$. Since f is a strict graph homomorphism, $f(e) \in E(B) \backslash image(\iota_B)$. Then since g is a strict graph homomorphism, $g \circ f(e) = g(f(e)) \in E(C) \backslash image(\iota_C)$. Hence the composition of strict graph homomorphisms is a strict graph homomorphism. □

Many Graph Theory textbooks, especially those aimed at undergraduates, restrict their graphs to be simple graphs without loops (see [2] and [5]). When the need for graphs with multiple edges or loops arise, they are often called multigraphs [2] or pseudographs [5]. This restriction on the type of graphs allowed in their discussion leads naturally to a graph category.

Definition 1.3.7. *The* Category of Simple Loopless Graphs with Strict Morphisms*, $\mathcal{SiLlStGraphs}$, is the category where the objects are simple graphs without loops, and the morphisms are strict graph homomorphisms.*

Proposition 1.3.8. *$\mathcal{SiLlStGraphs}$ is a category.*

Proof. Since $\mathcal{SiLlStGraphs}$ is contained in $\mathcal{SiStGraphs}$ as a restriction of the objects, the proof follows similarly to Proposition 1.3.5. □

The usual justification of the restriction is that it simplifies the theory. Many times this is the case and theorems related to both matchings and colorings of a graph require this restriction as a hypothesis, but we find taking this restriction as the definition of a graph leads to a impoverished category.

1.4 Categorial Constructions

We highlight some categorial constructions, providing definitions and basic results. This will not be a comprehensive list, but a comprehensive list of definitions for the categorial constructions used in this paper can be found in Appendix A. Since much of the literature related to the categories of graphs is focused on products (see [16] and [6]), we start by defining products.

Definition 1.4.1. Products *exist in a category* $\mathcal{C}$*, if for all objects* A *and* B *in* $\mathcal{C}$*, there exists an object* $A \times B$ *with morphisms* $\pi_A : A \times B \to A$ *and* $\pi_B : A \times B \to B$ *in* $\mathcal{C}$ *such that for all objects* X *with morphisms* $f_A : X \to A$ *and* $f_B : X \to B$*, there exists a unique morphism* $\overline{f} : X \to A \times B$ *such that* $f_A = \pi_A \circ \overline{f}$ *and* $f_B = \pi_B \circ \overline{f}$*.*

This definition is expressed as the following commuting diagram.

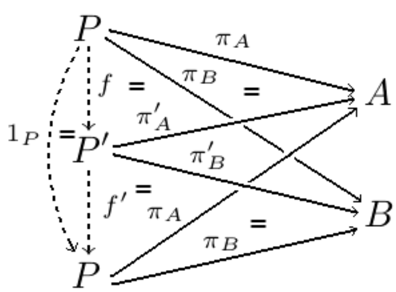
$$
\begin{tikzcd}
X \arrow[d, dashed, "\exists ! \overline{f}"'] \arrow[rr, bend left, "f_A"] \arrow[rd, bend left, "f_B"] & & \\
A \times B \arrow[rr, "\pi_A"] \arrow[rd, "\pi_B"'] & & A \\
& B &
\end{tikzcd}
$$

Figure 1.8: The Product

In the Category of Sets and Functions, $\mathcal{S}ets$, the product is the Cartesian product (for a proof see [12]). The form of the definition of the product, $\forall \ldots \exists \ldots$ such that $\forall \ldots \exists! \ldots$, is a general form for categorial constructions known as a universal mapping property. Universal mapping properties define an object universally such that every other object with the same properties “factors through” the universal object with a unique morphism. Such a definition gives rise to a generic proof that the “universally constructed” object is unique up to isomorphism in the category.

Proposition 1.4.2. *Given objects A and B in a category with products, then the product $A \times B$ is unique up to isomorphism.*

Proof. Suppose that for two objects A and B in the category, there are two products, P with morphisms $\pi_A : P \to A$ and $\pi_B : P \to B$, and P' with morphisms $\pi'_A : P' \to A$ and $\pi'_B : P' \to B$. Then by the definition of products there exists two unique morphisms $f : P \to P'$ and $f' : P' \to P$ such that the following diagram commutes.



Hence there is a unique morphism $f' \circ f : P \to P$ such that $\pi_A \circ (f' \circ f) = \pi_A$ and $\pi_B \circ (f' \circ f) = \pi_B$. However, the identity morphism 1_P satisfies those two equations as well. Since the morphism is unique, $f' \circ f = 1_P$.

A similar argument give us $f \circ f' = 1_{P'}$. Hence f is an isomorphism and $P \cong P'$. $\square$

We now consider the dual construction of the product. A dual construction in a category is obtained by “reversing the arrows” of a categorial construction. That is, the morphisms will be pointing the opposite way. Though in practice, when a diagram of the dual is drawn, the arrows point the same direction (i.e. left to right and up to down) and the objects are interchanged. The dual of the product is called the coproduct.

Definition 1.4.3. Coproducts *exist in a category $\mathcal{C}$, if for all objects A and B in $\mathcal{C}$, there exists an object $A + B$ with morphisms $i_A : A \to A + B$ and $i_B : B \to A + B$ such that for all objects X with morphisms $g_A : A \to X$ and $g_B : B \to X$, there exists a unique morphism $\overline{g} : A + B \to X$.*

We now give a similar diagram that represents the coproduct.

$$
\begin{tikzcd}
A \arrow[drr, "i_A"] \arrow[ddrr, "g_A"'] & & \\
B \arrow[rr, "i_B"] \arrow[drr, "g_B"'] & & A + B \arrow[d, dashed, "\exists! \overline{g}"] \\
& & X
\end{tikzcd}
$$

Figure 1.9: The Coproduct

In $\mathcal{S}ets$ the disjoint union of two sets is the coproduct (for a proof see [12]). Note that the coproduct is also defined in terms of a universal mapping property, thus a similar proof to Proposition 1.4.2. yields the following.

Proposition 1.4.4. *Given objects A and B in a category with coproducts, then the coproduct $A + B$ is unique up to isomorphism.*

A more general categorial construction is that of limits and colimits. We define these constructions in the way of [4]. We will first define a diagram in a category.

Definition 1.4.5. *In a category $\mathcal{C}$ a* diagram *$\mathcal{D}$ of $\mathcal{C}$ is a collection of objects $D_i, D_j, \ldots$ in $\mathcal{C}$ along with a collection of morphisms $f : D_i \to D_j$ between certain objects in the diagram.*

An example of a diagram of two objects with two morphisms is $A \rightrightarrows B$. Now that we have the definition of a diagram, we can define a cone for a diagram.

Definition 1.4.6. *A* cone *for a diagram $\mathcal{D}$ in a category $\mathcal{C}$ consists of an object C in $\mathcal{C}$ together with a morphism $f_i : C \to D_i$ for each object in $\mathcal{D}$ such that for every morphism $g : D_i \to D_j$ in $\mathcal{D}$, $g \circ f_i = f_j$.*

The definition of a cone asserts that the following diagram commutes for every morphism g in a diagram $\mathcal{D}$.

$$\begin{tikzcd}
C \arrow[d, "f_i"'] \arrow[dr, "f_j"] \arrow[dr, phantom, shift right=3ex, "="] & \\
D_i \arrow[r, "g"] & D_j
\end{tikzcd}$$

Figure 1.10: The Cone for a Diagram

For convenience, we often represent a cone for $\mathcal{D}$ by $\{f_i : C \to D_i\}$. We can now define a limit in terms of a cone for a diagram.

Definition 1.4.7. *A* limit *of a diagram $\mathcal{D}$ is a ("universal") cone for $\mathcal{D}$, $\{f_i : \underleftarrow{L} \to D_i\}$, such that for any other cone for $\mathcal{D}$, $\{f_i' : C' \to D_i\}$, there exists a unique morphism $\overline{f} : C' \to \underleftarrow{L}$ such that $f_i' = f_i \circ \overline{f}$ for all objects D_i in $\mathcal{D}$.*

The definition states that the following diagram commutes for all objects D_i of $\mathcal{D}$.

$$\begin{tikzcd}
C' \arrow[d, dashed, "\overline{f}"'] \arrow[dr, "f_i'"] \arrow[dr, phantom, shift right=3ex, "="] & \\
\underleftarrow{L} \arrow[r, "f_i"] & D_i
\end{tikzcd}$$

Figure 1.11: The Limit of a Diagram

The product is a limit of the diagram just consisting of two objects A and B with no morphisms between them. Many categorial constructions can be defined in terms of limits. To define the dual of the limit, the colimit, we first must define the co-cone.

Definition 1.4.8. *The* co-cone *for a diagram $\mathcal{D}$, $\{f_i : D_i \to C\}$, is an object C with morphisms $f_i : D_i \to C$ for every object D_i in $\mathcal{D}$, such that for every morphism $g : D_i \to D_j$ in $\mathcal{D}$, $f_i = f_j \circ g$.*

Similar to the definition of a cone, the definition of a co-cone states that the following digram commutes for all objects D_i of $\mathcal{D}$.

$$\begin{tikzcd}
D_i \arrow[r, "f"] \arrow[drr, "f_i"'] & D_j \arrow[dr, "f_j"] \arrow[d, phantom, "="] & \\
& {} & C
\end{tikzcd}$$

Figure 1.12: The Co-cone for a Diagram

The colimit is now defined in terms of the co-cone for a diagram.

Definition 1.4.9. *A* colimit *of a diagram $\mathcal{D}$ is a ("universal") co-cone for $\mathcal{D}$, $\{f_i : D_i \to \underrightarrow{L}\}$, such that for any other co-cone for $\mathcal{D}$, $\{f'_i : D_i \to C'\}$, these exists a unique morphism $\overline{f} : \underrightarrow{L} \to C'$ such that $f'_i = \overline{f} \circ f_i$ for all objects D_i in $\mathcal{D}$.*

The definition asserts that the following diagram commutes for all objects D_i of $\mathcal{D}$.

$$\begin{tikzcd}
D_i \arrow[r, "f_i"] \arrow[dr, "f'_i"'] \arrow[dr, phantom, "=", shift left=3] & \underrightarrow{L} \arrow[d, dashed, "\overline{f}"] \\
& C'
\end{tikzcd}$$

Figure 1.13: The Colimit of a Diagram

The coproduct is a colimit of the diagram just consisting of two objects A and B with no morphisms between them. Many co-constructions can be phrased in terms of colimits. We now turn our attention to special morphisms and concrete categories. We define a concrete category via a faithful functor as in [12].

1.5 Functors, Concrete Categories, and Special Morphisms

Definition 1.5.1. *A* functor $F : \mathcal{C} \rightsquigarrow \mathcal{B}$ *is a morphism of categories such that F assigns each object C of $\mathcal{C}$ an object $F(C)$ in $\mathcal{B}$ and for each morphism $g : C \to C'$ a morphism $F(g) : F(C) \to F(C')$ such that $F(1_C) = 1_{F(C)}$ and for each composition $g \circ h$ in $\mathcal{C}$, $F(g \circ h) = F(g) \circ F(h)$.*

Functors constitute a large part of the study of Category Theory. A proposition that follows straight from the definition of functors is that functors preserve isomorphisms. In Chapter 5, we will study functors between the categories of graphs.

Definition 1.5.2. *A functor $T : \mathcal{C} \rightsquigarrow \mathcal{B}$ is* faithful *if for every pair of objects C and C' in $\mathcal{C}$ and for every pair of parallel morphisms $f_1, f_2 : C \to C'$ of $\mathcal{C}$, $T(f_1) = T(f_2)$ implies $f_1 = f_2$.*

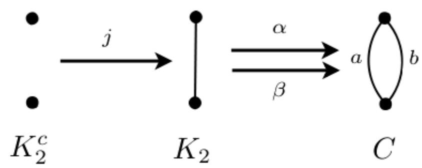
Definition 1.5.3. *A* concrete *category is a pair $\langle \mathcal{C}, U \rangle$ where $\mathcal{C}$ is a category and U is a faithful functor $U : \mathcal{C} \rightsquigarrow \mathcal{S}ets$.*

Since U is a faithful functor, we can identify each morphism f in $\mathcal{C}$ with a function $U(f)$. So we can think of objects C of a concrete category as having an underlying set $U(C)$ with added structure. U is often called the *underlying set functor*. Often $| - |$ is used for the underlying set functor and we will later use this notation, especially for the underlying vertex set functor.

Often times whenever functions or homomorphisms are discussed in any theory of mathematics, discourse about surjective functions and injective functions are not far behind. Since the definition of both surjective functions and injective functions (or homomorphisms) rely upon an underlying set, they can only be discussed in terms of concrete categories. A more general property surjective functions satisfy gives rise to the notion: epimorphism. We define an epimorphism following the style of [12].

Definition 1.5.4. *A morphism $h : A \twoheadrightarrow B$ in a category $\mathcal{C}$ is an* epimorphism *if for any two morphisms $g_1, g_2 : B \to C$ the equality $g_1 \circ h = g_2 \circ h$ implies $g_1 = g_2$.*

Epimorphisms are right cancellable. In *Sets* the epimorphisms are precisely the surjective functions. As we will prove, surjections are always epimorphisms, but the converse need not necessarily be true. For example, $j : K_2^c \hookrightarrow K_2$, the inclusion of the two vertices into the complete graph on two vertices, is injective and a monomorphism in all the categories of graphs, but j is not an epimorphism. Consider the following:



where α takes K_2 to the left (a) and β takes K_2 to the right (b), so $\alpha \neq \beta$ even though $\alpha \circ j = \beta \circ j$. However, j_V is surjective. Also, in the category of *SiLlStGraphs* this j is an epimorphism!

Proposition 1.5.5. *In concrete categories, surjective morphisms are epimorphisms.*

Proof. Let A and B be objects in our concrete category with a surjective morphism $f : A \to B$. Let $g_1, g_2 : B \to C$ be morphisms in our concrete category such that $g_1 \circ f = g_2 \circ f$, for some object C. Consider $U(g_1 \circ f)$ where U is the underlying set functor associated with our concrete category. Since U is a functor and $g_1 \circ f = g_2 \circ f$, $U(g_1 \circ f) = U(g_2 \circ f)$ and $U(g_1) \circ U(f) = U(g_1 \circ f) = U(g_2 \circ f) = U(g_2) \circ U(f)$.

Let $x \in U(B)$. Since $U(f)$ is an surjection, there is a $y \in U(A)$ such that $U(f)(a) = x$. Consider $U(g_1)(x)$. $U(g_1)(x) = U(g_1)(U(f)(a)) = U(g_2)(U(f)(a)) = U(g_2)(x)$. Hence $U(g_1) = U(g_2)$, and since U is a faithful functor, $g_1 = g_2$. Hence f is an epimorphism. □

As in the example above (for *SiLlStGraphs*), we give a new result that the vertex function of an epimorphism is surjective.

Proposition 1.5.6. *(i) In $\mathcal{Grphs}$, if $f : A \to B$ is an epimorphism, then the associated vertex function $f_V : V(A) \to V(B)$ is surjective.*
(ii) In $\mathcal{SiLlStGraphs}$ and $\mathcal{SiStGraphs}$, $f : A \to B$ is an epimorphism if and only if the vertex function $f_V : V(A) \to V(B)$ is surjective.

Proof. Part (i): Suppose f_V is not surjective. Then there exists $v \in V(B)\backslash image(f_V)$. Construct the graph C by appending a vertex v' to B such that v' is adjacent to every vertex v is adjacent to. By construction B is a subgraph of C.

Since $v \in V(B)\backslash image(f_V)$, no edge incident to v is in the image of f_E. Now consider $i : B \to C$ the inclusion morphism and $g : B \to C$ defined by $g(u) = i(u)$ for all $u \in V(B)\backslash\{v\}$, $g(v) = v'$, $g(e) = i(e)$ for all edges e not incident to v, and for edge f incident to v, set $g(f)$ to be the corresponding edge incident to v'. Then $i \circ f = g \circ f$ but $i \neq g$, a contradiction to f being an epimorphism. Hence epimorphisms in $\mathcal{Grphs}$ have surjective vertex set functions.
Part (ii): ($\Rightarrow$) The same proof in part (i) applies.
($\Leftarrow$) Suppose $f : A \to B$ is a morphism and f_V is surjective. We will show f if an epimorphism, i.e. for morphisms $h, k : B \to C$ such that $h \circ f = k \circ f$, we will show $h = k$. Since f_V is surjective and $h_V \circ f_V = k_V \circ f_V$, $h_V = k_V$. So if $h \neq k$ there exists a (nontrivial) edge $e \in B$ such that $h(e) \neq k(e)$, even though $h_V = k_V$. There are three possibilities for $h(e)$ and $k(e)$, either as different vertices, loops, or edges. But in both $\mathcal{SiLlStGraphs}$ and $\mathcal{SiStGraphs}$ this would contradict C being simple, or h and k being strict morphisms. $\square$

Just as there are epimorphisms which relate to surjections, there are monomorphisms which relate to injections. We will define a monomorphism following the style of [12].

Definition 1.5.7. *A morphism $h : B \rightarrowtail C$ in a category $\mathcal{C}$ is a* monomorphism *if for any two morphisms $f_1, f_2 : A \to B$ the equality $h \circ f_1 = h \circ f_2$ implies $f_1 = f_2$.*

Monomorphisms are left cancellable. In $\mathcal{Sets}$ the monomorphisms are precisely the injective functions. Similar to epimorphisms, injections are always monomorphisms, but the converse need not necessarily be true.

Proposition 1.5.8. *In concrete categories, injective morphisms are monomorphisms.*

Proof. Let B and C be objects in our concrete category with an injective morphism $m : B \to C$. Let $g_1, g_2 : A \to B$ be morphisms in our concrete category such that $m \circ g_1 = m \circ g_2$, for some object A. Consider $U(m \circ g_1)$ where U is the underlying set functor associated with our concrete category. Since U is a functor and $m \circ g_1 = m \circ g_2$, $U(m \circ g_1) = U(m \circ g_2)$ and $U(m) \circ U(g_1) = U(m \circ g_1) = U(m \circ g_2) = U(m) \circ U(g_2)$.

Let $x \in A$. Then $U(m) \circ U(g_1)(x) = U(m) \circ U(g_2)(x)$. Since $U(m)$ is an injection $U(g_1)(x) = U(g_2)(x)$. Hence since U is faithful $g_1 = g_2$, and m is a monomorphism. $\square$

We conclude this chapter by mentioning that while isomorphisms are epimorphisms and monomorphisms, the converse in not necessarily true in an abstract category. We shall see more explicit examples later (e.g. section 2.2).

Chapter 2

Categorial Comparisons of Simple Loopless Graphs with Strict Morphisms and Simple Graphs with Strict Morphisms

2.1 Lack of Topos-like Properties in $\mathcal{SiLlStGraphs}$

This chapter provides a new categorial perspective with an emphasis on the morphisms of two familiar categories of graphs. We first investigate the topos-like properties of $\mathcal{SiLlStGraphs}$. For this section, many of the existence constructions follow very closely to those of [3] and [8] for $\mathcal{SiStGraphs}$ as we will see in section 3 of this chapter, but the results pertaining to the lack of categorial structure are new. It is possible that $\mathcal{SiLlStGraphs}$ has been investigated before, but no results pertaining to it are in the literature.

In [4], an elementary topos is a category that has all finite limits, all finite colimits, exponentiation and evaluation, as well as a subobject classifier and all of these are defined by universal mapping properties using only elementary (first order) logic sentences. We investigate these properties in *SiLlStGraphs* starting with finite limits. Mac Lane [12] proves on page 113 that if a category has all products, all equalizers, and a terminal object, then that category has all finite limits.

Theorem 2.1.1. *In SiLlStGraphs*

(i) All finite products exist.

(ii) All equalizers exist.

(iii) A terminal object does not exist.

Before we prove this theorem, we will define the classical constructions for products (as in [8]) and for equalizers (as in [3]); and then in the proof of the theorem, we will show these classical definitions will satisfy the categorial universal mapping definitions

Definition 2.1.2. *Given two graphs G and H in SiLlStGraphs, the* classical product*, $G \times H$, is a graph with the vertex set $V(G) \times V(H)$ in which there is an edge $e \in E(G \times H) \backslash image(\iota_{G \times H})$ with $\partial_{G \times H}(e) = ((u,v)\_(u',v'))$ whenever there exists $f \in E(G) \backslash image(\iota_G)$ with $\partial_G(f) = (u\_u')$ and $g \in E(H) \backslash image(\iota_H)$ with $\partial_H(g) = (v\_v')$. The projection morphisms of $G \times H$, $\pi_G : G \times H \to G$ and $\pi_H : G \times H \to H$, are defined by the set maps $\pi_{V(G)} : V(G \times H) \to V(G)$ where $\pi_{V(G)}((u,v)) = u$, $\pi_{V(H)} : V(G \times H) \to V(H)$ where $\pi_{V(G)}((u,v)) = v$ for all $(u,v) \in V(G \times H)$, $\pi_{E(G)} : E(G \times H) \to E(G)$ where $\pi_{E(G)}((e,f)) = e$, and $\pi_{E(H)} : E(G \times H) \to E(H)$ where $\pi_{E(G)}((e,f)) = f$ for all $(e,f) \in E(G \times H)$.*

Definition 2.1.3. *Let G and H be two graphs in SiLlStGraphs with morphisms $f, g : G \to H$. The* classical equalizer *Eq with inclusion morphism $eq : Eq \hookrightarrow G$ is the vertex induced subgraph of G on the vertex set $V(Eq) = \{v \in V(G) | f(v) = g(v)\}$.*

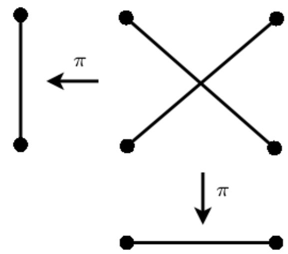


Figure 2.1: Example of $K_2 \times K_2$ in $\mathcal{SiLLStGraphs}$

Proof. Part (i): Let G and H be two graphs in $\mathcal{SiLLStGraphs}$. Let $G \times H$ be defined as above. We first show that π_G and π_H are indeed strict graph homomorphisms.

Let $(e, f) \in E(G \times H)$ with $\partial_{G\times H}((e, f)) = ((u, v)\_(u', v'))$ for some $(u, v), (u', v') \in V(G \times H)$. Consider $\partial_G(\pi_G((e, f)))$. By the definition of $G \times H$, $\partial_G(\pi_G((e, f))) = \partial_G(e) = (u\_u')$. Then since $(u\_u') = (\pi_G(u, v)\_\pi_G(u', v'))$, incidence is preserved and π_G is a graph homomorphism. Clearly by the definition of $G \times H$, π_G will send edges to edges and vertices to vertices, and hence π_G is a strict graph homomorphism. The proof for π_H follows similarly.

Now let X be a graph in $\mathcal{SiLLStGraphs}$ with morphisms $f_G : X \to G$ and $f_H : X \to H$. We need to show there exists a unique morphism $\overline{f} : X \to G \times H$ such that $\pi_G \circ \overline{f} = f_G$ and $\pi_H \circ \overline{f} = f_H$. Let $v \in V(X)$, and suppose $f_G(v) = v_G$ and $f_H(v) = v_H$. Then, since graph homomorphisms must send vertices to vertices, $\overline{f}(v) = (v_G, v_H)$ is the only possibility for such an $\overline{f}$ such that $\pi_G \circ \overline{f} = f_G$ and $\pi_H \circ \overline{f} = f_H$.

Let $e \in E(X)\backslash image(\iota_X)$ with $\partial_X(e) = (v_1\_v_2)$, $f_G(e) = a$ with $\partial_G(a) = (a_1\_a_2)$, and $f_H(e) = b$ with $\partial_H(b) = (b_1\_b_2)$. If $f_G(v_1) = a_1$ and $f_H(v_1) = b_1$ then $f_G(v_2) = a_2$ and $f_H(v_2) = b_2$. Then by the definition of $G \times H$, there is an edge $c \in E(G \times H)\backslash image(\iota_{G\times H})$ with $\partial_{G\times H}(c) = ((a_1, b_1)\_(a_2, b_2))$. Then for $\pi_G \circ \overline{f} = f_G$ and $\pi_H \circ \overline{f} = f_H$ to hold, $\overline{f}(e) = c$ and is uniquely determined. Similarly, if $f_G(v_1) = a_1$ and $f_H(v_1) = b_2$, or $f_G(v_1) = a_2$ and $f_H(v_1) = b_1$, there are edges $c_1, c_2 \in E(G \times H)$ such that $\partial_{G\times H}(c_1) = ((a_1, b_2)\_(a_2, b_1))$, or $\partial_{G\times H}(c_2) = ((a_2, b_1)\_(a_1, b_2))$, and $\overline{f}(e) = c_1$ or $\overline{f}(e) = c_2$ respectively. Hence such an $\overline{f}$ is uniquely determined by f_G and f_H and is clearly a morphism.

Part (ii): Let G and H be graphs in $\mathcal{SiLLStGraphs}$ with morphisms $f, g : G \to H$. Let the equalizer, Eq and $eq : Eq \to G$, be defined as above. Clearly by the definition of Eq, $f \circ eq = g \circ eq$. Now let X be in *SiLlStGraphs* with morphism $h : X \to G$ such that $f \circ h = g \circ h$. We must show there is a unique morphism $\overline{h}$ such that $eq \circ \overline{h} = h$.

Let $v \in V(X)$. Since $f \circ h = g \circ h$, $h(v) \in \{u \in V(G)|f(u) = g(u)\}$. Then $h(v)$ is in the image of $V(Eq)$ under eq, and hence the image of the $V(X)$ under h is contained in the image of $V(Eq)$ under eq. Since inclusion maps are injections, there is a unique $w \in V(G)$ such that $eq(w) = h(v)$. Then since $eq \circ \overline{h} = h$, $\overline{h}(v) = w$ is uniquely determined. Since the image of $V(X)$ under h is contained in the image of $V(Eq)$ under eq, as a vertex set map, $\overline{h}$ is well defined.

Now let $e \in E(X)$ with $\partial_X(e) = (v_1\_v_2)$ for some $v_1, v_2 \in V(X)$. Since $f \circ h = g \circ h$, $h(e) \in \{a \in E(G)|f(a) = g(a)$, and if $\partial_G(a) = (u_1\_u_2)$, $f(u_1) = g(u_1)$ and $f(u_2) = g(u_2)\}$, otherwise $f(h(v_1)) \neq g(h(v_1))$ and $f(h(v_2)) \neq g(h(v_2))$, and hence $f \circ h \neq g \circ h$.

Now note that $\{a \in E(G)|f(a) = g(a)$ and if $\partial_G(a) = (u_1\_u_2)$, $f(u_1) = g(u_1)$ and $f(u_2) = g(u_2)\}$ is precisely $E(Eq)$. So the image of $E(X)$ under h is contained in the image of $E(Eq)$ under eq. Then since eq is an injection, there is a unique $d \in E(Eq)$ such that $eq(d) = h(e)$. Then since $eq \circ \overline{h} = h$, $\overline{h}(e) = d$ is uniquely determined. Since the image of $E(X)$ under h is contained in the image of $E(Eq)$ under eq, as an edge set map, $\overline{h}$ is well defined. It is clear from the definition of $\overline{h}$ that incidence is preserved and edges map to edges. Thus $\overline{h}$ exists is a uniquely determined by h.

Part (iii): We show no terminal object exists by examining the two cases for a graph G in *SiLlStGraphs*. Either G has no edges, $E(G)\backslash image(\iota_G) = \emptyset$, or G has an edge, there exists an edge $e \in E(G)\backslash image(\iota_G)$. If $E(G)\backslash image(\iota_G) = \emptyset$, then since strict graph homomorphisms must send edges to edges, a graph that does contain an edge does not admit a strict graph homomorphism to G. Hence, G cannot be a terminal object.

If there is an edge $e \in E(G)\backslash image(\iota_G)$, since the graphs in *SiLlStGraphs* are loopless, $\partial_G(e) = (u\_v)$ for some $u, v \in V(G)$ where u and v are distinct. The consider the morphisms from K_1, the graph containing only a single vertex, w, to G. Since u and v are distinct, there are two distinct morphisms, $f, g : K_1 \to G$ defined by $f(w) = u$ and $g(w) = v$.

Hence G is not a terminal object since not every graph admits a unique morphism to G. $\square$

In $\mathcal{SiLlStGraphs}$, we notice that the product $K_2 \times K_2$ is isomorphic to the coproduct $K_2 + K_2$. Also in $\mathcal{SiLlStGraphs}$ $K_1 \times K_2$ is isomorphic to $K_1 + K_1$. Furthermore, the canonical projection morphism to each factor of the product in $\mathcal{SiLlStGraphs}$ is an epimorphism because the restriction to the vertex map is surjective (see Proposition 1.5.6).

We now investigate the dual; If all finite coproducts exist, all coequalizers exist, and an initial object exists, then the category has all finite colimits.

Theorem 2.1.4. *In $\mathcal{SiLlStGraphs}$*

(i) All finite coproducts exist.

(ii) Coequalizers do not exist.

(iii) An initial object exists and is the empty graph, $\emptyset$.

Before we begin the proof, we will define the classical coproduct and use it to satisfy the universal mapping definition.

Definition 2.1.5. *Let G and H be graphs in $\mathcal{SiLlStGraphs}$. The* classical coproduct *of G and H, $G+H$, is the disjoint union of graphs G and H with $V(G+H) = V(G)\dot{\cup}V(H)$ and $E(G+H) = E(G)\dot{\cup}E(H)$, natural inclusion maps, $i_G : G \hookrightarrow G+H$ and $i_H : H \hookrightarrow G+H$, and ∂_{G+H} defined by $\partial_{G+H}(e) = \partial_G(a)$ if $e = i_G(a)$ or $\partial_{G+H}(e) = \partial_H(a)$ if $e = i_H(a)$.*

Proof. Part (i): We first note that the classical coproduct defined above is indeed a conceptual graph for since $V(G) \hookrightarrow E(G)$ and $V(H) \hookrightarrow E(H)$, the natural inclusion map $\iota_{G+H} : V(G+H) \hookrightarrow E(G+H)$ exists, which was the missing part in the conceptual graph definition of coproduct.

Now let X be in $\mathcal{SiLlStGraphs}$ with morphisms $f_G : G \to X$ and $f_H : H \to X$. We must show there exists a unique morphism $\overline{f} : G+H \to X$ such that $f_G = \overline{f} \circ i_G$ and $f_H = \overline{f} \circ i_H$. Let $v \in V(G+H)$. If $v = i_G(a)$ for some $a \in V(G)$ then for $\overline{f} \circ i_G(a) = f_G(a)$ to hold, $\overline{f}(v) = f_G(a)$

is uniquely determined. Similarly if $v = i_H(b)$ for some $b \in V(H)$ then for $\overline{f} \circ i_H(b) = f_H(b)$ to hold, $\overline{f}(v) = f_H(b)$ is uniquely determined. Since $V(G+H)$ is the disjoint union of $V(G)$ and $V(H)$, v has a preimage in either $V(G)$ or $V(H)$ exclusively. Hence, as a vertex set map, $\overline{f}$ is well defined.

Let $e \in E(G+H)$, if $e = i_G(a)$ for some $a \in e(G)$ then for $\overline{f} \circ i_G(a) = f_G(a)$ to hold, $\overline{f}(e) = f_G(a)$ is uniquely determined. Similarly if $e = i_H(b)$ for some $b \in E(H)$ then for $\overline{f} \circ i_H(b) = f_H(b)$ to hold, $\overline{f}(e) = f_H(b)$ is uniquely determined. Since $E(G+H)$ is the disjoint union of $E(G)$ and $E(H)$, e has a preimage in either $E(G)$ or $E(H)$ exclusively. Hence, as an edge set map, $\overline{f}$ is well defined. Since any edge in $G+H$ is either in $E(G)$ or $E(H)$ it is incidence to vertices only in $V(G)$ or $V(H)$ respectively. Then since f_A and f_B are strict graph homomorphisms, $\overline{f}$ is a strict graph homomorphism, and is uniquely determined by f_A and f_B.

Part (ii): Assume coequalizers exist. Let $A = B = K_2$, the complete graph on 2 vertices a and b with edge e, and consider the following two morphisms $id, tw : A \to B$ where id is the identity morphism and tw is the morphism where $tw(a) = b$, $tw(b) = a$, and $tw(e) = e$. The coequalizer, $Coeq$ with morphism $coeq : B \to Coeq$ such that $coeq \circ id = coeq \circ tw$, exists by hypothesis. Since $coeq \circ id = coeq \circ tw$ and $id(e) = tw(e) = e$, $coeq(id(e)) = coeq(e) = coeq(tw(e))$, and since morphisms must send edges to edges, $coeq(e)$ is an edge of $Coeq$.

Let $\partial_{Coeq}(coeq(e)) = (u\_v)$ for some $u, v \in V(Coeq)$. Then since morphisms preserve incidence, $coeq(id(a)) = coeq(a)$ is incident to $coeq(e)$, and $coeq(id(b)) = coeq(b)$ is incident to $coeq(e)$. Hence $coeq(a) = u$ or $coeq(a) = v$.

Without loss of generality, let $coeq(a) = u$. Then $coeq(b) = v$, and since $coeq(id(a)) = coeq(tw(a))$, $u = coeq(a) = coeq(b) = v$. Hence e is a loop of $Coeq$, which contradicts our hypothesis that $Coeq$ was in $\mathcal{SiLlStGraphs}$. Thus coequalizers do not exist in $\mathcal{SiLlStGraphs}$.

Part (iii): A terminal object must have a unique morphism to every object in the category. Clearly $\emptyset$ has a unique morphism, namely inclusion, to every graph in $\mathcal{SiLlStGraphs}$. $\square$

It is evident that from part (ii), since we do not allow loops in our graphs in $\mathcal{SiLlStGraphs}$

we cannot use equivalence relations to form quotient graphs. But we will see that allowing loops is enough to certify the existence of coequalizers in $\mathcal{SiStGraphs}$. We now investigate the existence of exponentiation and evaluation and a subobject classifier in $\mathcal{SiLlStGraphs}$.

Before we do so, we need an adjoint relationship between the underlying vertex set functor $|-|_V : \mathcal{SiLlStGraphs} \rightsquigarrow \mathcal{Sets}$ where $|G|_V = V(G)$ for a graph G and $|f|_V = f_V$ for a morphism f, and the free graph functor $F(-) : \mathcal{Sets} \rightsquigarrow \mathcal{SiLlStGraphs}$, namely $F(-)$ is left adjoint to $|-|_V$, denoted $F(-) \dashv |-|_V$. To show $F(-) \dashv |-|_V$, we will show the empty edge graphs where $F(X)$ is the empty edge set graph with vertex set X, $E(F(X)) \backslash image(\iota_{F(X)}) = \emptyset$, and $F(g)$ is the strict graph homomorphism where $F(g)_V = g$ and $F(g)_E = g$ satisfy the universal mapping property of free objects.

Lemma 2.1.6. *The empty edge graphs, K_n^c for $n \geq 0$, are the free objects in $\mathcal{SiLlStGraphs}$; furthermore the free graph functor F is left adjoint to the underlying vertex set functor $|-|_V$, $F \dashv |-|_V$, i.e. $hom_{\mathcal{SiLlStGraphs}}(F(X), G) \cong hom_{\mathcal{Sets}}(X, |G|_V)$.*

Proof. Let X be a set in $\mathcal{Sets}$ with n elements, and let $F(X) = K_n^c$ where $V(F(X)) = X$. Now let G be a graph in $\mathcal{SiLlStGraphs}$ such that there is a function $g : X \to |G|_V$. We must show there is a unique graph morphism $\overline{g} : F(X) \to G$ such that $g = |\overline{g}|_V \circ u$ for some $u : X \to |F(X)|_V$. Note that $|F(X)|_V = V(F(X)) = X$. Hence define the function $u : X \to |F(X)|_V$ as $u = 1_X$.

Let $\overline{g}$ be the pair of function maps $\overline{g}_V = g$ and $\overline{g}_E = g$. Since there are no edges in $F(X)$, incidence is clearly preserved, edges are sent to edges vacuously, and $\overline{g}$ is a strict graph homomorphism. Then since $g = |\overline{g}|_V \circ u$ must hold, $u = 1_X$, and $|\overline{g}|_V = \overline{g}_V = g$, $\overline{g}$ is uniquely determined by g. $\square$

Robert Goldblatt proves [4, p. 441] that satisfying the universal mapping property for a free object in any concrete category provides the adjoint relationship: $F \dashv |-|$. Also, in general, if products exist and exponentiation with evaluation exists, then $(-) \times Y \dashv (-)^Y$; we will consider this situation in our graph category $\mathcal{SiLlStGraphs}$.

Proposition 2.1.7. *In $\mathcal{SiLlStGraphs}$*

(i) Exponentiation with evaluation does not exist.

(ii) A subobject classifier does not exist.

Proof. Part (i): Assume exponentiation with evaluation exists. We will first show that given this hypothesis, there is the adjoint bijection between $hom_{\mathcal{SiLlStGraphs}}(X \times Y, Z)$ and $hom_{\mathcal{SiLlStGraphs}}(X, Z^Y)$ for all graphs X, Y and Z. We prove this via [4] page 71.

Let X, Y, and Z be graphs in $\mathcal{SiLlStGraphs}$. We show there is a bijection $\phi : hom_{\mathcal{SiLlStGraphs}}(X \times Y, Z) \to hom_{\mathcal{SiLlStGraphs}}(X, Z^Y)$. Since, by hypothesis, exponentiation and evaluation exists, for $g : X \times Y \to Z$ there is a unique map $\overline{g} : X \to Z^Y$ such that for the evaluation map $ev : Z^Y \times X \to Z$, $ev \circ (\overline{g} \times 1_Y) = g$. We define ϕ by $\phi(g) = \overline{g}$.

If $\overline{g} = \overline{h}$ for some $\overline{h} : X \to Z^Y$, then $ev \circ (\overline{g} \times 1_Y) = ev \circ (\overline{h} \times 1_Y)$ and thus $g = h$. Hence ϕ is injective. Now let $\overline{h} : X \to Z^Y$ and define $g : X \times Y \to Z$ by $g = ev \circ (\overline{h} \times 1_Y)$. Then since exponentiation and evaluation exists, there exists a unique $\overline{g} : X \to Z^Y$ such that $g = ev \circ \overline{g} \times 1_Y$. Since $\overline{g}$ is unique, $\overline{g} = \overline{h}$. Hence ϕ is surjective, and thus a bijection of sets.

We can now achieve our contradiction. Consider $K_2^{K_2}$ where K_2 has vertices a and b and edge e. Since the free graph functor is left adjoint to the underlying vertex set functor, $V(K_2^{K_2}) = \{id, tw, \ulcorner a \urcorner, \ulcorner b \urcorner\}$. Where id is the identity vertex map, tw (called the "twist" map) is defined by $tw(a) = b$ and $tw(b) = a$, $\ulcorner a \urcorner$ is the "constantly a" map, and $\ulcorner b \urcorner$ is the "constantly b" map.

Since in $\mathcal{SiLlStGraphs}$, $K_2 \times K_2 \cong K_2 + K_2$ and there are 4 morphisms from $K_2 + K_2$ to K_2, namely the 4 combinations of id and tw from each component of $K_2 + K_2$ to K_2. Then since there is a bijection in sets of $hom_{\mathcal{SiLlStGraphs}}(K_2 \times K_2, K_2)$ and $hom_{\mathcal{SiLlStGraphs}}(K_2, K_2^{K_2})$ there are 4 morphisms from K_2 to $K_2^{K_2}$. Since K_2 admits two morphisms to any edge of any graph in $\mathcal{SiLlStGraphs}$, there are 2 edges in $K_2^{K_2}$.

Let $e \in E(K_2^{K_2})$ such that $\partial_{K_2^{K_2}}(e) = (\ulcorner a \urcorner \_ \ulcorner b \urcorner)$. We will now show that no other edge can be in $K_2^{K_2}$ without causing a contradiction in the evaluation map $ev : K_2^{K_2} \times K_2 \to K_2$. Let $f \in E(K_2^{K_2})$ be the other edge distinct from e. Since e is incident to $\ulcorner a \urcorner$ and $\ulcorner b \urcorner$, f must be incident to id or tw, for if not, it would also be incident to $\ulcorner a \urcorner$ and $\ulcorner b \urcorner$, which would be a multiple edge in $K_2^{K_2}$.

Case 1: f is incident to id. If f is also incident to tw, there is an edge f' in $K_2^{K_2} \times K_2$ such that $\partial_{K_2^{K_2} \times K_2}(f') = ((id, a)\_(tw, b))$. Hence since ev is a strict graph homomorphism, $\partial_{K_2}(ev(f')) = (a\_a)$, a contradiction to K_2 being loopless.

If f is also incident to $\ulcorner a \urcorner$, there is an edge f' in $K_2^{K_2} \times K_2$ such that $\partial_{K_2^{K_2} \times K_2}(f') = ((id, a)\_(\ulcorner a \urcorner, b))$. Hence $\partial_{K_2}(ev(f')) = (a\_a)$, a contradiction to K_2 being loopless. If f is also incident to $\ulcorner b \urcorner$, there is an edge f' in $K_2^{K_2} \times K_2$ such that $\partial_{K_2^{K_2} \times K_2}(f') = ((id, b)\_(\ulcorner b \urcorner, a))$. Hence $\partial_{K_2}(ev(f')) = (b\_b)$, a contradiction to K_2 being loopless. Hence f cannot be incident to id.

Case 2: f is incident to tw. From case 1, f cannot also be incident to id. If f is incident to $\ulcorner a \urcorner$ or $\ulcorner b \urcorner$, a similar contradiction to that of f being incident to id and $\ulcorner a \urcorner$ or $\ulcorner b \urcorner$ arises. Therefore there is at most one edge in $K_2^{K_2}$, a contradiction. Hence exponentiation with evaluation does not exist in $\mathcal{SiLLStGraphs}$.

Part (ii): Since a terminal object does not exist in $\mathcal{SiLLStGraphs}$, there is no subobject classifier. $\square$

2.2 Other Categorial Constructions in $\mathcal{SiLLStGraphs}$

We now move on to consider some other categorial properties in $\mathcal{SiLLStGraphs}$. The results in this section are all new results. We first begin with a lemma about the behavior of morphisms in this category.

Lemma 2.2.1. *In $\mathcal{SiLLStGraphs}$ the inclusion of K_n^c into a graph G with n vertices ($n > 1$) and at least one edge is an epimorphism (but not a surjection); and, of course, this inclusion is also a monomorphism (and an injection); furthermore this inclusion is not an isomorphism.*

Proof. Without loss of generality, since every empty edge graph on n vertices is isomorphic to every other empty edge graph on n vertices, assume $V(K_n^c) = V(G)$. Then let $i : K_n^c \to G$ be the inclusion morphism. Let f and g be morphisms $f, g : G \to H$ for some graph H such that $f \circ i = g \circ i$, and let $v \in V(G)$. Since $V(G) = V(K_n^c)$, $f(v) = f \circ i(v) = g \circ i(v) = g(v)$ and f and g agree on the vertices of G.

Now let e be an edge of G, $e \in E(G) \backslash image(\iota_G)$. Since the graphs in $\mathcal{SiLlStGraphs}$ are loopless, $\partial_G(e) = (u\_v)$ for distinct vertices $u, v \in V(G)$. Since the morphisms in $\mathcal{SiLlStGraphs}$ are strict and preserve incidence, $f(e)$ is incident to $f(u)$ and $f(v)$. Similarly $g(e)$ is incident to $g(u)$ and $g(v)$.

Since f and g agree on the vertices of G, $f(e)$ is incident to $f(u) = g(u)$ and $f(v) = g(v)$. Thus $f(e)$ and $g(e)$ are incident to the same vertices, and since the graphs of $\mathcal{SiLlStGraphs}$ are simple, $f(e) = g(e)$. Hence i is an epimorphism, and clearly not surjective.

Since the inclusion is an injection on the vertex sets (which composes K_n^c) it is a monomorphism. It is not an isomorphism as K_n does not admit a morphism to K_n^c (for $n > 1$). □

Since the inclusion of K_n^c into any graph G is a monomorphism, it is also evident that a morphism that is both a monomorphism and an epimorphism is not necessarily a isomorphism in $\mathcal{SiLlStGraphs}$. Lemma 2.1.6 states that the free objects in $\mathcal{SiLlStGraphs}$ are the empty edge graphs. The following proposition shows us there are no co-free objects in $\mathcal{SiLlStGraphs}$.

Lemma 2.2.2. *There are no cofree objects in* $\mathcal{SiLlStGraphs}$*.*

Proof. Assume there are. Let $X = \{x\}$ in $\mathcal{Sets}$ and $C(X)$ be the co-free graph associated with X and function $c : |C(X)|_V \to X$. Consider K_2 with vertices a and b and edge e and set function $g : |K_2|_V \to X$ defined by $g(a) = g(b) = x$. Then since $C(X)$ is a cofree object, there is a unique morphism in $\mathcal{SiLlStGraphs}$, $\overline{g} : K_2 \to C(X)$, such that $g = c \circ |\overline{g}|_V$. Since $\overline{g}$ is a strict graph homomorphism, it must send e to an edge in $C(X)$. Thus $\overline{g}(e) = f$ for some $f \in E(C(X))$. Since graph homomorphisms preserve incidence, f is incident to $\overline{g}(a) = |\overline{g}|_V(a) = a'$ for some vertex $a' \in V(C(X))$ and $\overline{g}(b) = |\overline{g}|_V(b) = b'$ for some vertex $b' \in V(C(X))$.

Since $C(X)$ is loopless, $a' \neq b'$. Then since $g = c \circ |\overline{g}|_V$, $g(a) = c(|\overline{g}|_V(a)) = x$ and $g(b) = c(|\overline{g}|_V(b)) = x$, $c(a') = c(b') = x$. Now consider the morphism $\overline{h} : K_2 \to C(X)$ defined by $\overline{h}(e) = f$, $\overline{h}(a) = b'$ and $\overline{h}(b) = a'$. Clearly $\overline{h} \neq \overline{g}$. Then $c(|\overline{h}|_V(a)) = c(b') = x = g(a)$ and $c(|\overline{h}|_V(b)) = c(a') = x = g(b)$. Thus $c \circ |\overline{h}|_V = g$, and $\overline{g}$ is not unique, which is a contradiction to the universal mapping description of the cofree object. □

The definitions for free objects and cofree objects are dependent on the category being a concrete category. We move on to other categorial constructions that are defined for any abstract category. We start with the injective objects and projective objects (see Appendix A for definitions). Note, since $|-|$ is reserved for the underlying set functor, when referring to the cardinality of a set X we will use $\sharp(X)$.

Theorem 2.2.3. *In $\mathcal{SiLLStGraphs}$*

(i): The projective objects are precisely the free objects and there are enough projective objects.

(ii): There are no injective objects.

Proof. Part (i): First, by Proposition 1.5.6., we note that if $f : A \to B$ is an epimorphism in $\mathcal{SiLLStGraphs}$ then the vertex set function f_V is surjective. We show that the free objects are projective objects. Clearly the empty graph $\emptyset$ is projective since it is the initial object. Now let X be a non-empty set in $\mathcal{Sets}$, G be a graph in $\mathcal{SiLLStGraphs}$ with a morphism $h : F(X) \to G$, and H be a graph in $\mathcal{SiLLStGraphs}$ with an epimorphism $g : H \to G$. We must show that there is a morphism $\overline{h} : F(X) \to H$ such that $g \circ \overline{h} = h$.

Since g is an epimorphism, g_V is a surjective function. Hence for all $v_i \in V(F(X))$, there is a $u_i \in V(H)$ such that $g(u_i) = h(v_i)$. Then define $\overline{h}(v_i) = u_i$ for every $v_i \in V(F(X))$. Then $g(\overline{h}(v_i)) = g(u_i) = h(v_i)$ for every vertex v_i of $F(X)$. Since $F(X)$ contains no edges, $\overline{h}$ is a strict graph homomorphism. Thus $F(X)$ is projective.

Now let A be a graph in $\mathcal{SiLLStGraphs}$ with at least 1 edge, and G be a graph on n vertices ($n > 1$) in $\mathcal{SiLLStGraphs}$ with a morphism $h : A \to G$. By Lemma 2.2.1. there is an epimorphism

$e : K_n^c \to G$. Since strict graph homomorphisms must send edges to edges, A admits no morphism to K_n^c. Thus A is not projective.

Let G be a graph in *SiLlStGraphs* with n vertices. To show there are enough projectives, we must show there is a projective object H and an epimorphism $e : H \to G$. By Lemma 2.2.1. the projective object K_n^c admits an epimorphism to G.

Part (ii): Suppose there were injective objects. Let Q be an injective object in *SiLlStGraphs*. Consider the complete graph K with $\sharp(V(K)) > \sharp(V(Q))$, the cardinality of $V(K)$ is greater than that of $V(Q)$. Then consider the morphisms $id : Q \to Q$ the identity on Q and $f : Q \to K$, the inclusion morphism of Q into K. Since inclusion morphisms are injections, they are monomorphisms.

Then since Q is injective, there is a morphism $\overline{f} : K \to Q$ such that $\overline{f} \circ f = id$. Since $\sharp(V(K)) > \sharp(V(Q))$ and $\overline{f}_V$ is a set map, there are two distinct vertices $u, v \in V(K)$ such that $\overline{f}(u) = \overline{f}(v)$. Since $\overline{f}$ is a strong morphism and K is a complete graph, the edge e incident to u and v in K must be sent to an edge in Q. Since graph homomorphisms preserve incidence and $\overline{f}(u) = \overline{f}(v)$, $\partial_Q(\overline{f}(e)) = (\overline{f}(u)\_\overline{f}(u))$, and $\overline{f}(e)$ is a loop. This contradicts Q being loopless. Hence no injective objects exist. $\square$

The last topic for consideration in *SiLlStGraphs* are the generators and cogenerators.

Theorem 2.2.4. *In SiLlStGraphs*

(i): the empty edge graphs, K_n^c, are precisely the generators (for $n \geq 1$).

(ii): no cogenerators exist.

Proof. Part (i): Let $n \geq 1$. First we show K_n^c is a generator, then we show that any graph with an edge is not a generator. Let X and Y be graphs in *SiLlStGraphs* with morphisms $f, g : X \to Y$ such that $f \neq g$. Then there is a vertex $v \in V(X)$ such that $f(x) \neq g(x)$, otherwise since the morphisms preserve incidence and there is at most one edge between any two vertices, $f(e) = g(e)$ for all edges $e \in E(X)$ and $f = g$.

First note $K_1 = K_1^c$. Now consider the map $h : K_1 \to X$ that sends the single vertex of K_1,

u, to v. Then $f(h(u)) = f(v) \neq g(v) = g(h(u))$. Hence $f \circ h \neq g \circ h$. Hence K_1 is a generator.

To show K_n^c is a generator, we consider the morphism $\pi : K_n^c \to K_1$ that sends every vertex of K_n^c to u. Then clearly $f \circ (h \circ \pi) \neq g \circ (h \circ \pi)$, and hence K_n^c is a generator of $\mathcal{SiLLStGraphs}$.

Now let G be a graph in $\mathcal{SiLLStGraphs}$ with at least one edge. Consider K_2^c with two vertices u and v with two morphisms $id, tw : K_2^c \to K_2^c$ where id is the identity morphism and tw is the "twist" morphism defined by $tw(u) = v$ and $tw(v) = u$. Clearly $id \neq tw$, but since morphisms must send edges to edges, G admits no map to K_2^c. Thus G is not a generator.

Part (ii): To show there are no cogenerators in $\mathcal{SiLLStGraphs}$, we show there is no graph X such that any graph G admits a morphism to X. Assume such a graph X exists. Consider the complete graph K such that $\sharp(V(K)) > \sharp(V(X))$. By hypothesis there is a morphism $f : K \to X$. Since $\sharp(V(K)) > \sharp(V(X))$ and $\overline{f}_V$ is a set map, there are two distinct vertices $u, v \in V(K)$ such that $\overline{f}(u) = \overline{f}(v)$. Let e be the edge in K incident to both u and v. Since graph homomorphisms preserve incidence and $\overline{f}(u) = \overline{f}(v)$, $\partial_X(\overline{f}(e)) = (\overline{f}(u)\_\overline{f}(u))$. Then since edges must be sent to edges, $\overline{f}(e)$ is a loop. This contradicts X being loopless. Hence no such object exists.

We now note that every graph G with at least two vertices admits at least two distinct morphisms to the complete graph K with $\sharp(V(K)) = \sharp(V(G))$, as the automorphism group of K is the symmetric group on $V(K)$. This fact coupled with the fact there is no graph such that any other graph admits a morphism to it proves no cogenerators exist. $\square$

We will now consider the same constructions in these last two sections in the category where we allow our graphs to have at most one loop on every vertex, but where the morphisms are still strict.

2.3 Existence of Topos-like Properties in *SiStGraphs*

We now consider the category *SiStGraphs* where the graphs are simple, but do allow at most one loop on any vertex, and the morphisms are still strict graph homomorphisms. The results in this section, except those about the subobject classifier, are known (see [3]) as this is the category most commonly used in the literature. We begin as we did in section 2.2, by investigating finite limits.

Theorem 2.3.1. *In SiStGraphs*

(i) All finite products exist.

(ii) All equalizers exist.

(iii) The graph with a single vertex and a loop at that vertex is the terminal object.

Hence all finite limits exist.

Proof. The products and equalizers in this proof have the same definition as those in *SiLlStGraphs* (definition 2.1.2. and definition 2.1.4.), and the proof of Part (i) and Part (ii) follows exactly as the proof given in Theorem 2.1.1.

Part (iii): Let $\mathbf{1}$ be defined as the graph with a single vertex, v, and a loop at that vertex, ℓ. Let G be a graph in *SiStGraphs*. Then the map $f : G \to \mathbf{1}$ that sends every vertex of G to v and every edge of G to ℓ preserves incidence and is clearly strict. Hence f is a morphism of *SiStGraphs*. Since any morphism in *SiStGraphs* must send vertices to vertices and edges to edges, f is the only morphism from G to $\mathbf{1}$. Thus $\mathbf{1}$ is the terminal object. $\square$

Hence all finite limits exist in *SiStGraphs*. We will now investigate finite colimits. Before we do so, we must define a new graph construction, the classical quotient graph.

Definition 2.3.2. *Given a graph G and an equivalence relation $\sim$ on $V(G)$, the* classical quotient graph*, Q, has vertex set $V(Q) = V(G)/\sim$ and there is an edge $e' \in E(Q)\backslash image(\iota_Q)$ with $\partial_Q(e') = ([u]\_[v])$ if there is a single edge $e \in E(G)\backslash image(\iota_G)$ with $\partial_G(e) = (u\_v)$ for* *some u and v representatives of the equivalence classes $[u]$ and $[v]$ respectively.*

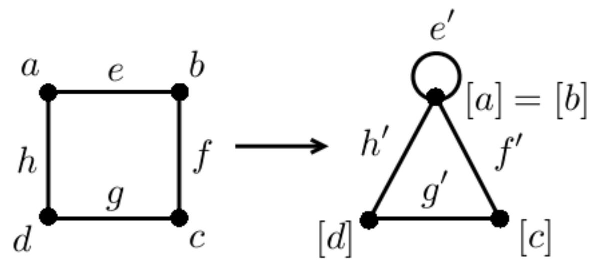


Figure 2.2: Example of a Quotient Graph in $\mathcal{SiStGraphs}$

Since there is at most one edge between any two distinct vertices and at most one loop at any single vertex, the quotient graph is clearly a graph in $\mathcal{SiStGraphs}$. We may think of the quotient graph as the graph obtained by identifying the vertices in the equivalence class, and then identifying any multiple edges or multiple loops that arise from the vertex identification. We now show there is a natural quotient morphism from a graph to its quotient graph.

Proposition 2.3.3. *Given a graph G of $\mathcal{SiStGraphs}$ and Q the quotient graph of G defined by an equivalence relation $\sim$ on $V(G)$, then there is a quotient morphism $q : G \to Q$ defined by $q(v) = [v]$ for $v \in V(G)$ and $q(e) = e'$ for $e \in E(G)\backslash image(\iota_G)$ where $\partial_G(e) = (u\_v)$ and e' is the edge in $E(Q)\backslash image(\iota_Q)$ with $\partial_Q(e') = ([u]\_[v])$.*

Proof. Clearly q sends vertices to vertices and edges to edges, so we must check that it preserves incidence. Let $e \in E(G)$ with $\partial_G(e) = (u\_v)$ for some $u, v \in V(G)$. Then since u is in the equivalence class $[u]$ and v is in the equivalence class $[v]$ with u and v incidence to e, then e' is in $E(Q)$ with $\partial_Q(e') = ([u]\_[v])$. Hence $\partial_Q(q(e)) = \partial_Q(e') = ([u]\_[v]) = (q(u)\_q(v))$ and incidence is preserved. $\square$

We can now define the classical coequalizer in $\mathcal{SiStGraphs}$, and use it for the categorial (universal mapping) coequalizer.

Definition 2.3.4. *Given graphs G and H in $\mathcal{SiStGraphs}$ with morphisms $f, g : G \to H$, the* coequalizer*, $Coeq$, with the quotient morphism $coeq : H \to Coeq$ is the quotient graph of H under the smallest equivalence relation $\sim$ on $V(H)$ generated by $f(x) \sim g(x)$ for some $x \in V(G)$.*

Theorem 2.3.5. *In $\mathcal{SiStGraphs}$*

(i) All finite coproducts exist.

(ii) All coequalizers exist.

(iii) The empty graph is the initial object.

Hence all finite colimits exist.

Proof. The classical coproducts in $\mathcal{SiStGraphs}$ use the same classically defined coproducts as those in $\mathcal{SiLlStGraphs}$ (definition 2.1.5.). The proofs of Part (i) and Part (iii) follow exactly as the proof given in Theorem 2.1.4.

Part (ii): Let G and H be graphs in $\mathcal{SiStGraphs}$ with morphisms $f, g : G \to H$. Consider $x \in V(G)$. Since $V(Coeq) = V(H)/\sim$ where $\sim$ is the (smallest) equivalence relation on $V(H)$ generated by $f(x) \sim g(x)$ for $x \in V(G)$, i.e. $g(x)$ is a representative of $[f(x)]$ and $coeq(f(x)) = [f(x)] = [g(x)] = coeq(g(x))$. Now consider $e \in E(G)\backslash image(\iota_G)$ with $\partial_G(e) = (u\_v)$ for some vertices $u, v \in V(G)$. Since $f(u) \sim g(u)$ and $f(v) \sim g(v)$, $\partial_{Coeq}(f(e)) = ([f(u)]\_[f(v)]) = ([g(u)]\_[g(v)]) = \partial_{Coeq}(g(e))$ and $coeq(f(e)) = coeq(g(e))$. Hence $coeq \circ f = coeq \circ g$.

Let X be a graph in $\mathcal{SiStGraphs}$ with morphism $h : H \to X$ such that $h \circ f = h \circ g$. We must show there is a unique morphism $\overline{h} : Coeq \to X$ such that $h = \overline{h} \circ coeq$. By the construction of $\sim$, all the vertices in the same equivalence class of $\sim$ must be sent to a single vertex in X or we would contradict $h \circ f = h \circ g$. For $h = \overline{h} \circ coeq$ to hold, define $\overline{h}([u]) = h(u)$ for u a representative of $[u] \in V(Coeq)$, since all choices of u will be mapped to the same vertex in X, this is a well defined vertex set map and is uniquely determined by h.

By construction of a classical quotient graph, an edge $e' \in E(Coeq)\backslash image(\iota_{Coeq})$ with $\partial_{Coeq}(e') = ([u]\_[v])$ exists only if there is an edge $e \in E(H)\backslash image(\iota_H)$ with $\partial_H(e) = (u\_v)$ for some $u, v \in V(H)$ representatives of $[u]$ and $[v]$ respectively. Since all the vertices in $[u]$

and all the vertices $[v]$ are mapped to the vertices $h(u)$ and $h(v)$ respectively, for such an edge $e' \in E(Coeq)\backslash image(\iota_{Coeq})$, there is an edge $e \in E(H)$ such that $\partial_X(h(e)) = (h(u)\_h(v))$. Then for $h = \overline{h} \circ coeq$ to hold, define $\overline{h}(e') = h(e)$ and, as edge set map, $\overline{h}$ is uniquely determined by h. Since this construction preserves incidence and sends edges (strictly) to edges, $\overline{h}$ is a morphism in *SiStGraphs* and is uniquely determined by h. □

We now investigate the last two topos-like properties of *SiStGraphs*, exponentiation with evaluation and a subobject classifier. We first define the classical exponential graph, generalizing that in [8, p. 46].

Definition 2.3.6. *Given graphs G and H in SiStGraphs, the* classical exponential graph*, H^G has vertex set $V(H^G) = \{f : V(G) \to V(H)\}$, (i.e the collection of vertex set functions: $V(H^G) = V(H)^{V(G)}$), and $e' \in E(H^G)\backslash image(\iota_{H^G})$ with $\partial_{H^G}(e') = (f\_g)$ only when for all vertices $u, v \in V(G)$, if there is an edge $e \in E(G)\backslash image(\iota_G)$ with $\partial_G(e) = (u\_v)$ then there is an edge $d \in E(H)\backslash image(\iota_H)$ such that $\partial_H(d) = (f(u)\_g(v))$.*

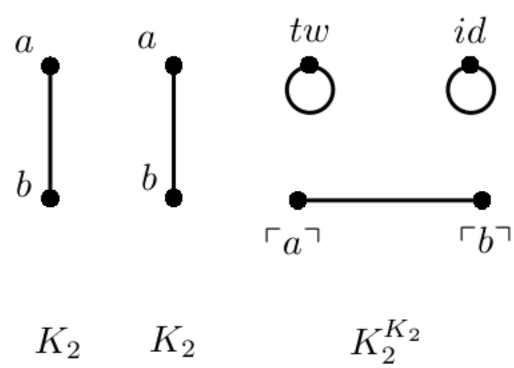


Figure 2.3: Example of a $K_2^{K_2}$ in *SiStGraphs*

Theorem 2.3.7. *In SiStGraphs*

(i) Exponentiation and evaluation exists.

(ii) A subobject classifier does not exist.

Hence along with Theorem 2.3.1. SiStGraphs is cartesian closed, but not a topos.

Proof. Part (i): First, define $ev : H^G \times G \to H$ by $ev((f,v)) = f(v)$ for all vertices $(f,v) \in V(H^G \times G)(= V(H)^{V(G)} \times V(G))$ and for $e \in E(H^G \times G)$ such that $\partial_{H^G \times G}(e) = ((f,v)\_(g,u))$ define $ev(e) = d$ for $d \in E(H)$ with $\partial_H(d) = (f(v)\_g(u))$. Such a d exists by construction of H^G, and by construction of H^G, ev is a strict graph homomorphism.

Now let X be a graph in $\mathcal{SiStGraphs}$ with morphism $g : X \times G \to H$. We must show there is a unique morphism $\overline{g} : X \to H^G$ such that $g = ev \circ (\overline{g} \times 1_G)$.

Let $x \in V(X)$ and consider $\{x\} \times G := \{(x,v)|(x,v) \in V(X \times G)$ for some $v \in V(G)\} \subseteq V(X \times G)$. Then $g|_{\{x\}\times G}$ induces a function $f_x : V(G) \to V(H)$ defined by $f_x(v) = g((x,v))$. Then for $g = ev \circ (\overline{g} \times 1_G)$ to hold, define $\overline{g}(x) = f_x$, and $\overline{g}$ is a vertex set function uniquely determined by g.

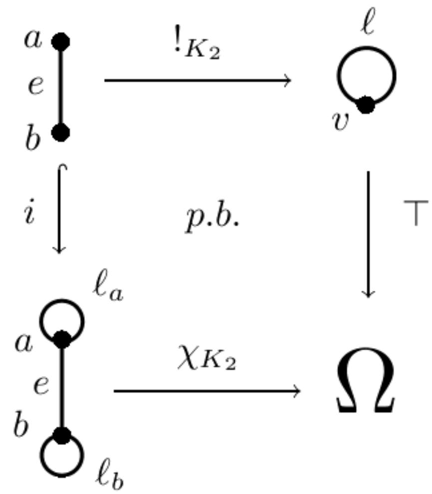
Now let $e \in E(X)$ with $\partial_X(e) = (x_1\_x_2)$. Consider $\{e\} \times G := \{d \in E(X \times G)|\partial_{X\times G}(d) = ((x_1,u)\_(x_2,v))$ for some $u,v \in V(G)\} \subseteq E(X \times G)$. Note that for an edge $d \in \{e\} \times G$, $\partial_{X\times G}(d) = ((x_1,u)\_(x_2,v))$ for some $u,v \in V(G)$ implies there is an edge $d' \in E(G)$ such that $\partial_G(d') = (u\_v)$.

For such a d, since g preserves incidence, $\partial_H(g(d)) = (g(x_1,u)\_g(x_2,v)) = (f_{x_1}(u)\_f_{x_2}(v))$. Then for $g = ev \circ (\overline{g} \times 1_G)$ to hold, define $\overline{g}(e) = a$ where $\partial_{H^G}(a) = (f_{x_1}\_f_{x_2})$ which exists by definition of H^G, and is uniquely determined by g. Clearly $\overline{g}$ is a morphism in $\mathcal{SiStGraphs}$ and is uniquely determined by g.

Part (ii): Given graphs A, B, and C in $\mathcal{SiStGraphs}$ with $f : A \to C$ and $g : B \to C$, we can find the pullback of f and g by taking the equalizer of $f', g' : A \times B \to C$ where $f' = f \circ \pi_A$ and $g' = g \circ \pi_B$ (for a proof see Proposition A.0.10.). Then the equalizer defined in this way will be the pullback with $\pi_A \circ eq : Eq \to A$ and $\pi_B \circ eq : Eq \to B$ such that $f \circ \pi_a \circ eq = g \circ \pi_B \circ eq$, where Eq is the vertex induce subgraph of $A \times B$ on $V(Eq) = \{(a,b) \in V(A \times B)|f(\pi_A((a,b))) = g(\pi_B((a,b)))\}$ with $eq : Eq \to A \times B$ the inclusion morphism.

Assume a subobject classifier, Ω, exists with morphism $\top : \mathbf{1} \to \Omega$. Consider K_2 having vertices a and b with an edge e between them with $!_{K_2} : K_2 \to \mathbf{1}$ the unique morphism to the terminal object. Let $i : K_2 \hookrightarrow K_2^\ell$ be inclusion where K_2^ℓ is K_2 together with a loops ℓ_a and

ℓ_b at vertices a and b respectively. Then there exists a unique $\chi_{K_2} : K_2^\ell \to \Omega$ such that K_2 is the pullback of $\top$ and χ_{K_2}. Then $\top \circ !_{K_2} = \chi_{K_2} \circ i$.



Since $!_{K_2}(a) = !_{K_2}(b) = v$ for v the vertex of $\mathbf{1}$ and $!_{K_2}(e) = \ell$ for ℓ the loop of $\mathbf{1}$, and since morphisms much send edges to edges, $\top(!_{K_2}(a)) = \top(!_{K_2}(b)) = \top(v)$ and $\top(!_{K_2}(e)) = \top(\ell)$ where $\partial_\Omega(\top(\ell)) = (\top(v)\_\top(v))$. Since $\top \circ !_{K_2} = \chi_{K_2} \circ i$, $\chi_{K_2}(i(a)) = \chi_{K_2}(i(b)) = \top(v)$. Then since morphisms preserve incidence, $\partial_\Omega(\chi_{K_2}(\ell_a)) = \partial_\Omega(\chi_{K_2}(\ell_b)) = (\top(v)\_\top(v))$. Since graphs in *SiStGraphs* can have at most one loop at any vertex, and morphisms must send edges to edges, $\chi_{K_2}(\ell_a) = \chi_{K_2}(\ell_b) = \top(\ell)$.

Now consider the pullback of χ_{K_2} and $\top$. It is the vertex induced subgraph of $K_2^\ell \times \mathbf{1}$ on $V(Eq) = \{(c, v) \in V(K_2^\ell \times \mathbf{1}) | \chi_{K_2}(\pi_{K_2^\ell}((c, v))) = \top(\pi_{\mathbf{1}}((c, v)))\}$. However, since $K_2^\ell \times \mathbf{1} \cong K_2^\ell$ and $\chi_{K_2}(\pi_{K_2^\ell}((a, v))) = \chi_{K_2}(a) = \top(v) = \chi_{K_2}(b) = \chi_{K_2}(\pi_{K_2^\ell}((b, v)))$, $V(Eq) = \{(a, v), (b, v)\}$ and $Eq \cong K_2^\ell$. This contradicts that K_2 is the pullback of χ_{K_2} and $\top$. Hence no subobject classifier exists. $\square$

Since Theorem 2.3.1. states that all finite limits exist, Theorem 2.3.5 states that all finite colimits exist, and Theorem 2.3.7 states that exponentiation with evaluation exists, then *SiStGraphs* is a "near" topos. The axiom it does not satisfy is the existence of a subobject classifier.

2.4 Other Categorial Constructions in $\mathcal{SiStGraphs}$

We continue in the same fashion as our investigation of $\mathcal{SiLlStGraphs}$. The results in this section are new results. We first show that epimorphisms in $\mathcal{SiStGraphs}$ are still not guaranteed to be surjections (see p. 28 for an analogous proposition in $\mathcal{SiLlStGraphs}$).

Lemma 2.4.1. *In $\mathcal{SiStGraphs}$ the inclusion of K_n^c into a graph G with n vertices ($n > 1$) and at least one edge is an epimorphism (but not a surjection); and of course, the inclusion is also a monomorphism (and an injection); furthermore, this inclusion is not an isomorphism.*

Proof. Without loss of generality, since every empty edge graph on n vertices is isomorphic to every other empty edge graph on n vertices, assume $V(K_n^c) = V(G)$. Then let $i : K_n^c \to G$ be the inclusion morphism. Let f and g be morphisms $f, g : G \to H$ for some graph H such that $f \circ i = g \circ i$, and let $v \in V(G)$. Since $V(G) = V(K_n^c)$, $f(v) = f \circ i(v) = g \circ i(v) = g(v)$ and f and g agree on the vertices of G.

For a non-loop edge $e \in E(G)$, the proof follows as in the proof of Lemma 2.2.1. and $f(e) = g(e)$. So consider a loop $\ell \in E(G)$ such that $\partial_G(\ell) = (u\_u)$ for some $u \in V(G)$. Since morphisms preserve incidence and f and g agree on the vertices of G, $\partial_H(f(\ell)) = (f(u)\_f(u)) = (g(u)\_g(u)) = \partial_H(g(\ell))$. Since morphisms must send edges to edges, there is at most one loop incident to any vertex, and $f(\ell)$ is incident to the same vertices as $g(\ell)$, $f(\ell) = g(\ell)$. Hence $f = g$ and i is an epimorphism. □

We now investigate the free objects and the cofree objects. The free objects are the same as in $\mathcal{SiLlStGraphs}$, but since loops are allowed in $\mathcal{SiStGraphs}$, we do have cofree objects.

Lemma 2.4.2. *In $\mathcal{SiStGraphs}$*

(i) the free objects are K_n^c for $n \geq 0$.

(ii) the cofree objects are the complete graphs with a loop at every vertex K_n^ℓ for $n \geq 1$.

Proof. The proof of part (i) follows similarly to the proof in *SiLlStGraphs* of Lemma 2.1.6.

Part (ii): Let X be a set in *Sets* and define $C(X)$ as the complete graph with a loop at every vertex with the vertex set $V(C(X)) = X$. Let G be a graph in *SiStGraphs* with set function $g : |G|_V \to X$. We must show that there is a unique strict graph homomorphism $\overline{g} : G \to C(X)$ such that $g = c \circ |\overline{g}|_V$ for some set function $c : |C(X)|_V \to X$ Note that $|C(X)|_V = V(C(X)) = X$. Hence we define c as 1_X.

For $g = 1_X \circ |\overline{g}|_V$ to hold, $\overline{g}_V = g$ is uniquely determined. Then let e be an edge of G incident to vertices $x, y \in V(G)$ where x and y are not necessarily distinct. Then since strict graph homomorphisms must send edges to edges and preserve incidence, for $\overline{g}$ to be a strict graph homomorphism, $\overline{g}(e)$ must map to the edge e' of $C(X)$ incident to vertices $g(x)$ and $g(y)$. By the definition of $C(X)$ such an edge e' exists. Hence $\overline{g}$ exists and is uniquely determined by g. □

We now investigate the projective and injective objects. The addition of loops to the graphs in our category allows it to have injective objects, and not only are there injective objects, there are enough injective objects.

Theorem 2.4.3. *In SiStGraphs*

(i) the projective objects are precisely the free objects, and there are enough projective objects.

(ii) the injective objects are precisely the cofree objects, and there are enough injective objects.

Proof. Since we proved Lemma 2.3.1. for *SiStGraphs*, the proof for part (i) follows similarly to the proof in Theorem 2.2.3. of the *SiLlStGraphs* equivalent.

Part (ii): We first show that if $f : G \to H$ is a monomorphism, then $f_V : V(G) \to V(H)$ is an injective function. If not, then there are two distinct vertices $u, v \in V(G)$ such that $f(u) = f(v)$. Now consider the two morphisms $g, h : K_1 \to G$ where $g(x) = u$ and $h(x) = v$ for x the vertex of K_1. Clearly $g \neq h$, but since $f(u) = f(v)$, $f(g(x)) = f(h(x))$. Hence $f \circ g = f \circ h$ contradicting that f is a monomorphism.

Let X be a nonempty set in *Sets*, and let G, H be graphs in *SiStGraphs* with a morphism

$f : G \to C(X)$ and a monomorphism $g : G \to H$. We must show there is a morphism $\overline{f} : H \to C(X)$ such that $f = \overline{f} \circ g$.

Since g is a monomorphism, $g_V : V(G) \to V(H)$ is an injection. Then for all $v \in image(g_V)$ there is a unique $v' \in V(G)$ such that $g_V(v') = v$. Since X is non-empty, there is an element $x \in X$. We define a function $\overline{f}_V : V(H) \to X$ by $\overline{f}_V(v) = f(v')$ if $v \in image(g_V)$ and $\overline{f}_V(v) = x$ otherwise. Then since $C(X)$ is a cofree object, there is a unique morphism $\overline{f} : H \to C(X)$ such that for $|\overline{f}|_V : V(H) \to |C(X)|_V = X$, $\overline{f}_V = 1_X \circ |\overline{f}|_V = |\overline{f}|_V$. Since $\overline{f}$ is unique, $|\overline{f}|_V = \overline{f}_V$. Then for all $v \in V(G)$, $\overline{f}(g(v)) = f(v)$.

Now let $e \in E(G) \backslash image(\iota_G)$ with $\partial_G(e) = (u\_v)$. Then since morphisms preserve incidence, $\partial_{C(X)}(f(e)) = (f(u)\_f(v)) = (\overline{f}(g(u))\_\overline{f}(g(v))) = \partial_{C(X)}(\overline{f}(g(e)))$. Since morphisms must map edges to edges and there is a most one edge incident to any two vertices (not necessarily distinct), $f(e) = \overline{f}(g(e))$. Hence $f = \overline{f} \circ g$ and $C(X)$ is an injective object.

Now let G be a graph in *SiStGraphs* that is not a cofree object. Assume it is an injective object of *SiStGraphs*. Then there are vertices $u, v \in V(G)$ (not necessarily distinct) such that there is no edge $e \in E(G) \backslash image(\iota_G)$ with $\partial_G(e) = (u\_v)$.

Then consider K_2^c with morphism $f : K_2^c \to G$ defined by $f(a) = u$ and $f(b) = v$, for a and b the two vertices of K_2^c, and $i : K_2^c \to K_2$ the inclusion morphism. Since the inclusion morphism is a monomorphism, there is a morphism $\overline{f} : K_2 \to G$ such that $\overline{f} \circ i = f$. Then $\overline{f}(i(a)) = \overline{f}(a) = u$ and $\overline{f}(i(b)) = \overline{f}(b) = v$. Since morphisms preserve incidence, $\partial_G(\overline{f}(e)) = (\overline{f}(a)\_\overline{f}(b)) = (u\_v)$. Then since edges must be sent to edges and there is at most one edge between any two vertices, there is an edge e' such that $\partial_G(e') = (u\_v)$, a contradiction. Hence G is not an injective object.

To show there are enough injective objects we must show that for any graph G in *SiStGraphs*, there is an injective object H with a monomorphism $f : G \to H$. If G is not the initial object, $C(V(G))$ is an injective object and $i : G \to C(V(G))$, the inclusion morphism, is a monomorphism. If $G = \emptyset$ then $\emptyset \hookrightarrow K_1^\ell$ suffices. Hence there are enough injective objects in *SiStGraphs*. $\square$

We end this section by investigating the generators and cogenerators of $\mathcal{SiStGraphs}$.

Theorem 2.4.4. *In $\mathcal{SiStGraphs}$*

(i): the empty edge graphs, K_n^c, are precisely the generators (for $n \geq 1$).

(ii): K_2^ℓ the complete graph on two vertices with a loop at every vertex is a cogenerator, and a cogenerators of $\mathcal{SiStGraphs}$ are precisely the graphs containing a subgraph isomorphic to K_2^ℓ.

Proof. The proof of part (i) follows similarly to the proof of Theorem 2.2.4. for the $\mathcal{SiLlStGraphs}$ equivalent.

Part (ii): Let K_2^ℓ have vertices u and v with edge e incident to u and v and loops ℓ_u and ℓ_v on u and v respectively. Let X and Y be graphs in $\mathcal{SiStGraphs}$ with morphisms $f, g : X \to Y$ such that $f \neq g$. Since there is at most one loop at a vertex and at most one edge between any two vertices, there is a vertex $x \in V(X)$ such that $f(x) \neq g(x)$. Define a map $h : Y \to K_2^\ell$ by $h(f(x)) = u$ and $h(y) = v$ for all vertices $y \in V(Y)\backslash\{f(x)\}$, and for $a \in E(Y)\backslash image(\iota_Y)$, $h(a) = \ell_v$ if $\partial_Y(a) = (y_1\_y_2)$ for $y_1, y_2 \in V(Y)\backslash\{f(x)\}$, $h(a) = \ell_u$ if $\partial_Y(a) = (f(x)\_f(x))$, and $h(a) = e$ if $\partial_Y(a) = (f(x)\_y)$ for $y \in V(Y)\backslash\{f(x)\}$.

We now must show h is a strict graph homomorphism. Let $a \in E(Y)\backslash image(\iota_y)$. If $h(a) = \ell_v$ then $\partial_{K_2^\ell}(h(a)) = (v\_v) = (h(y_1)\_h(y_2))$ for some $y_1, y_2 \in V(Y)\backslash\{f(x)\}$. If $h(a) = \ell_u$ then $\partial_{K_2^\ell}(a) = (u\_u) = (h(f(x))\_h(f(x)))$. If $h(a) = e$ then $\partial_{K_2^\ell}(a) = (u\_v) = (h(f(x))\_h(y))$ for some $y \in V(Y)\backslash\{f(x)\}$. Hence h preserves incidence, and since h sends edges to edges, h is a strict graph homomorphism.

Since $f(x) \neq g(x)$, $h(f(x)) = u$ and $h(g(x)) = v$. Hence $h \circ f \neq h \circ g$, and K_2^ℓ is a cogenerator of $\mathcal{SiStGraphs}$.

If G is a graph in $\mathcal{SiStGraphs}$ that contains a subgraph isomorphic to K_2^ℓ then clearly $(i \circ h) \circ f \neq (i \circ h) \circ g$, where i is the inclusion morphism (over the isomorphism) $i : K_2^\ell \to G$. Hence G is a cogenerator of $\mathcal{SiStGraphs}$.

We now show that any cogenerator C of $\mathcal{SiStGraphs}$ contains a subgraph isomorphic to K_2^ℓ. Suppose C does not, then no two vertices of G with loops are incident to the same edge. Consider the two morphisms $id, tw : K_2^\ell \to K_2^\ell$ where id is the identity morphism and tw is the morphism defined by $tw(u) = v$, $tw(v) = u$, $tw(e) = e$, $tw(\ell_u) = \ell_v$, and $tw(\ell_v) = \ell_u$. Since C is a cogenerator, there is a morphism $h : K_2^\ell \to C$ such that $h \circ id \neq h \circ tw$.

Let $h(u) = u'$ for some $u' \in C$ and $h(v) = v'$ for some $v' \in C$. If $u' = v'$, then since edges must be sent to edges and incidence is preserved, $\partial_C(h(id(\ell_v))) = (h(v)\_h(v)) = (v'\_v') = (u'\_u') = \partial_C(h(tw(\ell_v)))$. Since there is at most one loop at a vertex, then $h(id(\ell_v)) = h(tw(\ell_v))$. Similarly $h(id(\ell_u)) = h(tw(\ell_u))$ and $h(id(e)) = h(tw(e))$. Hence $h \circ id = h \circ tw$, a contradiction. Thus $u' \neq v'$.

Since morphisms must send edges to edges, $\partial_C(h(\ell_u)) = (u'\_u')$, and $\partial_C(h(\ell_v)) = (v'\_v')$, u' has a loop $\ell_{u'}$ and v' has a loop $\ell_{v'}$. Now consider $h(e)$. Since $\partial_C(h(e)) = (h(u)\_h(v)) = (u'\_v')$, u' and v' are two vertices with loops adjacent to the same edge, a contradiction. Hence C must contain a subgraph isomorphic to K_2^ℓ. $\square$

In the last section, we showed $\mathcal{SiStGraphs}$ is a "near" topos, as it only lacks a subobject classifier. In this section, we show that $\mathcal{SiStGraphs}$ also contains a variety of useful categorial constructions, some of which will be crucial in defining graph-like objects in an abstract category.

2.5 $\mathcal{SiLlStGraphs}$ is a Topos Impoverished Category

We conclude this chapter by comparing $\mathcal{SiLlStGraphs}$ to $\mathcal{SiStGraphs}$. It becomes apparent that loops are needed for a graph category to have structure while using strict graph homomorphisms as the only morphisms. By the proof of Theorem 2.2.4. part (ii), since there are no loops in $\mathcal{SiLlStGraphs}$, there is no graph which all graphs admit a morphism to. This causes a lack of a terminal object, quotient graphs, coequalizers, cofree objects, injective objects, and cogenerators.

When loops are allowed, we find there is much more structure. $\mathcal{SiStGraphs}$ has finite limits, finite colimits, and exponentiation, each of which is lacking in $\mathcal{SiLlStGraphs}$. Furthermore, *SiStGraphs* has cofree objects, injective objects (and enough of them), and cogenerators.

The reason for this lack of structure in *SiLlStGraphs* is because if loops are not allowed in a graph category with only strict graph homomorphisms as morphisms, there is no morphism that allows two vertices to be identified. While it is evident that this prevents quotient graphs from being inside the category, we also have shown it prevents the existence of many other categorial structures. We sum it up with a simple sentence. *SiLlStGraphs* is an impoverished category (mostly because of the strictness of morphisms).

We conclude this chapter by noting that both categories lack a subobject classifier, which was the third requirement for a topos.

Chapter 3

Investigation of Graph-like Objects in an Abstract Category

3.1 Abstract Categorial Definitions of Graph-like Objects

The results in this chapter are new results. In this chapter we investigate elementary graph theory objects such as a vertex, an edge, and a loop. We will view the objects in $\mathcal{Grphs}$, and then give categorial definitions for each of these as objects in a category. We will then prove our definitions are the correct objects in our categories of graphs and then look at instances of these definitions in the Category of Sets and Functions $\mathcal{Sets}$ and the Category of Abelian Groups, $\mathcal{Ab}$.

We want a way to find a vertex in an abstract category. In $\mathcal{Grphs}$, as well as $\mathcal{SiStGraphs}$ and $\mathcal{SiLlStGraphs}$, K_1 has a special role. Any graph homomorphism $f : K_1 \to G$ for a graph G will map the vertex of K_1 to a single vertex, v, of G. Furthermore, such a graph homomorphism will be the only morphism from K_1 to G that has v as its image. So if we label the morphisms by the vertices of G that they map to, we can recover $V(G)$ as $hom(K_1, G)$. This leads us to define K_1 as the vertex object in our categories of graphs.

Definition 3.1.1. *K_1 is the* vertex object *in $\mathcal{Grphs}$, $\mathcal{SiStGraphs}$, and $\mathcal{SiLlStGraphs}$*

To achieve a categorial definition, we must view the special categorial properties of K_1. K_1 is a free object in $\mathcal{SiLlStGraphs}$ and $\mathcal{SiStGraphs}$, but defining K_1 as a free object limits it to only being defined in categories that are concrete. K_1 is also a generator in $\mathcal{SiLlStGraphs}$ and $\mathcal{SiStGraphs}$, but it turns out K_1 is not a generator of $\mathcal{Grphs}$. We see this by considering two morphisms $id, g : K_1^\ell \to K_1^\ell$ where id is the identity morphism and g is the morphism that maps the loop of K_1^ℓ to the vertex of K_1^ℓ. Since K_1 only has the inclusion morphism to K_1^ℓ and that morphism cannot differentiate between id and g, K_1 is not a generator. However, as we will prove in section 2 of this chapter, K_1 is still projective in $\mathcal{Grphs}$.

In order to define K_1 as a projective object, since there are an infinite number of projective objects in the graph categories, we need another way of describing K_1. Another property that defines K_1 is that it is a "very small" graph. One way to describe "very small" graphs is to define a minimum object in a collection (or class) of objects.

Definition 3.1.2. *Given a collection X of objects of a category $\mathcal{C}$. The* minimum object *of X, is the object M such that M admits a monomorphism to every other object in X, and if any other object in X admits a monomorphism to M then it is isomorphic to M.*

It follows immediately from the definition that a minimum object of a collection is unique up to isomorphism. Since the empty graph is trivially a projective object and it admits the inclusion monomorphism to K_1, we must first remove it from the collection of projective objects under consideration. We are now ready to define the vertex object.

Definition 3.1.3. *The* vertex object*, V_O, of a category $\mathcal{C}$ is the minimum non-initial projective object.*

Since the vertex object is a minimum object in the collection of non-initial projective objects, it is unique up to isomorphism. We can now give a categorial definition of a vertex of an object.

Definition 3.1.4. *Given a category $\mathcal{C}$ with a vertex object V_O. A* vertex of an object *A in $\mathcal{C}$ is a morphism $v : V_O \to A$.*

Finding an edge of a graph is more complex. So we consider the loop. The obvious choice to consider is K_1^ℓ. We will show in section 2 of this chapter that a morphism in $\mathcal{Grphs}$ that sends the loop of K_1^ℓ to a vertex is not a monomorphism, and hence a monomorphism from K_1^ℓ into a graph will identify a loop in each category of graphs. In $\mathcal{SiStGraphs}$, since the morphisms are strict, K_2 admits a single morphism to K_1^ℓ, but in $\mathcal{Grphs}$, K_2 admits two morphisms to K_1^ℓ. So we define a loop separately.

Definition 3.1.5. *K_1^ℓ is the* loop object *in $\mathcal{Grphs}$, $\mathcal{SiStGraphs}$, and $\mathcal{SiLlStGraphs}$*

To give a categorial definition, we first notice that in the categories of graphs, there is a single morphism from $K_1 \to K_1^\ell$ which is clearly not an isomorphism. However, in $\mathcal{Grphs}$, K_1 admits such a morphism to an infinite number of one vertex graphs because multiple loops are allowed. So we will again use the idea of a minimum object. We will also require that the morphism K_1 admits to K_1^ℓ to be a monomorphism. This is trivially the case in the categories of graphs, but this will allow the definition to exclude the zero object in abelian categories, an object which is both the initial object and terminal object.

Definition 3.1.6. *Given a category $\mathcal{C}$ with a vertex object V_O, the* loop object*, L_O, is the minimum object for which V_O admits a single monomorphism, $v_\ell : V_O \to L_O$, and $V_O \ncong L_O$.*

We can now give the categorial definition of a loop of an object.

Definition 3.1.7. *Give a category $\mathcal{C}$ with a loop object L_O, a* loop of an object *A is a monomorphism $\ell : L_O \to A$, and the* vertex incident to the loop *is the monomorphism $\ell \circ v_\ell$.*

Now that we have the categorial definition of a loop, we can define a loopless object.

Definition 3.1.8. *In a category $\mathcal{C}$ an object A is* loopless *if when there is a loop object L_O there is no monomorphism $\ell : L_O \to A$.*

If a category does not have a loop object, then all the objects of that category are vacuously loopless. We now consider a non-loop edge. The obvious object to consider in the categories of graphs is K_2, for as a graph K_2 has two vertices, u and v, connected by a single edge, e. To view how it can identify an edge of a graph, consider a graph G in $\mathcal{G}rphs$ with a non-loop edge a incident to vertices a_1 and a_2. Then K_2 admits two monomorphisms that send edge e to edge a, say $f, g : K_2 \to G$ where $f(u) = a_1$, $f(v) = a_2$, $g(u) = a_2$, and $g(u) = a_1$. With this pair of monomorphisms we can identify the edge a, and with the inclusion of K_1 into K_2 we can identify the incident vertices a_1 and a_2 of edge a.

Definition 3.1.9. *K_2 is the edge object in $\mathcal{G}rphs$, $\mathcal{S}i\mathcal{S}t\mathcal{G}raphs$, and $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$*

So to give a categorial definition of K_2, we first note that it has two vertices, and categorially that is described as $\sharp(hom(K_1, K_2)) = 2$. However in $\mathcal{G}rphs$, there are an infinite number of non-isomorphic graphs with 2 vertices because multiple edges are allowed. However, if we consider the graphs with an edge connecting the two vertices, K_2 is the minimum one. To be able to identify an edge in a set of multiple edges, we encapsulate the pair of monomorphisms to an edge by using the twist automorphism, tw, inherent to K_2.

Definition 3.1.10. *Given a category $\mathcal{C}$ with coproducts and a vertex object V_O. The edge object, E_O, is the minimum object such that $\sharp(hom(V_O, E_O)) = 2$, $E_O \ncong V_O + V_O$, E_O is loopless, and there exists an automorphism $tw : E_O \to E_O$ such that for the two distinct vertices $u, v : V_O \to E_O$, $tw \circ u = v$ and $tw \circ v = u$.*

$$\begin{tikzcd}
 & E_O \arrow[d, "tw"] \\
V_O \arrow[ur, "u"] \arrow[r, "v"] \arrow[dr, "u"'] & E_O \arrow[d, "tw"] \\
 & E_O
\end{tikzcd}$$

Figure 3.1: The Twist Automorphism, tw

We now define a non-loop edge of an object, as well as the incident vertices to the edge.

Definition 3.1.11. *Given a category $\mathcal{C}$ with an edge object E_O. A* non-loop edge of an object *A, $e = \langle e_1, e_2 \rangle$, is a pair of distinct monomorphisms $e_1, e_2 : E_O \to A$ such that $e_2 \circ tw = e_1$ and $e_1 \circ tw = e_2$.*

Definition 3.1.12. *Given an edge $e = \langle e_1, e_2 \rangle$ of an object A, $e_1, e_2 : E_O \to A$, in a category $\mathcal{C}$, the* vertices incident to e *in A are $e_1 \circ u$ and $e_1 \circ v$ where u and v are the two distinct morphisms $u, v : V_O \to E_O$.*

Now that we have a categorial definition of edges, both loop and non-loop edges, we can finally give a categorial definition of a strict morphism in a category.

Definition 3.1.13. *Given a category $\mathcal{C}$ with a vertex object V_O, a morphism $h : A \to B$ is* strict *when*

(i): if there is an edge object E_O in $\mathcal{C}$, then for all edges $e = \langle e_1, e_2 \rangle$ of A, there either exists an edge $\overline{e} = \langle \overline{e_1}, \overline{e_2} \rangle$ such that $h \circ e_1 = \overline{e_1}$ and $h \circ e_2 = \overline{e_2}$ or if there is also a loop object L_O in $\mathcal{C}$, then for all epimorphisms $\ell : E_O \to L_O$, there is a loop ℓ_0 of B such that $h \circ e_1 = \ell \circ \ell_0$ and $h \circ e_2 = \ell \circ \ell_0$.

(ii): if there is a loop object, L_O, then for all loops ℓ_0 of A, there exists a loop $\overline{\ell}_0$ of B such that $h \circ \ell_0 = \overline{\ell}_0$.

In section 2 we will show that in $\mathcal{G}rphs$ the morphism that sends the edge of E_O to the single vertex in L_O is not an epimorphism. Then we can think of an epimorphism $\ell : E_O \to L_O$ as an edge to loop morphism that allows us to identify when a non-loop edge is sent to a loop.

3.2 Graph-like Objects in the Categories of Graphs

We now check our definitions in the categories of graphs to ensure that our definitions identify the correct graph objects. We begin with the vertex object in $\mathcal{G}rphs$.

Proposition 3.2.1. *In $\mathcal{G}rphs$, K_1 is the vertex object V_O.*

Proof. We must show K_1, with vertex u, is projective, then show K_1 admits a monomorphism to every other non-initial projective object, and then show that if a non-initial projective object admits a monomorphism to K_1, then it is an isomorphism.

By Proposition 1.5.6. part (i), any epimorphism is a surjective function on the vertex set. Let G and H be graphs in $\mathcal{Grphs}$ with morphism $f : K_1 \to G$ and epimorphism $g : H \to G$. Then since g is an epimorphism, $g_V : V(H) \to V(G)$ is a surjection. Hence for every vertex v of G, there is a vertex v' of H such that $g(v') = v$.

Let $f(u) = w$. Then define $\overline{f} : K_1 \to H$ by $\overline{f}(u) = w'$. Then $g(\overline{f}(u)) = g(w') = w = f(u)$ and $g \circ \overline{f} = f$. Hence K_1 is projective.

We first note that $\emptyset$ is the initial object of $\mathcal{Grphs}$. Let P be a projective object that is not the initial object. Since P is not the initial object, it contains at least one vertex, v. Then the inclusion morphism $i : K_1 \to P$, with $i(u) = v$ is a monomorphism.

Now let Q be a non-initial projection with a monomorphism $m : Q \to K_1$. Since Q is non-initial, Q has a vertex x. Suppose P has another vertex y, $y \neq x$. Then consider the morphisms $f, g : K_1 \to Q$ defined by $f(u) = x$ and $g(u) = y$. Then $f \neq g$ and $m \circ f = m \circ g$. This contradicts m being a monomorphism. Hence Q has only one vertex.

Now suppose Q has a loop ℓ. Then consider the morphisms $id, \ulcorner x \urcorner : Q \to Q$ where id is the identity morphism and $\ulcorner x \urcorner$ is the morphism that sends all loops to the vertex x. Then $\ulcorner x \urcorner \neq id$ and $m \circ \ulcorner x \urcorner = m \circ id$. This contradicts m being a monomorphism. Hence Q is loopless and m is and isomorphism. □

We now check the loop object and loops in $\mathcal{Grphs}$.

Proposition 3.2.2. *In $\mathcal{Grphs}$*

(i): the loop object L_O is K_1^ℓ.

(ii): given a loop p of a graph G, there is a unique monomorphism $\tilde{p} : K_1^\ell \to G$ such that $\tilde{p}(\ell) = p$, and given a monomorphism $\tilde{q} : K_1^\ell \to G$, a loop q of G is identified.

Proof. Part (i): Let v_ℓ be the vertex of K_1^ℓ and ℓ be the loop. V_O only admits a single morphism to K_1^ℓ, the inclusion morphism which is a monomorphism but not an isomorphism. Let G be a graph such that V_O admits only a single monomorphism to G and it is not an isomorphism. We must show that K_1^ℓ admits a monomorphism to G.

Since all morphisms from V_O are monomorphisms in $\mathcal{Grphs}$, V_O admits only a single morphism to G and G has only a single vertex u. Since G is not isomorphic to V_O, G must contain a loop a. Then K_1^ℓ admits an inclusion monomorphism into G.

Now let H be a graph such that V_O only admits a single monomorphism to H which is not an isomorphism, with a monomorphism $m : H \to K_1^\ell$. Similarly to G, H has a single vertex v and at least one loop b. Suppose it has another loop c.

Consider $f, g : K_1^\ell \to H$ defined by $f(v_\ell) = v$, $f(\ell) = b$, $g(v_\ell) = v$, and $g(\ell) = c$. Then $f \neq g$, but $m \circ f = m \circ g$, a contradiction to m being a monomorphism. Hence H has only a single loop and m is clearly an isomorphism to K_1^ℓ.

Part (ii): Let p be a loop in G incident to a vertex u. Then the morphism $\tilde{p} : K_1^\ell \to G$ defined by $\tilde{p}(v_\ell) = u$ and $\tilde{p}(\ell) = p$ suffices. To show it is unique, consider another such monomorphism $\tilde{\tilde{p}} : K_1^\ell \to G$ such that $\tilde{\tilde{p}}(\ell) = p$. Since p is incident to u and incidence must be preserved, $\tilde{\tilde{p}}(v_\ell) = u$ and $\tilde{\tilde{p}} = \tilde{p}$.

Now let a $\tilde{q}$ be a monomorphism $\tilde{q} : K_1^\ell \to G$. Since vertices must be sent to vertices, $\tilde{q}(v_\ell) = v$ for some $v \in V(G)$. We then show that the morphism $f : K_1^\ell \to G$ defined by $f(v_\ell) = v$ and $f(\ell) = v$ is not a monomorphism. Consider the morphism $id, \ulcorner v_\ell \urcorner : K_1^\ell \to K_1^\ell$ where id is the identity morphism and $\ulcorner v_\ell \urcorner$ is the morphism defined by $\ulcorner v_\ell \urcorner(v_\ell) = v_\ell$ and $\ulcorner v_\ell \urcorner(\ell) = v_\ell$. Then $id \neq \ulcorner v_\ell \urcorner$, but $f \circ id = f \circ \ulcorner v_\ell \urcorner$. Hence f is not a monomorphism. The only other morphism K_1^ℓ admits is to map ℓ to a loop b incident to v in G. □

We now check the edge object and edge of an object definitions in $\mathcal{Grphs}$.

Proposition 3.2.3. *In* $\mathcal{Grphs}$

(i): the edge object E_O *is* K_2.

(ii): given a non-loop edge, a*, of a graph* G*, there is a unique pair of distinct monomorphisms* $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ *such that* $\tilde{a}_1(e) = \tilde{a}_2(e) = a$*,* $\tilde{a}_1 \circ tw = \tilde{a}_2$*, and* $\tilde{a}_1 = \tilde{a}_2 \circ tw$*, and given a pair of*

distinct monomorphisms $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ *such that* $\tilde{b}_1 \circ tw = \tilde{b}_2$ *and* $\tilde{b}_1 = \tilde{b}_2 \circ tw$*, a non-loop edge* b *is identified.*

Proof. Part (i): Clearly $\sharp(\hom(V_O, K_2)) = 2$, K_2 is loopless, and $K_2 \ncong V_O + V_O \cong K_2^c$. Let $u : K_1 \to K_2$ and $v : K_1 \to K_2$ be the two distinct vertices of K_2 and let e be the edge incident to both u and v. Then $tw : K_2 \to K_2$ defined by $tw(u) = v$, $tw(v) = u$ and $tw(e) = e$ suffices as the tw automorphism. Now let G be a graph such that $\sharp(\hom(V_O, G)) = 2$, G is loopless, $G \ncong K_2^c$, and for the two distinct morphisms $x, y : K_1 \to G$, there is an automorphism tw_G such that $tw_G \circ x = y$ and $tw_G \circ y = x$. We must show K_2 admits a monomorphism to G. Since G is loopless, and $G \ncong K_2^c$, the is an edge a of G incident to both x and y. Then K_2 admits an inclusion monomorphism $i : K_2 \to G$ defined by $i(u) = x$, $i(v) = y$, and $i(e) = a$.

Now let H be a graph such that $\sharp(\hom(V, H)) = 2$, H is loopless, $H \ncong K_2^c$, for the two distinct morphisms $x, y : K_1 \to H$, there is an automorphism tw_H such that $tw_H \circ x = y$ and $tw_H \circ y = x$, and H admits a morphism $m : H \to K_2$. Since H is loopless and $H \ncong K_2^c$, H has an edge a incident to both x and y. Suppose H has another edge b. Consider the morphisms $f, g : K_2 \to H$ defined by $f(u) = x$, $f(v) = y$, $f(e) = a$, $g(u) = x$, $g(v) = y$, and $g(e) = b$. Then $f \neq g$ but $m \circ f = m \circ g$ which contradicts m being a monomorphism. Hence there is only one edge in H and m is an isomorphism.

Part (ii): Let a be a non-loop edge incident to x and y in G. The pair of monomorphism $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ with $\tilde{a}_1, \tilde{a}_2 : K_2 \to G$ defined by $\tilde{a}_1(u) = x$, $\tilde{a}_1(v) = y$, $\tilde{a}_1(e) = a$, $\tilde{a}_2^(u) = y$, $\tilde{a}_2(v) = x$, and $\tilde{a}_2(e) = a$ suffices. To show it is unique, suppose there were another pair of distinct monomorphisms $\tilde{\tilde{a}} = \langle \tilde{\tilde{a}}_1, \tilde{\tilde{a}}_2 \rangle$ with $\tilde{\tilde{a}}_1, \tilde{\tilde{a}}_2 : K_2 \to G$ such that $\tilde{\tilde{a}}_1(e) = \tilde{\tilde{a}}_2(e) = a$. Then since vertices are sent to vertices and incidence is preserved, $\tilde{\tilde{a}}_1(u)$ is sent to either x or y. Without loss of generality, $\tilde{\tilde{a}}_1(u) = x$. Then $\tilde{\tilde{a}}_1(v) = y$ and since $\tilde{\tilde{a}}_1$ and $\tilde{\tilde{a}}_2$ are distinct, $\tilde{\tilde{a}}_2(u) = y$ and $\tilde{\tilde{a}}_2(v) = x$. Hence $\tilde{a}_1 = \tilde{\tilde{a}}_1$ and $\tilde{a}_2 = \tilde{\tilde{a}}_2$.

Let $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ be a pair of distinct monomorphisms $\tilde{b}_1, \tilde{b}_2 : K_2 \to G$ such that $\tilde{b}_1 \circ tw = \tilde{b}_2$ and $\tilde{b}_1 = \tilde{b}_2 \circ tw$. Since vertices must be mapped to vertices, there are vertices x and y in G such that $\tilde{b}_1(u) = x$ and $\tilde{b}_1(v) = y$. Since $\tilde{b}_1 \circ tw = \tilde{b}_2$, $\tilde{b}_2(u) = y$ and $\tilde{b}_2(v) = x$.

Suppose $x = y$. Then since $\tilde{b}_1$ and $\tilde{b}_2$ are distinct and $\tilde{b}_1(u) = \tilde{b}_1(v) = \tilde{b}_2(u) = \tilde{b}_2(v)$, $\tilde{b}_1(e) = a$ and $\tilde{b}_2(e) = b$ for two distinct loops a and b in G. However, $\tilde{b}_1(tw(e)) = \tilde{b}_1(e) \neq \tilde{b}_2(e)$, a contradiction. Hence x and y are distinct. Then since incidence must be preserved, there is a non-loop edge b in G such that $\tilde{b}_1(e) = b$. Then $b = \tilde{b}_1(e) = \tilde{b}_1(tw(e)) = \tilde{b}_2(e)$, and b is identified. □

These propositions are what we should expect. Next we check that the strict graph homomorphisms of $\mathcal{G}rphs$ are strict morphisms in the categorial sense.

Proposition 3.2.4. *In $\mathcal{G}rphs$, $f : G \to H$ is a strict graph homomorphism if and only if it is strict morphism (in the categorial sense).*

Proof. Let $f : G \to H$ be a strict graph homomorphism. We first prove there is one epimorphism $el : K_2 \to K_1^\ell$ defined by $el(u) = el(v) = v_\ell$ and $el(e) = \ell$. By inspection, the only other morphism from K_2 to K_1^ℓ is $\ulcorner v_\ell \urcorner$ defined by $\ulcorner v_\ell \urcorner(u) = \ulcorner v_\ell \urcorner(v) = \ulcorner v_\ell \urcorner(e) = v_\ell$. Consider the morphisms $id, g : K_1^\ell \to K_1^\ell$ where id is the identity morphism and g is defined by $g(\ell) = g(v_\ell) = v_\ell$. Then $id \neq g$, but $id \circ \ulcorner v_\ell \urcorner = g \circ \ulcorner v_\ell \urcorner$. Hence $\ulcorner v_\ell \urcorner$ is not an epimorphism.

If G contains no edges, then this is a vacuously true statement. So let a be a non-loop edge of G incident to vertices x and y. Then by Proposition 3.2.3. (ii), there is a unique pair of distinct monomorphisms $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ from K_2 such that $\tilde{a}_1(e) = \tilde{a}_2(e) = a$. Since vertices must be mapped to vertices, without loss of generality $\tilde{a}_1(u) = x = \tilde{a}_2(v)$ and $\tilde{a}_1(v) = y = \tilde{a}_2(u)$. Since edges must be mapped to edges by a strict graph homomorphism, $f(a) = b$ for some edge b in H.

Suppose b is a non-loop edge incident to distinct vertices x' and y' such that $f(x) = x'$ and $f(y) = y'$. Then since b is a non-loop edge of H there exists a unique pair of distinct monomorphisms $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ from K_2 such that $\tilde{b}_1(e) = \tilde{b}_2(e) = b$. Since vertices must be mapped to vertices and incidence preserved, without loss of generality $\tilde{b}_1(u) = \tilde{b}_2(v) = x'$ and $\tilde{b}_1(v) = \tilde{b}_2(u) = y'$. Then $\tilde{b}_1 = f \circ \tilde{a}_1$ and $\tilde{b}_2 = f \circ \tilde{a}_1$ as desired.

Suppose b is a loop incident to vertex $x' = f(x) = f(y)$ in H. Then by Proposition 3.2.2. (ii) there is a unique monomorphism $\tilde{b} : K_1^\ell \to H$ such that $\tilde{b}(\ell) = b$. Since incidence is preserved, $\tilde{b}(v_\ell) = x'$. Since el is uniquely defined above, clearly $\tilde{b} \circ el = f \circ \tilde{a}_1$ and $\tilde{b} \circ el = f \circ \tilde{a}_2$, as desired.

Now let a be a loop of G incident to a vertex x. Then by Proposition 3.2.2. (ii) there is a unique monomorphism $\tilde{a} : K_1^\ell \to G$ such that $\tilde{a}(\ell) = a$. Since incidence is preserved, $\tilde{a}(v_\ell) = x$. Since edges must be sent to edges and incidence preserved, there is a loop b in H incident to a vertex x' such that $f(a) = b$ and $f(x) = x'$. Then since b is a loop of H there is a unique monomorphism $\tilde{b} : K_1^\ell \to H$ such that $\tilde{b}(\ell) = b$, $\tilde{b}(v_\ell) = x'$. Then $f \circ \tilde{a} = \tilde{b}$ as desired.

Conversely, let $f : G \to H$ by a strict morphism in $\mathcal{Grphs}$. If G contains no edges, then by Proposition 3.2.2. (ii) and Proposition 3.2.3. (ii) there is no monomorphism to G from K_2 or K_1^ℓ and the result is a vacuously true statement. So let a be an edge of G.

If a is a non-loop edge, then by Proposition 2.3.3. (ii) there is a unique pair of distinct monomorphisms $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ from K_2 such that $\tilde{a}_1(e) = \tilde{a}_2(e) = a$. Since f is a strict morphism and a is a non-loop edge, there is either a pair of distinct monomorphisms $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ from K_2 such that $f \circ \tilde{a} = \tilde{b}_1$ and $f \circ \tilde{a}_2 = \tilde{b}_2$ or a monomorphism $\tilde{\tilde{b}}$ from K_1^ℓ such that $f \circ \tilde{a}_1 = \tilde{\tilde{b}}$ and $f \circ \tilde{a}_2 = \tilde{\tilde{b}}$.

In the first case, by Proposition 3.2.3. (ii) $\tilde{b}$ identifies a non-loop edge b in H such that $f(\tilde{a}(e)) = f(a) = \tilde{b}_1(a) = b$ and f sent an edge to an edge. In the second case by Proposition 3.2.2. (ii) $\tilde{\tilde{b}}$ identifies a loop b in H such that $f(\tilde{a}(e)) = \tilde{\tilde{b}}(\ell) = b$, and f sent an edge to a loop. Hence f sends non-loop edges to edges.

Now let a be a loop of G. By Proposition 3.2.2. (ii) there is a unique monomorphism $\tilde{a}$ from K_1^ℓ such that $\tilde{a}(\ell) = a$. Since f is a strict morphism, there is a monomorphism $\tilde{b} : K_1^\ell \to H$ such that $f \circ \tilde{a} = \tilde{b}$. By Proposition 3.2.2. (ii) $\tilde{b}$ identifies a loop b in H such that $\tilde{b}(\ell) = b$. Hence $f(\tilde{a}(\ell)) = f(a) = \tilde{b}(\ell) = b$ and f sends loops to loops. Hence f is a strict graph homomorphism. $\square$

We now check that these definitions of categorial graph-like objects are the correct objects in $\mathcal{SiStGraphs}$.

Proposition 3.2.5. *In $\mathcal{SiStGraphs}$*

(i) the vertex object V_O is K_1.

(ii) the loop object L_O is K_1^ℓ.

(iii) given a loop p of a graph G, there is a unique monomorphism $\tilde{p} : K_1^\ell \to G$ such that $\tilde{p}(\ell) = p$, and given a monomorphism $\tilde{q} : K_1^\ell \to G$, a loop q of G is identified.

(iv) the edge object E_O is K_2.

(v) given a non-loop edge, a, of a graph G, there is a unique pair of distinct monomorphisms $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ such that $\tilde{a}_1(e) = \tilde{a}_2(e) = a$, and $\tilde{a}_1 \circ tw = \tilde{a}_2$ and $\tilde{a}_1 = \tilde{a}_2 \circ tw$, and given a pair of distinct monomorphisms $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ such that $\tilde{b}_1 \circ tw = \tilde{b}_2$ and $\tilde{b}_1 = \tilde{b}_2 \circ tw$, a non-loop edge b is identified.

Proof. Part (i): By Theorem 2.5.3. the non-initial projective objects of $\mathcal{SiStGraphs}$ are K_n^c for $n \geq 1$. Clearly K_1 admits a monomorphism to every K_n^c with $n \geq 1$. The rest of the proof follows similarly to the proof of Proposition 3.2.1.

Part (ii): Since K_1 admits more than one monomorphism to all graphs with two or more vertices, and since K_1 does not admit a morphism to $\emptyset$, K_1 admits only a single morphism to graphs with 1 vertex. There are only two non-isomorphic graphs with 1 vertex in $\mathcal{SiStGraphs}$, K_1 and K_1^ℓ. Hence K_1^ℓ is the only object that K_1 admits a single monomorphism to that is not an isomorphism. Thus K_1^ℓ is trivially the minimum such object.

Part (iii): Since $\mathcal{SiStGraphs}$ is a subcategory of $\mathcal{Grphs}$, the proof follows from Proposition 3.2.2. (ii).

Part (iv): Since K_2 is the only loopless graph with $\sharp(hom(K_1, K_2)) = 2$ that is not isomorphic to $K_1 + K_1 = K_2^c$, it is trivially the minimum such object. The proof K_2 has the tw automorphism follows similar to proof in Proposition 3.2.3. (i).

Part (v): Since $\mathcal{SiStGraphs}$ is a subcategory of $\mathcal{Grphs}$, the proof follows from Proposition 3.2.3. (iii).

□

Corollary 3.2.6. *The morphisms of $\mathcal{SiStGraphs}$ are strict morphisms (in the categorial sense).*

Proof. Since the vertex object, the loop object, and the edge object of $\mathcal{S}i\mathcal{S}t\mathcal{G}raphs$ are the same as those in $\mathcal{G}rphs$, and since all morphism are strict graph homomorphisms in $\mathcal{S}i\mathcal{S}t\mathcal{G}raphs$, the result follows from Proposition 3.2.4. □

We now check our definitions are the correct objects in $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$.

Proposition 3.2.7. *In $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$*

(i) the vertex object V_O is K_1.

(ii) there is no loop object L_O.

(iii) the edge object E_O is K_2.

(iv) given a non-loop edge, a, of a graph G, there is a unique pair of distinct monomorphisms $\tilde{a} = \langle \tilde{a}_1, \tilde{a}_2 \rangle$ such that $\tilde{a}_1(e) = \tilde{a}_2(e) = a$, and $\tilde{a}_1 \circ tw = \tilde{a}_2$ and $\tilde{a}_1 = \tilde{a}_2 \circ tw$, and given a pair of distinct monomorphisms $\tilde{b} = \langle \tilde{b}_1, \tilde{b}_2 \rangle$ such that $\tilde{b}_1 \circ tw = \tilde{b}_2$ and $\tilde{b}_1 = \tilde{b}_2 \circ tw$, a non-loop edge b is identified.

Proof. The proofs of parts (i), (iii), and (iv) follow similarly to those in Proposition 3.2.5.

Part (ii): The only graph in $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$ that K_1 admits a single morphism to is K_1. Hence no loop object exists. □

Corollary 3.2.8. *The morphisms of $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$ are strict morphisms (in the categorial sense).*

Proof. Since the morphisms of $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$ are strict graph homomorphisms, and the vertex object and edge object of $\mathcal{S}i\mathcal{L}l\mathcal{S}t\mathcal{G}raphs$ are the same as those in $\mathcal{G}rphs$, the result follows from Proposition 3.2.4. □

We then see that in each of our defined categories of graphs, the categorial definitions of the graph-like objects correctly identify graph objects.

3.3 Graph-like Objects in Other Categories

We now investigate our categorial definitions of graph-like objects in $\mathcal{Sets}$ and $\mathcal{Ab}$. We will start with $\mathcal{Sets}$. First note that epimorphisms are surjections and monomorphisms are injections in $\mathcal{Sets}$ (for proof, see page 19 [12]). We will require the following lemma.

Lemma 3.3.1. *Every object is projective in $\mathcal{Sets}$.*

Proof. Since $\emptyset$ is the initial object of sets, $\emptyset$ is projective. Let X be a non-empty set, we will show X is projective. Let Y and Z be sets with a function $f : X \to Y$ and a surjection $g : Z \to Y$.

Since g is a surjection, for every $y \in Y$, there is a $y' \in Z$ such that $g(y') = y$ (use the Axiom of Choice). Then define a function $\overline{f} : X \to Z$ by $\overline{f}(x) = y'$ for $f(x) = y$. Then $f = g \circ \overline{f}$. □

We can now view our graph-like objects in $\mathcal{Sets}$.

Proposition 3.3.2. *In $\mathcal{Sets}$*

(i) $\mathbf{1}$*, the one element set, is the vertex object.*

(ii) there is no loop object.

(iii) there is no edge object.

(iv) all morphisms are strict.

Proof. Part (i): By Lemma 3.3.1., we must show $\mathbf{1}$ is the minimum non-initial set. Clearly there is an injection from $\mathbf{1}$ into any other non-empty set. Now let Y be a non-empty set such that $f : Y \to \mathbf{1}$ is a injection. Suppose Y has two or more elements. Then let $x, y \in Y$ be distinct. Since $\mathbf{1}$ has only one element, $f(x) = f(y)$. Sine f is an injection, $x = y$, a contradiction. Hence Y is a one element set and $\mathbf{1} \cong Y$.

Part (ii): Since $\sharp(\hom(\mathbf{1}, X)) = \sharp(X)$ and every function from $\mathbf{1}$ is an injection, no loop object exists.

Part (iii): Since coproducts are disjoint union in $\mathcal{Sets}$, [12], $\mathbf{1}+\mathbf{1}$ is a two element set. Since all two element sets are isomorphic in $\mathcal{Sets}$, and $\sharp(\hom(\mathbf{1},X)) = \sharp(X)$, no edge object exists.

Part (iv): This result follows vacuously since there is no loop object nor edge object. □

We see then, for a set X, the "graph" of X is just an empty edge graph with the vertex set X. We now move onto $\mathcal{Ab}$. We first note a widely known result. An abelian group is projective if and only if it is free (for a proof see [10] pg. 48). Furthermore free abelian groups are isomorphic to a direct product of the integers $\mathbb{Z}$ under addition, and given an element, x, of a free abelian group, x can be written uniquely as a linear combination of the generators (with coefficients from $\mathbb{Z}$). We also note that in $\mathcal{Ab}$, monomorphisms are injections and epimorphisms are surjections (for a proof see [12] pg. 24).

Proposition 3.3.3. *In $\mathcal{Ab}$*

(i) the group ($\mathbb{Z}$,+) is the vertex object.

(ii) there is no loop object.

(iii) there is no edge object.

(iv) all morphisms are strict.

Proof. Part (i): We first note that the initial object is the zero object in $\mathcal{Ab}$, the trivial group. Let F be a non-initial free abelian group. Then there is a generator $x \in F$, and $f : \mathbb{Z} \to F$ defined by $f(1) = x$ is an injection.

Now let F be a non-initial free abelian group with an injection $m : F \to \mathbb{Z}$. Suppose F has two distinct generators x and y. Then since m is an injection $m(x) \neq m(y)$. Let $m(x) = k$ and $m(y) = n$ for some integers $k \neq n$. Then $m(n * x) = n * m(x) = k * m(y) = m(k * y)$. Since m is an injection, $n * x = k * y$ and the element $n * x$ is not uniquely represented, a contradiction to F being free. Hence $m(x) = m(y)$ and since m is an injection $x = y$. Then F is generated by one element and $F \cong \mathbb{Z}$.

Part (ii): Suppose G is an abelian group for which $\mathbb{Z}$ admits a only single injection $f : \mathbb{Z} \to G$ and $G \ncong \mathbb{Z}$. Supposed $f(1) = x$ for some $x \in G$. Then $\langle x \rangle$, the subgroup of G generated by

x, is isomorphic to $\mathbb{Z}$.

Now consider $g : \mathbb{Z} \to \langle x \rangle$ defined by $g(1) = 2 * x$. Let $a, b \in \mathbb{Z}$ such that $g(a) = g(b)$, then $2a * x = 2b * x$. Since $\langle x \rangle \cong \mathbb{Z}$, x is not a torsion element and $2a = 2b$. Thus $a = b$, and g is an injection. Since the inclusion morphism $i : \langle x \rangle \hookrightarrow G$ is an injection, $i \circ g : \mathbb{Z} \to G$ is an injection. Hence $\mathbb{Z}$ admits two injections to G, a contradiction. Hence no loop object exists.

Part (iii): Let G be an abelian group for which $\mathbb{Z}$ admits only two distinct group homomorphisms $f, g : \mathbb{Z} \to G$. Since $\ulcorner e \urcorner : \mathbb{Z} \to G$, the morphism which maps all elements to the identity, e, of G, is always a group homomorphism, without loss of generality let $f = \ulcorner e \urcorner$. Since $g \neq f$ there is an element $x \in G$ such that $g(1) = x$. Since automorphisms of G are a group homomorphisms, and group homomorphisms must send e to e, there is no automorphism tw, such that $tw \circ f = g$. Hence there is no edge object.

Part (iv): As with $\mathcal{Sets}$, this follows vacuously since there is no loop object or edge object. $\square$

We end this chapter with an interesting note about the vertex object of $\mathcal{Ab}$. Since $\mathbb{Z}$ has a group homomorphism $x : \mathbb{Z} \to \mathbb{Z}$ such that $x(1) = x$ for every element $x \in \mathbb{Z}$, i.e. each x is a "vertex" of the object $\mathbb{Z}$ in the category $\mathcal{Ab}$; and there are no "edges" of the object $\mathbb{Z}$ in the category $\mathcal{Ab}$. This gives the following "graph" of $\mathbb{Z}$.

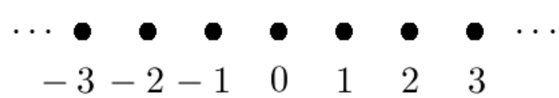


Figure 3.2: The "Graph" of ($\mathbb{Z}$,+)

Chapter 4

Reflective and Coreflective Subcategories of Graph Categories

4.1 The Theory of Reflective and Coreflective Subcategories

We finally look at the relationships of our Categories of Graphs. We first develop the theory of reflective and co-reflective subcategories following [12] pg. 90 and [9] pg. 275. We first define a reflective subcategory.

Definition 4.1.1. *A subcategory $\mathcal{A}$ of $\mathcal{B}$ is a* reflective subcategory *if the inclusion functor $I : \mathcal{A} \hookrightarrow \mathcal{B}$ has a left adjoint $R : \mathcal{B} \rightsquigarrow \mathcal{A}$, $R \dashv I$. We call the functor R a reflector.*

Dually we defined a coreflective subcategory.

Definition 4.1.2. *A subcategory $\mathcal{A}$ of $\mathcal{B}$ is a* co-reflective subcategory *if the inclusion functor $I : \mathcal{A} \hookrightarrow \mathcal{B}$ has a right adjoint $C : \mathcal{B} \rightsquigarrow \mathcal{A}$, $I \dashv C$. We call the functor C a co-reflector.*

Our first theorem states that adjoints imply continuity, i.e. colimits commute with left adjoints and limits commute with right adjoints. Mac Lane calls this "the most useful property of adjoints", [12, p. 114].

Theorem 4.1.3. *Let $\mathcal{A}$ and $\mathcal{B}$ be categories with functors $G : \mathcal{A} \to \mathcal{B}$ and $F : \mathcal{B} \to \mathcal{A}$ such that F is left adjoint to G, $F \dashv G$, then*

(i) F is right continuous, i.e. for $\underrightarrow{L}_D(-)$ the colimit of a diagram D, $F(\underrightarrow{L}_D(-)) = \underrightarrow{L}_D(F(-))$.

(ii) G is right continuous, i.e. for $\underleftarrow{L}_D(-)$ the limit of a diagram D, $G(\underleftarrow{L}_D(-)) = \underleftarrow{L}_D(G(-))$.

We then apply the theorem to reflective and coreflective subcategories.

Corollary 4.1.4. *(i) If $\mathcal{A}$ is a reflective subcategory of $\mathcal{B}$, then $R \circ \underrightarrow{L}_D = \underrightarrow{L}_D \circ R$ and $I \circ \underleftarrow{L}_D = \underleftarrow{L}_D \circ I$.*

(ii) If $\mathcal{A}$ is a coreflective subcategory of $\mathcal{B}$, then $C \circ \underleftarrow{L}_D = \underleftarrow{L}_D \circ C$ and $I \circ \underrightarrow{L}_D = \underrightarrow{L}_D \circ I$.

We call a category complete when it has all limits, and cocomplete when it has all colimits. Then as a consequence of Corollary 4.1.4. we get the following two theorems and corollaries (see chapter 10 of Herrlich and Strecker [9]).

Theorem 4.1.5. *If $\mathcal{A}$ is reflective in $\mathcal{B}$, then $\mathcal{A}$ is closed under limits in $\mathcal{B}$.*

Corollary 4.1.6. *Reflective subcategories of complete categories are complete.*

Theorem 4.1.7. *If $\mathcal{A}$ is reflective in $\mathcal{B}$ with reflector R, then the colimit, $\underrightarrow{L}_D$, in $\mathcal{A}$ is the reflection of the colimit in $\mathcal{B}$, $R(\underrightarrow{L}_D)$.*

Corollary 4.1.8. *Reflective subcategories of cocomplete categories are cocomplete.*

Dually we also have the following two theorems and corollaries.

Theorem 4.1.9. *If $\mathcal{A}$ is coreflective in $\mathcal{B}$, then $\mathcal{A}$ is closed under colimits in $\mathcal{B}$.*

Corollary 4.1.10. *Coreflective subcategories of cocomplete categories are cocomplete.*

Theorem 4.1.11. *If $\mathcal{A}$ is coreflective in $\mathcal{B}$ with coreflector C, then the limit, $\underleftarrow{L}_D$, in $\mathcal{A}$ is the reflection of the limit in $\mathcal{B}$, $C(\underleftarrow{L}_D)$.*

Corollary 4.1.12. *Coreflective subcategories of complete categories are complete.*

4.2 Relationships in the Categories of Graphs

The results in this section are new results. We now investigate the relationships of the Categories of Graphs we use in this paper. We have the containment $\mathcal{SiLlStGraphs} \hookrightarrow \mathcal{SiStGraphs} \hookrightarrow \mathcal{Grphs}$. Are any of these subcategories reflective or coreflective? We answer this question by first investigating $\mathcal{SiLlStGraphs} \hookrightarrow \mathcal{SiStGraphs}$.

Proposition 4.2.1. *$\mathcal{SiLlStGraphs}$ is neither a reflective nor coreflective subcategory of $\mathcal{SiStGraphs}$.*

Proof. Since the limit, the terminal object, does not exist in $\mathcal{SiLlStGraphs}$, then by the contrapositive to Theorem 4.1.5. $\mathcal{SiLlStGraphs}$ is not a reflective subcategory of $\mathcal{SiStGraphs}$. Since the colimit, the coequalizer, does not exist in $\mathcal{SiLlStGraphs}$, then by the contrapositive to Theorem 4.1.9., $\mathcal{SiLlStGraphs}$ is not a coreflective subcategory of $\mathcal{SiStGraphs}$. □

Since coequalizers exist in $\mathcal{Grphs}$ and the terminal object exists in $\mathcal{Grphs}$, the same proof yields the following proposition.

Proposition 4.2.2. *$\mathcal{SiLlStGraphs}$ is neither a reflective nor coreflective subcategory of $\mathcal{Grphs}$.*

Lastly we investigate $\mathcal{SiStGraphs} \hookrightarrow \mathcal{Grphs}$.

Proposition 4.2.3. *$\mathcal{SiStGraphs}$ is not a reflective or coreflective subcategory of $\mathcal{Grphs}$*

Proof. Suppose $\mathcal{SiStGraphs}$ is a reflective subcategory of $\mathcal{Grphs}$ with reflector R. Then by Theorem 4.1.7. the terminal object of $\mathcal{SiStGraphs}$ is a reflection of the terminal object in $\mathcal{Grphs}$. Since every graph in $\mathcal{Grphs}$ admits a unique morphism to K_1 where every vertex and edge is mapped to the single vertex, K_1 is the terminal object of $\mathcal{Grphs}$. Hence $R(K_1) = K_1^{\ell}$.

Since $R \dashv I$, where I is the inclusion functor, $\hom_{\mathcal{SiStGraphs}}(R(K_1), K_2) \cong \hom_{\mathcal{Grphs}}(K_1, I(K_2))$. Since $R(K_1) = K_1^{\ell}$, $\sharp(\hom_{\mathcal{SiStGraphs}}(R(K_1), K_2)) = 0$. Since $I(K_2) = K_2$, $\sharp(\hom_{\mathcal{Grphs}}(K_1, I(K_2))) = 2$. Then no such bijection exists, a contradiction. Hence $\mathcal{SiStGraphs}$ is not a reflective subcategory of $\mathcal{Grphs}$.

Now suppose $\mathcal{SiStGraphs}$ is a coreflective subcategory of $\mathcal{Grphs}$ with coreflector C. In $\mathcal{Grphs}$, since K_1 is the terminal object, $K_1 \times K_2 = K_2$. Then by Theorem 4.1.11. in $\mathcal{SiLlStGraphs}$, $C(K_1 \times K_2) = K_1 \times K_2 = K_2^c$. Then since $I \dashv C$, where I is the inclusion functor, $\hom_{\mathcal{Grphs}}(I(K_2), K_1 \times K_2) \cong \hom_{\mathcal{SiStGraphs}}(K_2, C(K_1 \times K_2))$. But $\sharp(\hom_{\mathcal{Grphs}}(I(K_2), K_1 \times K_2)) = 4$ and $\sharp(\hom_{\mathcal{SiStGraphs}}(K_2, C(K_1 \times K_2))) = 0$, a contradiction. Hence $\mathcal{SiStGraphs}$ is not a coreflective subcategory of $\mathcal{Grphs}$. □

We finish this chapter by looking at an overview of the categories of graphs. We form a "Hasse Diagram" using the inclusion functors of 6 categories of graphs (as well as the Category of Sets and Functions viewed as a category of edgeless graphs). In the following diagram $\mathcal{G}$ stands for $\mathcal{Grphs}$ while the modifiers $\mathcal{Si}$, $\mathcal{Ll}$, and $\mathcal{St}$ stand for the restrictions of *simple*, *loopless*, and *strict* respectively. The functors represented by dashed arrows are adjoints to the inclusion functor.

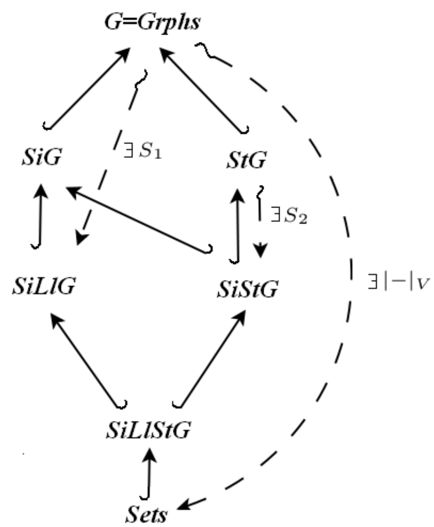


Figure 4.1: The Categories of Graphs

The functor $S_1 : \mathit{Grphs} \rightsquigarrow \mathit{SiLlGraphs}$ is the simplification functor that removes loops (i.e. collapses loops to their incident vertex) and identifies multiple edges as a single edge. S_1 is left adjoint to inclusion and, as such, *SiLlGraphs* is a reflective subcategory of *Grphs*. The functor $S_2 : \mathit{StGraphs} \rightsquigarrow \mathit{SiStGraphs}$ is the simplification functor that identifies loops as a single loop and multiple edges as a single edge. S_2 is left adjoint to inclusion and, as such, *SiStGraphs* is a reflective subcategory of *StGraphs*. We last note that if we consider *Sets* as the empty edge graphs, $|-|_V : \mathit{Grphs} \rightsquigarrow \mathit{Sets}$ is right adjoint to inclusion and, as such, *Sets* is a coreflective subcategory of *Grphs*.

Chapter 5

Conclusion

We have discovered that $\mathcal{SiLlStGraphs}$ lacks many categorial constructions, and our investigation into $\mathcal{SiStGraphs}$ gives us a glimpse as to why. If in a graph category, there is no morphism to a graph obtained by identifying vertices, many categorial constructions such as quotient graphs, coequalizers, and injective objects do not exist. However, we do find that both $\mathcal{SiLlStGraphs}$ and $\mathcal{SiStGraphs}$ do not have a subobject classifier, and therefore are not topoi.

Keeping the goal of an axiomatization of the categories of graphs in mind, we see that the categorial objects inherent in both $\mathcal{SiStGraphs}$ and $\mathcal{SiLlStGraphs}$ are necessary conditions that must be satisfied. With the categorial definitions of graph-like objects, and especially strict morphisms, we have a categorial way of differentiating between $\mathcal{Grphs}$, $\mathcal{SiStGraphs}$ and $\mathcal{SiLlStGraphs}$.

However, our list of necessary conditions is not a list of sufficient conditions. A further area of study would be to expand our list until an independent sufficient list of conditions is found.

We discovered that $\mathcal{SiStGraphs}$ and $\mathcal{SiLlStGraphs}$ are not reflective or coreflective subcategories of $\mathcal{Grphs}$. Another area of study would be to find reflective and coreflective subcategories of $\mathcal{Grphs}$ which will inherit much of the structure of $\mathcal{Grphs}$.

Appendix A

Primer of Category Theory

Here you will find the definitions for all the categorial constructions used in this paper that were not included in the main body. We follow the format of [4] .

Definition A.0.4. *An object* $\mathbf{0}$ *is* initial *in a category* $\mathcal{C}$ *if for every other object in the category, A, there is one unique morphism from* $\mathbf{0}$ *to* A.

The initial object is the colimit of the empty diagram. In $\mathcal{Sets}$ the initial object is the empty set, $\emptyset$, and in $\mathcal{Ab}$ the initial object is the trivial group.

Definition A.0.5. *An object* $\mathbf{1}$ *is* terminal *in a category* $\mathcal{C}$ *if for every other object in the category, A, there is one unique morphism from* A *to* $\mathbf{1}$.

The terminal object is the limit of the empty diagram. In $\mathcal{Sets}$ the terminal object is the one element set, and in $\mathcal{Ab}$ the terminal object is the trivial group.

Definition A.0.6. *Given a pair of morphisms* $f, g : A \to B$ *in a category* $\mathcal{C}$*, the* equalizer *is an object* Eq *with a morphism* $eq : Eq \to A$ *such that:*

(i) $f \circ eq = g \circ eq$ *and,*

(ii) whenever there is an object X and a morphism $h : X \to A$ such that $f \circ h = g \circ h$, there is a unique morphism $\overline{h} : X \to Eq$ such that $h = eq \circ h$.

The definition states that the following diagram commutes.

$$\begin{tikzcd}
X \arrow[d, dashed, "\overline{h}"'] \arrow[dr, "h"] & & \\
Eq \arrow[r, "eq"] \arrow[u, phantom, shift right=-6ex, "="] & A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B
\end{tikzcd}$$

Figure A.1: The Equalizer

The equalizer is the limit of the diagram: $\bullet \rightrightarrows \bullet$. In $\mathcal{S}ets$ the equalizer of two functions is the subset of the domain in which the functions agree along with the inclusion function, and in $\mathcal{A}b$ the equalizer of two group homomorphisms f and g is $ker(f - g)$.

Definition A.0.7. *Given a pair of morphism $f, g : A \to B$ in a category $\mathcal{C}$, the* coequalizer *is an object $Coeq$ and a morphism $coeq : B \to Coeq$ such that:*

(i) $coeq \circ f = coeq \circ g$ and,

(ii) whenever there is an object X with a morphism $h : B \to X$ such that $h \circ f = g \circ f$, there is a unique morphism $\overline{h} : Coeq \to X$ such that $\overline{h} \circ coeq = h$.

The definition states that the following diagram commutes.

$$\begin{tikzcd}
A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B \arrow[r, "coeq"] \arrow[dr, "h"'] \arrow[dr, phantom, shift left=3ex, "="] & Coeq \arrow[d, dashed, "\overline{h}"] \\
& & X
\end{tikzcd}$$

Figure A.2: The Coequalizer

The coequalizer is the colimit of the diagram: $\bullet \rightrightarrows \bullet$. In $\mathcal{S}ets$ the coequalizer of two functions is the set of congruence classes defined by equivalence relation generated by the functions, and in $\mathcal{Ab}$ the coequalizer of two group homomorphisms f and g is the factor group obtained from the codomain through $image(f - g)$.

Definition A.0.8. Products *exist in a category* $\mathcal{C}$, *if for all objects* A *and* B *in* $\mathcal{C}$, *there exists an object* $A \times B$ *with morphisms* $\pi_A : A \times B \to A$ *and* $\pi_B : A \times B \to B$ *in* $\mathcal{C}$ *such that for all objects* X *with morphisms* $f_A : X \to A$ *and* $f_B : X \to B$, *there exists a unique morphism* $\overline{f} : X \to A \times B$ *such that* $f_A = \pi_A \circ \overline{f}$ *and* $f_B = \pi_B \circ \overline{f}$.

This definition states the following diagram commutes.

$$
\begin{tikzcd}
X \arrow[d, dashed, "\exists!\overline{f}"'] \arrow[dr, bend left, "f_A"] \arrow[ddr, bend left, "f_B"] & \\
A \times B \arrow[r, "\pi_A"] \arrow[dr, "\pi_B"'] & A \\
& B
\end{tikzcd}
$$

Figure A.3: The Product

The product is the limit of the diagram $\bullet \quad \bullet$. In $\mathcal{Sets}$ the product of two sets is the cartesian product, and in $\mathcal{Ab}$ the product of two abelian groups is the direct product.

Definition A.0.9. Coproducts *exist in a category* $\mathcal{C}$, *if for all objects* A *and* B *in* $\mathcal{C}$, *there exists an object* $A + B$ *with morphisms* $i_A : A \to A + B$ *and* $i_B : B \to A + B$ *such that for all objects* X *with morphisms* $g_A : A \to X$ *and* $g_B : B \to X$, *there exists a unique morphism* $\overline{g} : A + B \to X$.

This definition states the following diagram commutes.

$$\begin{tikzcd}
A \arrow[dr, "i_A"] \arrow[ddr, "g_A"'] & \\
B \arrow[r, "i_B"] \arrow[dr, "g_B"'] & A+B \arrow[d, dashed, "\exists!\overline{g}"] \\
& X
\end{tikzcd}$$

Figure A.4: The Coproduct

The coproduct is the colimit of the diagram $\bullet \quad \bullet$. In $\mathcal{S}ets$ the coproduct of two sets is the disjoint union, and in $\mathcal{A}b$ the coproduct of two abelian groups is the direct sum. In $\mathcal{A}b$ the product and coproduct of two finite abelian groups are naturally isomorphic.

Definition A.0.10. *A category $\mathcal{C}$ has* exponentiation with evaluation *if it has a product for any two objects and given two objects A and B, there is an object B^A in $\mathcal{C}$ with a morphism $ev : B^A \times A \to B$ such that for every other object X with morphism $g : X \times A \to B$, then there is a unique morphism $\overline{g} : X \to B^A$ such that $ev \circ (\overline{g} \times 1_A) = g$. The assignment of $\overline{g}$ to g establishes a bijection $hom_{\mathcal{C}}(X \times A, B) \cong hom_{\mathcal{C}}(X, B^A)$ in $\mathcal{S}ets$.*

The definition states that the following diagram commutes.

$$\begin{tikzcd}
X \times A \arrow[d, dashed, "\overline{g}\times 1_A"'] \arrow[dr, "g"] & \\
B^A \times A \arrow[r, "ev"] & B
\end{tikzcd}$$

Figure A.5: Exponentiation and Evaluation

In $\mathcal{S}ets$, given two sets A and B, B^A is the set of all functions from A to B. In $\mathcal{A}b$, given two abelian groups, G and H, H^G is an abelian group with elements the group homomorphisms from G to H, with addition defined element-wise and with evaluation the natural $(f, a) \mapsto f(a)$.

Definition A.0.11. *Given objects A, B, and C in a category $\mathcal{C}$ with morphisms $f : A \to C$ and $g : B \to C$, the* pullback *of f and g is an object D with morphisms $\overline{f} : D \to B$ and $\overline{g} : D \to A$ such that:*

(i) $f \circ \overline{g} = g \circ \overline{f}$ and,

(ii): whenever there is an object X with morphisms $h : X \to A$ and $j : X \to B$ such that $f \circ h = f \circ j$, there exists a unique morphism $k : X \to D$ such that $h = \overline{g} \circ k$ and $j = \overline{f} \circ k$.

The definition states that the following diagram commutes,

$$\begin{tikzcd}
X \arrow[dr, dashed, "k"] \arrow[drr, bend left, "j"] \arrow[ddr, bend right, "h"'] & & \\
& D \arrow[r, "\overline{f}"] \arrow[d, "\overline{g}"'] \arrow[dr, phantom, "="] & B \arrow[d, "g"] \\
& A \arrow[r, "f"'] & C
\end{tikzcd}$$

Figure A.6: The Pullback

and the following diagram is called a pullback square.

$$\begin{tikzcd}
D \arrow[r, "\overline{f}"] \arrow[d, "\overline{g}"'] \arrow[dr, phantom, "p.b."] & B \arrow[d, "g"] \\
A \arrow[r, "f"'] & C
\end{tikzcd}$$

Figure A.7: The Pullback Square

The pullback is the limit of the diagram $\bullet \to \bullet \leftarrow \bullet$. In $\mathcal{S}ets$ and $\mathcal{A}b$, the pullback of $A \hookrightarrow C$ and $B \hookrightarrow C$ is the intersection $A \cap B$.

Proposition A.0.12. *If a category $\mathcal{C}$ has products and equalizers, then the pullback of $f : A \to C$ and $g : B \to C$ is the equalizer of $f \circ \pi_A : A \times B \to C$ and $g \circ \pi_B : A \times B \to C$.*

Proof. Let A, B, and C be objects in a category $\mathcal{C}$ with morphisms $f : A \to C$ and $g : B \to C$.

$$\begin{tikzcd} & A \arrow[d, "f"] \\ B \arrow[r, "g"'] & C \end{tikzcd}$$

We first take the product of A and B yielding $A \times B$ with the morphisms $\pi_A : A \times B \to A$ and $\pi_B : A \times B \to B$.

$$\begin{tikzcd} A \times B \arrow[r, "\pi_A"] \arrow[d, "\pi_B"'] & A \arrow[d, "f"] \\ B \arrow[r, "g"'] & C \end{tikzcd}$$

We then take the equalizer of $f \circ \pi_A$ and $g \circ \pi_B$ yielding Eq with morphism $eq : Eq \to A \times B$ such that $f \circ \pi_A \circ eq = g \circ \pi_B \circ eq$.

$$
\begin{tikzcd}
Eq \arrow[dr, "eq"] & & \\
& A \times B \arrow[r, "\pi_A"] \arrow[d, "\pi_B"'] \arrow[dr, phantom, "="] & A \arrow[d, "f"] \\
& B \arrow[r, "g"'] & C
\end{tikzcd}
$$

We claim Eq is the pullback for f and g with morphisms $\pi_A \circ eq : Eq \to A$ and $\pi_B \circ eq : Eq \to B$. Let X be an object in $\mathcal{C}$ with morphism $h : X \to B$ and $j : X \to A$ such that $g \circ h = f \circ j$.

$$
\begin{tikzcd}
X \arrow[rrrdd, bend left, "j"] \arrow[rrdddd, bend right, "h"'] & & & \\
& & & \\
& Eq \arrow[dr, "eq"] & & \\
& & A \times B \arrow[r, "\pi_A"] \arrow[d, "\pi_B"'] \arrow[dr, phantom, "="] & A \arrow[d, "f"] \\
& & B \arrow[r, "g"'] & C
\end{tikzcd}
$$

Then by the universal property of $A \times B$, there is a unqiue morphism $k : X \to A \times B$ such that $\pi_A \circ k = j$ and $\pi_B \circ k = h$.

$$
\begin{tikzcd}
X \arrow[rrrdd, bend left, "j"] \arrow[rrdd, bend left, "k"] \arrow[rrddd, bend right, "h"'] & & & \\
& Eq \arrow[dr, "eq"] & & \\
& & A \times B \arrow[r, "\pi_A"] \arrow[d, "\pi_B"'] \arrow[dr, phantom, "="] & A \arrow[d, "f"] \\
& & B \arrow[r, "g"'] & C
\end{tikzcd}
$$

Then since $g \circ h = f \circ j$, $g \circ \pi_B \circ k = f \circ \pi_A \circ k$. So by the universal property of Eq, there exists a unique morphism $l : X \to Eq$ such that $eq \circ l = k$.

$$
\begin{tikzcd}
X \arrow[rrrdd, bend left, "j"] \arrow[rrdd, bend left, "k"] \arrow[rrddd, bend right, "h"'] \arrow[dr, dashed, "l"] & & & \\
& Eq \arrow[dr, "eq"] & & \\
& & A \times B \arrow[r, "\pi_A"] \arrow[d, "\pi_B"'] \arrow[dr, phantom, "="] & A \arrow[d, "f"] \\
& & B \arrow[r, "g"'] & C
\end{tikzcd}
$$

Since $eq \circ l = k$, $\pi_B \circ eq \circ l = h$ and $\pi_A \circ eq \circ l = j$ as desired. $\square$

Definition A.0.13. *Given a category $\mathcal{C}$ with a terminal object $\mathbf{1}$, then a* subobject classifier *for $\mathcal{C}$ is an object Ω (called the "subobject classifier") with a morphism $\top : \mathbf{1} \to \Omega$ (called "truth") such that for every monomorphism $f : A \rightarrowtail D$, there is a unique morphism $\chi_f : D \to \Omega$ (some say χ_A) such that*

$$\begin{tikzcd} A \arrow[r, "A_1"] \arrow[d, hook, "f"'] \arrow[dr, phantom, "p.b."] & \mathbf{1} \arrow[d, "\top"] \\ D \arrow[r, "\chi_f"] & \Omega \end{tikzcd}$$

is a pullback square.

In $\mathcal{S}ets$, the subobject classifier is the two element set $\Omega = \{0, 1\}$ and $\top : \mathbf{1} = \{1\} \hookrightarrow \Omega = \{0, 1\}$; and there is no subobject classifier in $\mathcal{A}b$.

Definition A.0.14. *Given a functor $F : \mathcal{A} \rightsquigarrow \mathcal{B}$ and a functor $G : \mathcal{B} \rightsquigarrow \mathcal{A}$, F is* left adjoint of *G, $F \dashv G$, if there is a natural bijection in $\mathcal{S}ets$ of $hom_{\mathcal{B}}(F(A), B) \cong hom_{\mathcal{A}}(A, G(B))$.*

Definition A.0.15. *A* concrete *category is a category with an underlying set functor.*

Definition A.0.16. *Given a concrete category $\mathcal{C}$ with an underlying set functor $| - | : \mathcal{C} \rightsquigarrow \mathcal{S}ets$, the* free object *on a set B, $F(B)$, is an object in $\mathcal{C}$ with a function $u : B \to |F(B)|$ in $\mathcal{S}ets$ such that for any objects A in $\mathcal{C}$ with function $g : B \to |A|$ in $\mathcal{S}ets$, there is a unique morphism $\overline{g} : F(B) \to A$ in $\mathcal{C}$ such that $|\overline{g}| \circ u = g$ in $\mathcal{S}ets$.*

Note that this defines a functor $F : \mathcal{S}ets \rightsquigarrow \mathcal{C}$ such that $F \dashv |-|$. The definition states that the following diagram commutes.

$$
\begin{tikzcd}
F(B) \arrow[d, dashed, "\overline{g}"'] \\
A
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
B \arrow[r, "u"] \arrow[dr, "g"'] & {|F(B)|} \arrow[d, dashed, "|\overline{g}|"] \arrow[dl, phantom, "="] \\
& {|A|}
\end{tikzcd}
$$

in $\mathcal{C}$ in $\mathcal{S}ets$

Figure A.8: The Free Object

Definition A.0.17. *Given a concrete category $\mathcal{C}$ with underlying set functor $|-| : \mathcal{C} \rightsquigarrow \mathcal{S}ets$, the* cofree object on a set B, $C(B)$, *is an object in $\mathcal{C}$ with a function $c : |C(B)| \to B$ in $\mathcal{S}ets$ such that for any object A in $\mathcal{C}$ with function $g : |A| \to B$ in $\mathcal{S}ets$, there is a unique morphism $\overline{g} : A \to C(B)$ such that $c \circ |\overline{g}| = g$ in $\mathcal{S}ets$.*

Note that this defines a functor $C : \mathcal{S}ets \rightsquigarrow \mathcal{C}$ such that $|-| \dashv C$. The definition states that the following diagram commutes.

$$
\begin{tikzcd}
A \arrow[d, dashed, "\overline{g}"'] \\
C(B)
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
{|A|} \arrow[d, dashed, "|\overline{g}|"'] \arrow[dr, "g"] \arrow[dr, phantom, "=", shift right=3ex] & \\
{|C(B)|} \arrow[r, "c"] & B
\end{tikzcd}
$$

in $\mathcal{C}$ in $\mathcal{S}ets$

Figure A.9: The Cofree Object

For the next four definitions, we follow [12].

Definition A.0.18. *In a category $\mathcal{C}$ an object P is called* projective *if for every morphism $h : P \to C$ and for every epimorphism $g : B \twoheadrightarrow C$, there is a morphism $\overline{h} : P \to C$ such that $h = g \circ \overline{h}$.*

The definition states that the following diagram commutes.

$$\begin{tikzcd}
& P \arrow[dl, dashed, "\overline{h}"'] \arrow[d, "h"] \arrow[dl, phantom, shift right=3ex, "="] \\
B \arrow[r, two heads, "g"'] & C
\end{tikzcd}$$

Figure A.10: A Projective Object

In $\mathcal{S}ets$, every set is projective, and in $\mathcal{A}b$ the projectives are the free abelian groups.

Definition A.0.19. *A category $\mathcal{C}$* has enough projectives *if for any object C of $\mathcal{C}$ there exists a projective object P in $\mathcal{C}$ and an epimorphism $e : P \twoheadrightarrow C$.*

Definition A.0.20. *In a category $\mathcal{C}$ an object Q is called* injective *if for every morphism $h : C \to Q$ and monomorphism $g : C \rightarrowtail B$, there is a morphism $\overline{h} : B \to Q$ such that $h = \overline{h} \circ g$.*

The definition states that the following diagram commutes.

$$\begin{tikzcd}
C \arrow[r, tail, "g"] \arrow[d, "h"'] \arrow[dr, phantom, shift left=3ex, "="] & B \arrow[dl, dashed, "\overline{h}"] \\
Q &
\end{tikzcd}$$

Figure A.11: An Injective Object

In $\mathcal{S}ets$ every set is injective; and in $\mathcal{A}b$, the divisible abelian groups (like $\mathbb{Q}$) are injectives.

Definition A.0.21. *A category $\mathcal{C}$* has enough injectives *if for any object C of $\mathcal{C}$ there exists an injective object Q in $\mathcal{C}$ and monomorphism $m : C \rightarrowtail Q$.*

Definition A.0.22. *An object G in category $\mathcal{C}$ is a* generator *(also called a* separator*) if for all morphisms $f, g : X \to Y$ in $\mathcal{C}$ such that $f \neq g$, there is a morphism $h : G \to X$ such that $f \circ h \neq g \circ h$.*

In $\mathcal{S}ets$, any non-empty set, e.g. a one element set, is a generator; and in $\mathcal{A}b$, the infinite abelian group $(\mathbb{Z}, +)$ is a generator.

Definition A.0.23. *An object C in a category $\mathcal{C}$ is a* cogenerator *(also called a* coseparator*) if for all morphisms $f, g : X \to Y$ in $\mathcal{C}$ such that $f \neq g$, there is a morphism $h : Y \to C$ such that $h \circ f \neq h \circ g$.*

In $\mathcal{S}ets$, any two element set (or superset there of) is a cogenerator; and in $\mathcal{A}b$ the circle group $(\mathbb{R}/\mathbb{Z}, +)$ is a cogenerator.

Bibliography

[1] J. A. Bondy and U. S. R. Murty, *Graph theory with applications*, American Elsevier Publishing Co., Inc., New York, 1976.

[2] Reinhard Diestel, *Graph theory*, Third, Graduate Texts in Mathematics, vol. 173, Springer-Verlag, Berlin, 2005.

[3] Anton Dochtermann, *Hom complexes and homotopy theory in the category of graphs*, arXiv:math.CO/0605275, 2006.

[4] Robert Goldblatt, *Topoi, the categorial analysis of logic*, Revised, Dover Publications, Inc., Mineola, NY, 2006.

[5] Edgar G. Goodaire and Michael M. Parmenter, *Discrete mathematics with graph theory*, Third, Prentice Hall, Upper Saddle River, New Jersey, 2006.

[6] R. Häggkvist, P. Hell, D. J. Miller, and V. Neumann Lara, *On multiplicative graphs and the product conjecture*, Combinatorica **8** (1988), no. 1, 63–74.

[7] Pavol Hell, *An introduction to the category of graphs*, Topics in graph theory, Ann. New York Acad. Sci. **328** (1977), 120–136.

[8] Pavol Hell and Jaroslav Nešetřil, *Graphs and homomorphisms*, Oxford Lecture Series in Mathematics and its Applications, vol. 28, Oxford University Press, Oxford, 2004.

[9] Horst Herrlich and George E. Strecker, *Category theory: an introduction*, Allyn and Bacon Inc., Boston, Mass., 1973. Allyn and Bacon Series in Advanced Mathematics.

[10] Peter Hilton and Yel Chiang Wu, *A course in modern algebra*, Wiley Classics Library, John Wiley & Sons Inc., New York, 1989. Reprint of the 1974 original, A Wiley-Interscience Publication.

[11] F. William Lawvere, *An elementary theory of the category of sets*, Proc. Nat. Acad. Sci. U.S.A. **52** (1964), 1506–1511.

[12] Saunders Mac Lane, *Categories for the working mathematician*, Second, Graduate Texts in Mathematics, vol. 5, Springer-Verlag, New York, 1998.

[13] Barry Mitchell, *Theory of categories*, Pure and Applied Mathematics, Vol. XVII, Academic Press, New York, 1965.

[14] Jaroslav Nešetřil and Xuding Zhu, *On sparse graphs with given colorings and homomorphisms*, J. Combin. Theory Ser. B **90** (2004), no. 1, 161–172.

[15] Dana I. Schlomiuk, *An elementary theory of the category of topological spaces*, Trans. Amer. Math. Soc. **149** (1970), 259–278.

[16] Claude Tardif, *Multiplicative graphs and semi-lattice endomorphisms in the category of graphs*, J. Combin. Theory Ser. B **95** (2005), no. 2, 338–345.