UiO : Department of Mathematics
University of Oslo

Applications of Directed Algebraic Topology in Optimization Theory

Anders Schnell

Master's Thesis, Spring 2016

Cover design by Martin Helsø

The front page depicts a section of the root system of the exceptional Lie group E_8, projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Abstract

In this thesis, we consider applications of directed algebraic topology in optimization theory, by representing directed graphs as directed topological spaces. We review the classical max-flow min-cut theorem and a generalization of the theorem from numerical to semimodule-valued edge weights, which we use to develop a generalization of the linear programming duality theorem from numerical to semimodule-valued variables for linear programs that correspond to max-flow and min-cut problems.

Acknowledgements

I wish to thank my supervisor Arne B. Sletsjøe for outlining a thesis that accomodated my interest in optimization theory, but also furthered my interest in algebra, and for providing academic advice, such as enlightening insights on algebraic topology and sheaf theory, as well as personal advice, such as how to approach academic articles by reading the conclusions first. I also wish to thank all the people and organizations affiliated with the Faculty of Mathematics and Natural Sciences at the University of Oslo for providing community, opportunity, and, most importantly, coffee and waffles.

-Anders

Contents

Introduction

In this thesis, we consider applications of directed algebraic topology in optimization theory, by representing directed graphs as directed topological spaces. We review the classical max-flow min-cut theorem and a generalization of the theorem from numerical to semimodule-valued edge weights, which we use to develop a generalization of the linear programming duality theorem from numerical to semimodule-valued variables for linear programs that correspond to max-flow and min-cut problems.

In Chapter 1, we review some concepts and results from linear programming, category theory and sheaf theory that will be used in subsequent chapters. We focus on understanding the concepts through examples, since I want to avoid spending too much time on preliminaries.

In Chapter 2, we review the classical max-flow min-cut (MFMC) theorem, which says that the maximum amount of flow between two vertices in a directed graph is equal to the capacity of the smallest bottleneck. The MFMC theorem is an important theorem in optimization theory, which has theoretical applications, such as in deriving Menger's theorem, König's theorem and many other graph-theoretical results, as well as practical applications, such as in operations research and image processing. The definitions, results and proofs are from Geir Dahl's *Network flows and combinatorial matrix theory* [4] and Alexander Schrivjer's *A course in combinatorial optimization* [1], but I have added structure and details to all the proofs and developed examples of all the concepts to better understand them.

In Chapter 3, we review a generalization of Chapter 2's max-flow min-cut (MFMC) theorem from digraphs with numerical edge weights to digraphs with semimodule-valued edge weights, which are represented as partially ordered topological spaces with sheaves of partial semimodules over semirings. The generalized MFMC theorem can be used to solve optimization problems that are expressed as max-flow problems with semimodule-valued edge weights, like vectors, probability distributions and logical statements. The definitions, results and proofs are from Sanjeevi Krishnan's *Flow-cut dualities for sheaves on graphs* [8], but I have added structure and details to all the proofs, though they are otherwise unchanged, and developed examples of many of the concepts to better understand them.

In Chapter 4, we informally consider a generalization of Chapter 1's linear programming (LP) duality theorem from numerical variables to semimodule-valued variables for linear programs that correspond to Chapter 2 and 3's max-flow and min-cut problems. The generalized LP theorem can be used to tabulate and solve graph-related optimization problems that are expressed as max-flow problems with semimodule-valued variables, like vectors, probability distributions and logical statements, or used to tabulate and solve similar optimization problems that are not really graph-related, in which case using the generalized MFMC theorem makes less sense, though I have yet to actually find any such problem. The idea of a "topological approach to LP duality" was suggested in Robert Ghrist and Sarnjeevi Krishnan's *A Topological Max-Flow-Min-Cut Theorem* [9] as a possible application of Krishnan's generalized MFMC theorem, which we review in Chapter 3, and which I have used to develop and prove a generalized LP duality theorem for a special case.

In general, I have focused most of my time and effort on trying to really understand some of the most central concepts in algebraic topology and optimization theory, specifically homology, cohomology, and the relation between (co)homology and graph-problems and linear programs, while approaching directed algebraic topology, category theory and sheaf theory as useful and fascinating tools to obtain that goal.

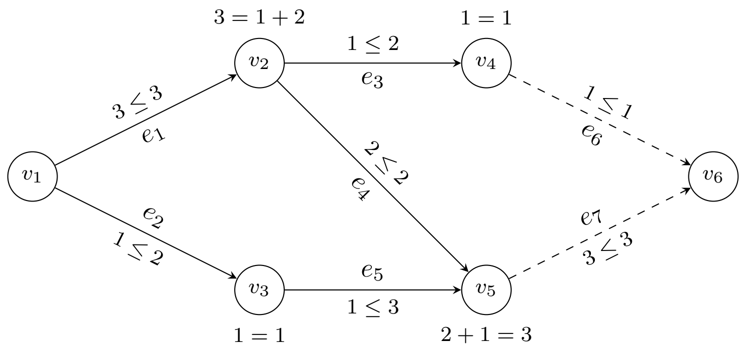

Figure 1: A directed graph containing a max-flow from v_1 to v_6 with value 4 and a min-cut with capacity 4, which exemplifies the max-flow min-cut theorem.

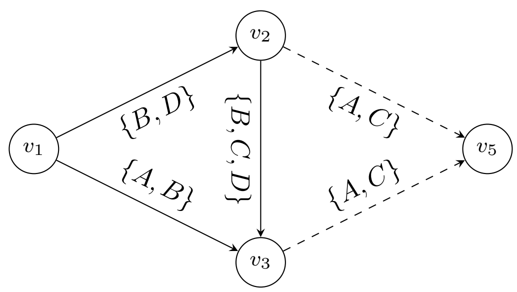

Figure 2: A weighted digraph that has a max-flow consisting of A and $\emptyset$, while the intersection of the min-cut also consists of A and $\emptyset$, which exemplifies the generalized max-flow min-cut theorem.

Chapter 1

Preliminaries

In this chapter, we review some concepts and results from linear programming, category theory and sheaf theory that will be used in subsequent chapters. We focus on understanding the concepts through examples, since I want to avoid spending too much time on preliminaries.

1.1 Linear programming

In this section, we review linear programming, which concerns the problems of maximizing or minimizing a linear function subject to linear constraints defined by equalities or inequalities. Linear programming will be used to formulate the max-flow and min-cut problems in Chapter 2 and the generalized linear programming duality theorem in Chapter 4. The definitions and results are from Alexander Schrivjer's *A course in combinatorial optimization* [1], but I have developed examples of all the concepts to better understand them.

In the first subsection, we consider the standard maximum and minimum problems. In the second and last subsection, we consider the linear programming duality theorem, which says that the solutions of certain pairs of standard problems have the same value.

1.1.1 The standard maximum and minimum problems

We need to define the standard maximum and minimum problems before we can formulate the linear programming duality theorem.

Definition 1.1. *The* ***standard maximum problem*** *is to*

maximize $c^T x$

subject to $Ax \leq b$ *and* $x \in \mathbb{R}^m_+$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ *and* $b \in \mathbb{R}^m$.

Example 1.1. *Consider the problem*

maximize $3x_1 + 2x_2$

subject to $\begin{matrix} x_1 + 2x_2 \leq 2 \\ -x_1 - x_2 \leq -1 \end{matrix}$ *and* $(x_1, x_2) \in \mathbb{R}^2_+$

Then $x = (x_1, x_2) = (2, 0)$ *is an optimal solution with value* 6*, since* $2 + 2 * 0 = 2 \le 2$*,* $-2 - 0 = -2 \le -1$*,* $(2, 0) \in \mathbb{R}^2_+$*, and there is no point with greater value.*

Definition 1.2. *The* ***standard minimum problem*** *is to*

minimize $b^T y$

subject to $A^T y \ge c$ *and* $y \in \mathbb{R}^n_+$

where $b \in \mathbb{R}^m$*,* $A \in \mathbb{R}^{m \times n}$ *and* $c \in \mathbb{R}^n$*.*

Example 1.2. *Consider the problem*

minimize $2y_1 - y_2$

subject to $\begin{matrix} y_1 - y_2 \ge 3 \\ 2y_1 - y_2 \ge 2 \end{matrix}$ *and* $(y_1, y_2) \in \mathbb{R}^2_+$

Then $y = (y_1, y_2) = (3, 0)$ *is an optimal solution with value* 6*, since* $3 - 0 = 3 \ge 3$*,* $2 * 3 - 0 = 6 \ge 2$*,* $(3, 0) \in \mathbb{R}^2_+$*, and there is no point with smaller value.*

Definition 1.3 (Dual, primal)**.** *The* ***primal*** *standard maximum problem*

maximize $c^T x$

subject to $Ax \le b$ *and* $x \in \mathbb{R}^m_+$

has the ***dual*** *standard minimum problem*

minimize $b^T y$

subject to $A^T y \ge c$ *and* $y \in \mathbb{R}^n_+$

where $b \in \mathbb{R}^m$*,* $A \in \mathbb{R}^{m \times n}$ *and* $c \in \mathbb{R}^n$*.*

Remark. *The optimal value of our minimum problem is equal to the optimal value of our maximum problem because our minimum problem is the dual of our maximum problem, which exemplifies the linear programming duality theorem.*

There are criteria for when standard problems have certain types of solutions.

Definition 1.4 (Feasible set, feasible, infeasible)**.** *The* ***feasible set*** *of a standard problem is the polytope subset of* $\mathbb{R}^k$ *that satisfies all the constraints. The problem is* ***feasible*** *if the feasible set is non-empty; otherwise, it is* ***infeasible****.*

Definition 1.5 (Unbounded, bounded)**.** *A standard maximum (minimum) problem is* ***unbounded*** *if its function is positively (negatively) unbounded on the feasible set; otherwise, it is* ***bounded****.*

Remark. *This means that a standard problem is either: (i) bounded feasible, (ii) unbounded feasible, or (iii) infeasible.*

Example 1.3. *See Figure 1.1.*

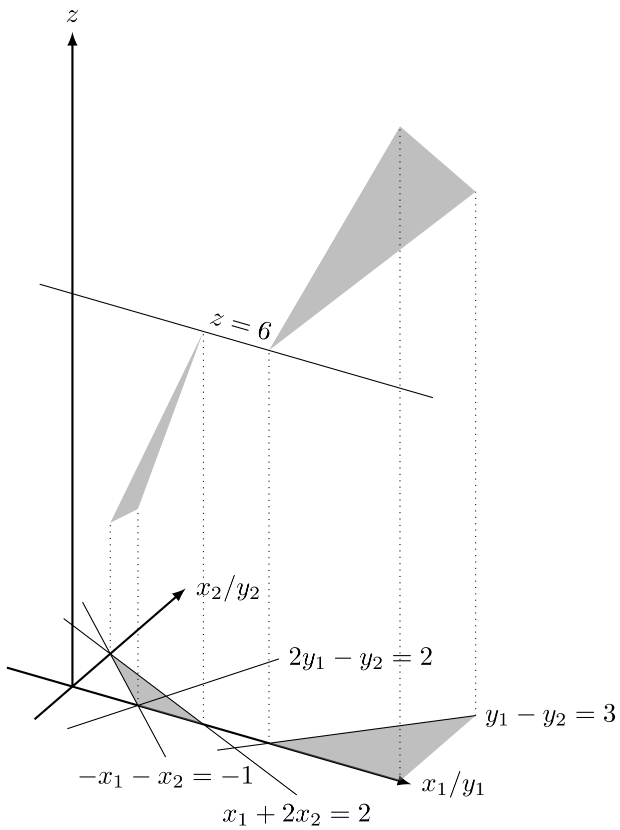

Figure 1.1: Feasible sets from Example 1.1 and Example 1.2 (below), and their images (above), in which both solutions have the same value (line).

1.1.2 The linear programming duality theorem

We can now formulate the linear programming duality theorem, which says that the solutions of primal-dual problems have the same value, and which is an important result in optimization theory that has theoretical applications, such as in deriving Chapter 2's max-flow min-cut theorem, as well as practical applications, such as in operations research.

Theorem 1.1 (The linear programming duality theorem)**.** *Suppose a standard maximum problem is bounded feasible with an optimal solution x^*. Then the dual standard minimum problem is bounded feasible with an optimal solution y^* such that $c^T x^* = b^T y^*$.*

Proof. See [1, p. 35]. □

1.2 Category theory

In this section, we review category theory, which concerns sets of objects together with morphisms between them. Category theory will be used to develop the directed sheaf (co)homology theories and the generalized max-flow min-cut theorem in Chapter 3. The definitions and examples are from Steve Awodey's *Category Theory* [2], but I have added details to some of the examples to better understand them.

In the first subsection, we define categories, functors and natural transformations, which form the basis of category theory. In the second subsection, we define equalizers and coequalizers. In the third subsection, we define limits and colimits. In the fourth and final subsection, we define products, coproducts and monoidal categories.

1.2.1 Categories, functors and natural transformations

We need to define categories and functors before we can define natural transformations.

Definition 1.6 (Category). *A* ***category*** $\mathcal{C}$ *consists of:*

i) *a set* $\mathrm{Ob}(\mathcal{C})$ *of* ***objects***

ii) *for each pair* $X, Y \in \mathrm{Ob}(\mathcal{C})$*, a set* $\mathrm{Mor}(X, Y)$ *of* ***morphisms****, including an identitiy morphism* $\mathbb{1} = \mathbb{1}_X \in \mathrm{Mor}(X, X)$ *when* $Y = X$

iii) *for each triple* $X, Y, Z \in \mathrm{Ob}(\mathcal{C})$*, a* ***composition of morphisms*** *function* $\circ : \mathrm{Mor}(X, Y) \times \mathrm{Mor}(Y, Z) \to \mathrm{Mor}(X, Z)$ *such that* $f \circ \mathbb{1} = f$*,* $\mathbb{1} \circ f = f$*, and* $(f \circ g) \circ h = f \circ (g \circ h)$ *for all appropriate morphisms between* X*,* Y *and* Z *in* $\mathcal{C}$*.*

Example 1.4. *The category* Top *of topological spaces consists of: (i) the set* $\mathrm{Ob}(\mathrm{Top})$ *of all spaces; (ii) for each pair* $X, Y \in \mathrm{Ob}(\mathrm{Top})$*, the set* $C(X, Y)$ *of all continuous functions from* X *to* Y*, including the regular identity function* $\mathbb{1} = \mathbb{1}_X$ *when* $Y = X$*; and (iii) for each triple* $X, Y, Z \in \mathrm{Ob}(\mathrm{Top})$*, regular composition of functions, which satisfies* $f \circ \mathbb{1} = f$*,* $1 \circ f = f$*, and* $(f \circ g) \circ h = f \circ (g \circ h)$ *for all appropriate continuous functions between* X*,* Y *and* Z*.*

Example 1.5. *The category* $\mathrm{Ch}_\bullet(\mathrm{Top})$ *of chain complexes of topological spaces consists of: (i) the set* $\mathrm{Ob}(\mathrm{Ch}_\bullet(\mathrm{Top}))$ *of all chain complexes of spaces: (ii) for each pair* $C_\bullet, D_\bullet \in \mathrm{Ob}(\mathrm{Ch}_\bullet(\mathrm{Top}))$*, the set of all chain maps* $f = \{f_n : C_n \to D_n\}$ *between* $C_\bullet$ *and* $D_\bullet$*; and (iii) for each triple* $C_\bullet, D_\bullet, E_\bullet \in \mathrm{Ob}(\mathrm{Ch}_\bullet(\mathrm{Top}))$*, regular composition of chain maps.*

Example 1.6. *The category* Grp *of groups consists of: (i) the set* $\mathrm{Ob}(\mathrm{Grp})$ *of all groups; (ii) for each pair* $A, B \in \mathrm{Ob}(\mathrm{Grp})$*, the set* $\mathrm{Hom}(A, B)$ *of all group homomorphisms from* A *to* B*, including the identity homomorphism* $\mathbb{1} = \mathbb{1}_A$ *when* $A = B$*; and (iii) for each triple* $A, B, C\, \mathrm{Ob}(\mathrm{Grp})$*, regular composition of homomorphism, which satisfies* $f \circ \mathbb{1} = f$*,* $\mathbb{1} \circ f = f$*, and* $(f \circ g) \circ h = f \circ (g \circ h)$ *for all appropriate homomorphisms between* A*,* B *and* C*.*

Example 1.7. *The category* Set *of sets consists of: (i) the set* $\mathrm{Ob}(\mathrm{Set})$ *of all sets; (ii) for each pair* $M, N \in \mathrm{Ob}(\mathrm{Set})$*, the set of all functions from* M *to* N*, including the identity function* $\mathbb{1} = \mathbb{1}_M$ *when* $M = N$*; and (iii) for each triple* $M, N, O \in \mathrm{Ob}(\mathrm{Set})$*, regular composition of functions, which satisfies* $f \circ \mathbb{1} = f$*,* $\mathbb{1} \circ f = f$*, and* $(f \circ g) \circ h = f \circ (g \circ h)$ *for all appropriate functions between* M*,* N *and* O*.*

Definition 1.7 (Functor, covariant)**.** *A **(covariant) functor** F from a category $\mathcal{C}$ to a category $\mathcal{D}$ consists of:*

i) for each object $X \in \mathrm{Ob}(\mathcal{C})$ in $\mathcal{C}$, an object $F(X) \in \mathrm{Ob}(\mathcal{D})$ in $\mathcal{D}$

ii) for each morphism $f \in \mathrm{Mor}(X, Y)$ in $\mathcal{C}$, a morphism $F(f) \in \mathrm{Mor}(F(X), F(Y))$ in $\mathcal{D}$ such that $F(\mathbb{1}) = \mathbb{1}$ and $F(f \circ g) = F(f) \circ F(g)$ for all morphisms $f : X \to Y$ and $g : Y \to Z$ in $\mathcal{C}$

A contravariant functor similarly assignes a morphism from $F(Y)$ to $F(X)$, rather than from $F(X)$ to $F(Y)$, while the order for composition of morphisms is $F(f \circ g) = F(g) \circ F(f)$, rather than $F(f \circ g) = F(f) \circ F(g)$.

Definition 1.8 (Contravariant functor)**.** *A **contravariant functor** F from a category $\mathcal{C}$ to a category $\mathcal{D}$ consists of:*

i) for each object $X \in \mathrm{Ob}(\mathcal{C})$ in $\mathcal{C}$, an object $F(X) \in \mathrm{Ob}(\mathcal{D})$ in $\mathcal{D}$

ii) for each morphism $f \in \mathrm{Mor}(X, Y)$ in $\mathcal{C}$, a morphism $F(f) \in \mathrm{Mor}(F(Y), F(X))$ in $\mathcal{D}$ such that $F(\mathbb{1}) = \mathbb{1}$ and $F(f \circ g) = F(g) \circ F(f)$ for all morphisms $f : X \to Y$ and $g : Y \to Z$ in $\mathcal{C}$

Example 1.8. *The singular chain complex functor from the category of spaces* Top *to the category of chain complexes of spaces* $\mathrm{Ch}_\bullet(\mathrm{Top})$ *consists of: (i) for each space $X \in \mathrm{Ob}(\mathrm{Top})$, the chain complex of singular chains in X; and (ii) for each continuous function $f \in C(X, Y)$, the induced chain map.*

Example 1.9. *The algebraic homology functor from the category of chain complexes of spaces* $\mathrm{Ch}_\bullet(\mathrm{Top})$ *to the category of sequences of abelian groups* sAb *consists of: (i) for each chain complex, its sequence of homology groups; and (ii) for each chain map, the induced homomorphism on homology.*

Example 1.10. *The functor that assigns to a space its singular homology groups is the composition of the two preceding functors from the category of spaces* Top *to the category of sequences of abelian groups* sAb *consisting of: (i) for each space $X \in \mathrm{Ob}(\mathrm{Top})$, its sequence of homology groups; and (ii) for each continuous function $f \in C(X, Y)$, the induced homomorphism on homology.*

We can now define natural transformations.

Definition 1.9 (Natural transformation)**.** *Let $\mathcal{C}$ and $\mathcal{D}$ be two categories with two functors $F, G : \mathcal{C} \to \mathcal{D}$. A **natural transformation** T from F to G assigns a morphism $T_X : F(X) \to G(X)$ to each object $X \in \mathrm{Ob}(\mathcal{C})$ such that for each morphism $f : X \to Y$ in $\mathcal{C}$, the following diagram commutes:*

$$\begin{array}{ccc} F(X) & \xrightarrow{F(f)} & F(Y) \\ \Big\downarrow{\scriptstyle T_x} & & \Big\downarrow{\scriptstyle T_y} \\ G(X) & \xrightarrow{G(f)} & G(Y) \end{array}$$

Remark. *Natural transformations are similarly defined for contravariant functors.*

Example 1.11. *Consider the long exact sequence*

$$\cdots H_{n+1}(X,A) \xrightarrow{\partial_*} H_n(A) \xrightarrow{i_*} H_n(X) \xrightarrow{j_*} H_n(X,A) \xrightarrow{\partial_*} H_{n-1}(A) \to \cdots$$

of a good pair (X,A) in singular homology. Then the collection of boundary maps $\{\partial_*\}$ is a collection of natural transformations from the relative singular homology functor $H_n(-,-) : \mathrm{Ch}_\bullet(\mathrm{Top}) \to \mathrm{sAb}$ to the ordinary singular homology functor $H_n(-) : \mathrm{Ch}_\bullet(\mathrm{Top}) \to \mathrm{sAb}$.

1.2.2 Equalizers and coequalizers

An equalizer is a generalization of the kernel of the difference of two functions to objects from a category.

Definition 1.10 (Equalizer)**.** *An **equalizer** of a pair of maps $f, g : X \to Y$ between two objects $X, Y \in \mathrm{Ob}(\mathcal{C})$ in a category $\mathcal{C}$ consists of an object $E \in \mathrm{Ob}(\mathcal{C})$ and a map $e : E \to X$ such that*

i) $f \circ e = g \circ e$

ii) for any other map $e' : E' \to X$ such that $f \circ e' = g \circ e'$, there is a unique map $\eta : E' \to E$ such that $e' = e \circ \eta$, i.e., the following diagram commutes

$$\begin{tikzcd}
E \arrow[r, "e"] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \\
E' \arrow[u, "\eta"] \arrow[ur, "e'"'] & &
\end{tikzcd}$$

Example 1.12. *For a pair of maps $f, g : X \to Y$ between two sets $X, Y \in \mathrm{Ob}(\mathrm{Set})$ in the category of sets* Set*, the equalizer is the set $E = \{x \in X : f(x) = g(x)\}$ together with a map $e : E \to X$ that equalizes f and g on X.*

A coequalizer is a generalization of the quotient of a set by an equivalence relation to objects from a category.

Definition 1.11 (Coequalizer)**.** *A **coequalizer** of a pair of maps $f, g : X \to Y$ between two objects $X, Y \in \mathrm{Ob}(\mathcal{C})$ in a category $\mathcal{C}$ consists of an object $C \in \mathrm{Ob}(\mathcal{C})$ and a map $c : X \to Y$ such that*

i) $c \circ f = c \circ g$

ii) for any other map $c' : Y \to C'$ such that $c' \circ f = c' \circ g$, there is a unique map $\gamma : C \to C'$ such that $c' = \gamma \circ c$, i.e., the following diagram commutes

$$\begin{tikzcd}
X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \arrow[r, "c"] \arrow[dr, "c'"'] & C \arrow[d, "\gamma"] \\
& & C'
\end{tikzcd}$$

Example 1.13. *For a pair of maps $f, g : X \to Y$ between two sets $X, Y \in \mathrm{Ob}(\mathrm{Set})$ in the category of sets* Set*, the coequalizer is the quotient set $C = Y/\sim$ together with the canonical map $c : Y \to C$, where $\sim$ is the minimal equivalence relation which identifies $f(x)$ and $g(x)$ for all $x \in X$.*

1.2.3 Limits and colimits

We need to define diagrams and cones before we can define limits.

A diagram is a generalization of indexed collections of sets to collections of objects and morphisms from a category.

Definition 1.12 (Diagram). *A **diagram** of type $\mathcal{J}$ in a category $\mathcal{C}$ is a functor $D : \mathcal{J} \to \mathcal{C}$, where the category $\mathcal{C}$ is called the **index category**.*

Example 1.14. *Suppose A is an object in a category $\mathcal{C}$. Then the constant diagram maps all objects in $\mathcal{J}$ to A and all morphisms in $\mathcal{J}$ to the identity morphism of A.*

Example 1.15. *Suppose $\mathcal{C}$ is a category and $\mathcal{J}$ is a discrete category, whose only morphisms $f : X \to Y$ between objects $X, Y \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$ are the identity morphisms $f = \mathrm{id}_X$ when $Y = X$. Then a diagram of type $\mathcal{J}$ is just a collection of objects in $\mathcal{C}$ indexed by $\mathcal{J}$.*

Definition 1.13 (Cone). *For a diagram $D : \mathcal{J} \to \mathcal{C}$ of type $\mathcal{J}$ in a category $\mathcal{C}$, a **cone** of D is an object $N \in \mathrm{Ob}(\mathcal{C})$ in $\mathcal{C}$ together with a collection $\psi_X : N \to F(X)$ of moprhisms indexed by the objects $X \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$ such that, for every morphism $f : X \to Y$ between objects $X, Y \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$, $F(f) \circ \psi_X = \psi_Y$.*

We can now define limits.

Definition 1.14 (Limit). *For a diagram $D : \mathcal{J} \to \mathcal{C}$ of type $\mathcal{J}$ in a category $\mathcal{C}$, a **limit** of D is a cone (L, ϕ) of D such that, for any other cone (N, ψ) of D, there is a unique morphism $u : N \to L$ such that $\phi_X u = \psi_X$ for all objects $X \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$.*

$$
\begin{tikzcd}
& N \arrow[d, dashed, "u"] \arrow[ddl, "\psi_X"'] \arrow[ddr, "\psi_Y"] & \\
& L \arrow[dl, "\phi_X"] \arrow[dr, "\phi_Y"'] & \\
F(X) \arrow[rr, "F(f)"'] & & F(Y)
\end{tikzcd}
$$

Example 1.16. *Suppose $\mathcal{C}$ is a category and $\mathcal{J}$ is a category with two objects and two morphisms from one of the objects to the other object. Then a diagram of type $\mathcal{J}$ in $\mathcal{C}$ is a pair of morphisms in $\mathcal{C}$, whose limit is an equalizer of those morphisms.*

The dual concepts of limits and cones are colimits and cocones, respectively.

Definition 1.15 (Cocone). *For a diagram $D : \mathcal{J} \to \mathcal{C}$ of type $\mathcal{J}$ in a category $\mathcal{C}$, a **cocone** of D is an object $N \in \mathrm{Ob}(\mathcal{C})$ in $\mathcal{C}$ together with a collection $\psi_X : F(X) \to N$ of moprhisms indexed by the objects $X \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$ such that, for every morphism $f : X \to Y$ between objects $X, Y \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$, $\psi_Y \circ F(f) \circ = \psi_X$.*

Definition 1.16 (Colimit). *For a diagram $D : \mathcal{J} \to \mathcal{C}$ of type $\mathcal{J}$ in a category $\mathcal{C}$, a **colimit** of D is a cone (L, ϕ) of D such that, for any other cocone (N, ψ)*

of D, there is a unique morphism $u : L \to N$ such that $u \circ \phi_X = \psi_X$ for all objects $X \in \mathrm{Ob}(\mathcal{J})$ in $\mathcal{J}$.

$$
\begin{tikzcd}
F(X) \arrow[rr, "F(f)"] \arrow[dr, "\phi_X"] \arrow[ddr, "\psi_Y"'] & & F(Y) \arrow[dl, "\phi_Y"'] \arrow[ddl, "\psi_Y"] \\
& L \arrow[d, dashed, "u"] & \\
& N &
\end{tikzcd}
$$

Example 1.17. *Consider a diagram of type $\mathcal{J}$ in $\mathcal{C}$ from the previous example, which consists of a pair of morphisms in $\mathcal{C}$. Its colimit is a coequalizer of those morphisms.*

1.2.4 Products, coproducts and monoidal categories

A product is a generalization of the Cartesian product of sets to objects from a category.

Definition 1.17 (Product)**.** *A **product** of a pair of objects $X_1, X_2 \in \mathrm{Ob}(\mathcal{C})$ in a category $\mathcal{C}$ is an object $X = X_1 \times X_2 \in \mathrm{Ob}(\mathcal{C})$ together with a pair of morphisms $\pi_1 : X_1 \times X_2 \to X_1, \pi_2 : X_1 \times X_2 \to X_2$ such that, for every object $Y \in \mathrm{Ob}(\mathcal{C})$ and pair of morphisms $f_1 : Y \to X_1, f_2 : Y \to X_2$, there exists a unique morphism $f : Y \to X$ such that the following diagram commutes:*

$$
\begin{tikzcd}
& Y \arrow[dl, "f_1"'] \arrow[d, dashed, "f"] \arrow[dr, "f_2"] & \\
X_1 & X_1 \times X_2 \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & X_2
\end{tikzcd}
$$

Example 1.18. *Suppose $X_1, X_2 \in \mathrm{Ob}(\mathrm{Set})$ are two objects in the category of sets* Set*. Then the product of X_1, X_2 consisting of the object $X_1 \times X_2 = \{(x_1, x_2) : x_1 \in X_1, x_2 \in X_2\} \in \mathrm{Ob}(\mathrm{Set})$ together with the pair of morphisms $\pi_1 : X_1 \times X_2 \to X_1 : (x_1, x_2) \mapsto x_1, \pi_2 : X_1 \times X_2 \to X_2 : (x_1, x_2) \mapsto x_2$ is the Cartesian product.*

A coproduct is a generalization of the disjoint union of sets to objects from a category.

Definition 1.18 (Coproduct)**.** *A **coproduct** of a pair of objects $X_1, X_2 \in \mathrm{Ob}(\mathcal{C})$ in a category $\mathcal{C}$ is an object $X = X_1 \coprod X_2 \in \mathrm{Ob}(\mathcal{C})$ together with a pair of morphisms $i_1 : X_1 \to X_1 \coprod X_2, i_2 : X_2 \to X_1 \coprod X_2$ such that, for every object $Y \in \mathrm{Ob}(\mathcal{C})$ and pair of morphisms $f_1 : X_1 \to Y, f_2 : X_2 \to Y$, there exists a unique morphism $f : X_1 \coprod X_2 \to Y$ such that the following diagram commutes:*

$$
\begin{tikzcd}
& Y & \\
X_1 \arrow[ur, "f_1"] \arrow[r, "i_1"'] & X_1 \coprod X_2 \arrow[u, dashed, "f"] & X_2 \arrow[ul, "f_2"'] \arrow[l, "i_2"]
\end{tikzcd}
$$

Example 1.19. *Suppose* $X_1, X_2 \in \mathrm{Ob}(\mathrm{Set})$ *are two objects in the category of sets* Set*. Then the coproduct of* X_1, X_2 *consisting of the object* $X_1 \coprod X_2 = \{(x_1, 1) : x_1 \in X_1\} \cup \{(x_2, 2) : x_2 \in X_2\} \in \mathrm{Ob}(\mathrm{Set})$ *together with the pair of morphisms* $i_1 : X_1 \to X_1 \coprod X_2 : x_1 \mapsto (x_1, 1), i_2 : X_2 \to X_1 \coprod X_2 : x_2 \mapsto (x_2, 2)$ *is the disjoint union.*

We need to define bifunctors before we can define monoidal categories.

Definition 1.19 (Bifunctor)**.** *A* ***bifunctor*** *is a functor* $\otimes : \mathcal{C} \times \mathcal{D} \to \mathcal{E}$ *from the product category of two categories* $\mathcal{C}$ *and* $\mathcal{D}$ *to a category* $\mathcal{E}$*.*

We can now define monoidal categories.

Definition 1.20 (Monoidal category)**.** *A* ***monoidal category*** $\mathcal{M}_C$ *consists of:*

i) *a category* $\mathcal{C}$

ii) *a bifunctor* $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ *on* $\mathcal{C}$

iii) *an object* $1 \in \mathcal{C}$*, called the* ***unit object***

iv) *a natural isomorphism* α *with a component* $\alpha_{A,B,C} : (A \otimes B) \otimes \simeq A \otimes (B \otimes C)$ *for each triple* $A, B, C \in \mathrm{Ob}(\mathcal{C})$*, called the* ***associator***

v) *a natural isomorphism* λ *with a component* $\lambda_A : 1 \otimes A \to A$ *for each* $A \in \mathrm{Ob}(\mathcal{C})$*, called the* ***left unitor***

vi) *a natural isomorphism* ρ *with a component* $\rho : A \otimes 1 \to A$ *for each* $A \in \mathrm{Ob}(\mathcal{C})$*, called the* ***right unitor***

where the three natural transformations are such that, for all $A, B, C, D \in \mathrm{Ob}(\mathcal{C})$*, the* ***pentagon diagram*** *and the* ***triangle diagram*** *commutes:*

$$\begin{tikzcd}
((A \otimes B) \otimes C) \otimes D \arrow[r, "\alpha_{A,B,C} \otimes 1_D"] \arrow[d, "\alpha_{A \otimes B, C, D}"] & (A \otimes (B \otimes C)) \otimes D \arrow[r, "\alpha_{A, B \otimes C, D}"] & A \otimes ((B \otimes C) \otimes D) \arrow[d, "1_A \otimes \alpha_{B,C,D}"] \\
(A \otimes B) \otimes (C \otimes D) \arrow[rr, "\alpha_{A,B,C \otimes D}"] & & A \otimes (B \otimes (C \otimes D))
\end{tikzcd}$$

$$\begin{tikzcd}
(A \otimes 1) \otimes B \arrow[rr, "\alpha_{A,1,B}"] \arrow[dr, "\rho_A \otimes 1_B"] & & A \otimes (1 \otimes B) \arrow[dl, "1_A \otimes \lambda_B"] \\
& A \otimes B &
\end{tikzcd}$$

Example 1.20. *The category of vector spaces together with the ordinary tensor product is a monoidal category.*

Definition 1.21 (Monoid object)**.** *A* ***monoid object*** *in a monoidal category* $M_{\mathcal{C}}$ *consists of:*

i) *an object* $M \in \mathrm{Ob}(\mathcal{C})$

ii) *a morphism* $\mu : M \otimes M \to M$*, called* ***multiplication***

iii) *a morphism* $\eta : 1 \to M$*, called* ***unit*** *where the two morphisms are such that the* ***pentagon diagram*** *and the* ***unitor diagram*** *commutes:*

$$\begin{tikzcd}
(M \otimes M) \otimes M \arrow[r, "\alpha"] \arrow[d, "\mu\otimes 1"] & M \otimes (M \otimes M) \arrow[r, "1\otimes\mu"] & M \otimes M \arrow[d, "\mu"] \\
M \otimes M \arrow[rr, "\mu"] & & M
\end{tikzcd}$$

$$\begin{tikzcd}
I \otimes M \arrow[r, "\eta\otimes 1"] \arrow[dr, "\lambda"'] & M \otimes M \arrow[r, leftarrow, "1\otimes\eta"'] \arrow[d, "\mu"] & M \otimes 1 \arrow[dl, "\rho"] \\
& M &
\end{tikzcd}$$

Example 1.21. *The monoid objects in the monoidal category of sets together with the Cartesian product are the monoids of abstract algebra.*

1.3 Sheaf theory

In this section, we review sheaf theory, which concerns assignments of objects from categories to the open sets of topological spaces together with morphisms between them. Sheaf theory will be used to develop the directed sheaf (co)homology theories and the generalized max-flow min-cut theorem in Chapter 3. The definitions and examples are from B.R. Tennison's *Sheaf Theory* [3], but I have added details to some of the examples to better understand them.

In the first subsection, we consider presheaves, which are assignments of objects from a category to the open sets of a topological space. In the second subsection, we consider the stalks of a presheaf, which characterize the presheaf in the neighborhood of a point. In the third and final subsection, we consider sheaves, which are presheaves that satisfy two additional conditions.

1.3.1 Presheaves

We need to define presheaves before we can define the stalks of presheaves and proper sheaves.

A presheaf is an assignment of objects from a cateogry to the open sets of a topological space together with restriction maps from each open set to each open set contained in it.

Definition 1.22 (Presheaf)**.** *Let X be a topological space. A* ***presheaf*** *F of sets on X consists of*

(i) for each open set U of X, a set $F(U)$, which is called the set of ***sections*** *of F over U*

(ii) for each pair of open sets $V \subseteq U$ of X, a ***restriction map*** *$\rho_{V,U} : F(U) \to F(V)$ such that $\rho_{U,U} = \mathrm{id}_U$ and, for each open set $W \subseteq V$, $\rho_{W,U} = \rho_{W,V} \circ \rho_{V,U}$*

Presheaves can also be defined category theoretically.

Definition 1.23 (Presheaf)**.** *Let X be a topological space. A* ***presheaf*** *F of sets on X is a functor $F : X \to \mathrm{Set}$.*

Example 1.22. *Let X be a topological space and let A be a set. The constant presheaf F over X with value A consist of*

i) for each open set U of A, the set $F(U) = A$

ii) for each pair of open sets $V \subseteq U$ of X, the restriction map $\rho_{V,U} : F(U) \to F(V)$ defined by $\rho_{V,U} = \mathbb{1}_A$.

Example 1.23. *Let $X = \{p, q\}$ be a two-point space with discrete topology, whose open sets are $\{\emptyset\}$, $\{p\}$, $\{q\}$ and $\{p, q\}$. The constant presheaf F over X with value the integers $\mathbb{Z}$ consists of*

i) the sets $F(\{\emptyset\}) = \mathbb{Z}$, $F(\{p\}) = \mathbb{Z}$, $F(\{q\}) = \mathbb{Z}$ and $F(\{p, q\}) = \mathbb{Z}$

ii) the restriction maps $\rho_{\emptyset,\{p,q\}} = \mathrm{id}_{\mathbb{Z}}$, $\rho_{\{p\},\{p,q\}} = \mathrm{id}_{\mathbb{Z}}$, $\rho_{\{q\},\{p,q\}} = \mathrm{id}_{\mathbb{Z}}$, $\rho_{\emptyset,\{p\}} = \mathrm{id}_{\mathbb{Z}}$ and $\rho_{\emptyset,\{q\}} = \mathrm{id}_{\mathbb{Z}}$

$$
\begin{tikzcd}
 & F(\emptyset) = \mathbb{Z} & \\
F(\{p\}) = \mathbb{Z} \arrow[ur, "\mathrm{id}_{\mathbb{Z}}"] & & F(\{q\}) = \mathbb{Z} \arrow[ul, "\mathrm{id}_{\mathbb{Z}}"'] \\
 & F(\{p, q\}) = \mathbb{Z} \arrow[uu, "\mathrm{id}_{\mathbb{Z}}"] \arrow[ul, "\mathrm{id}_{\mathbb{Z}}"] \arrow[ur, "\mathrm{id}_{\mathbb{Z}}"'] &
\end{tikzcd}
$$

1.3.2 Stalks of presheaves

We need to define preorder, preordered sets, directed sets and direct limits of directed systems before defining the stalks of a presheaves.

Definition 1.24 (Preorder, preordered set)**.** *A **preorder** on a set Λ is a binary relation $\leq$ such that*

i) for all $\alpha \in \Lambda$, $\alpha \leq \alpha$ (reflexive)

ii) for all $\alpha, \beta, \gamma \in \Lambda$, if $\alpha \leq \beta$ and $\beta \leq \gamma$, then $\alpha \leq \gamma$ (transitive)

*A set together with a preorder is called a **preordered set**.*

Definition 1.25 (Directed set)**.** *A **directed set** is a set Λ together with a preorder $\leq$ such that for all $\alpha, \beta \in \Lambda$, there exists a $\gamma \in \Lambda$ such that $\alpha \leq \gamma$ and $\beta \leq \gamma$.*

Notation. $\Lambda_1 = \{(\alpha, \beta) \in \Lambda \times \Lambda : \alpha \leq \beta\}$ *denotes a directed set Λ together with a preorder $\leq$.*

Example 1.24. *Suppose X is a topological space. Then the set $\mathcal{T}$ of open sets of X together with the relation $\leq$ defined by $U \leq V$ if and only if $V \subseteq U \in \mathcal{T}$ constitute a directed set $\mathcal{T}_1$.*

Definition 1.26 (Direct system)**.** *A **direct system** of sets indexed by a directed set Λ is a collection of sets $(U_\alpha)_{\alpha \in \Lambda}$ together with a collection of maps $(\rho_{\alpha\beta} : U_\alpha \to U_\beta)_{(\alpha,\beta) \in \Lambda_1}$ such that* *i) for all* $\alpha \in \Lambda$, $\rho_{\alpha\alpha} = \mathrm{id}_{U_\alpha}$

ii) for all $\alpha, \beta, \gamma \in \Lambda$, *if* $a \leq \beta \leq \gamma$, *then* $\rho_{\alpha\gamma} = \rho_{\beta\gamma} \circ \rho_{\alpha\beta}$.

Example 1.25. *Suppose* F *is a presheaf on a topological space* X. *Let* $\rho_{UV} = \rho_{V,U}$ *be the restriction map from the sheaf when* $U \leq V$. *Then the collection of sets* $(F(U))_{U\in\mathcal{T}}$ *together with the collection of maps* $(\rho_{UV})_{U,V\in\mathcal{T}}$ *constitute a direct system of sets indexed by the directed set* $\mathcal{T}_1$ *from the previous example.*

Definition 1.27 (Target)**.** *Let* Λ_1 *be a direct system. A* ***target*** *for the system is a set* V *together with a collection of maps* $(\rho_\alpha : U_\alpha \to V)_{\alpha\in\Lambda}$ *such that for all* $\alpha \leq \beta$, $\rho_\alpha = \rho_\beta \circ \rho_{\alpha\beta}$.

Definition 1.28 (Direct limit)**.** *Let* Λ_1 *be a direct system. A* ***direct limit*** *for the system is a target* $(V, (\tau_\alpha : U_\alpha \to U)_{\alpha\in\Lambda})$ *such that for any other target* $(V, (\rho_\alpha : U_\alpha \to V)_{\alpha\in\Lambda})$, *there exists a unique map* $f : U \to V$ *such that for all* $\alpha \in \Lambda$, $\rho_\alpha = \tau_\alpha \circ f$.

Remark. *This means that a direct limit is the most direct target among all targets. Furthermore, we can speak of a singular direct limit, since all direct limits of a direct system are naturally isomorphic.*

Notation. $\varinjlim_{\alpha\in\Lambda} U_\alpha$ *denotes the direct limit of the direct system* Λ_1.

We can now define the stalks of a presheaf. In the following, suppose that F is a presheaf over a topological space X and that $x \in X$.

Proposition 1.1. *The collection of sets of sections* $(F(U))_{U\ni x}$ *together with the collection of restriction maps* $(\rho_{V,U} : F(U) \to F(V))_{U\supseteq V\ni x}$ *constitute a direct system.*

Definition 1.29 (Stalk, germs)**.** *The* ***stalk*** F_x *of* F *at* x *is the direct limit*

$$\varinjlim_{U\ni x} F(U)$$

of the direct system from the previous proposition together with the collection of maps

$$(F(U) \to F_x : s \mapsto s_x)_{U\ni x}$$

The elements of F_x *are called* ***germs*** *of sections of* F.

Example 1.26. *Consider the constant presheaf* F *over* X *with value* A *from Example 1.22. Then, for each* $x \in X$, *the stalk of* F *at* x *is* $F_x = A$.

Proposition 1.2. *For each germ* $t \in F_x$, *there exists a neighborhood* U *of* x *such that* $t = s_x$ *for some* $s \in F(U)$.

Proposition 1.3. *For each pair of germs* $s_x, t_x \in F_x$ *with* $s \in F(U)$ *and* $t \in F(V)$, $s_x = t_x$ *if and only if there exists an open set* $W \subseteq U \cap V$ *such that* $\rho_{W,U}(s) = \rho_{W,V}(t)$.

1.3.3 Sheaves

A sheaf is a presheaf that satisifes two additional conditions, concerning the existence and uniqueness of sections with certain local properties that assures they can be glued together in a consistent way.

Definition 1.30 (Sheaf, locality, gluing)**.** *Let X be a topological space. A* ***sheaf*** *is a presheaf F of sets on X such that*

(i) for each open covering $(U_\alpha)_{\alpha\in\mathcal{A}}$ of an open subset U of X and for each pair of sections s, t in $F(U)$ such that $\rho_{U_\alpha,U}(s) = \rho_{U_\alpha,U}(t)$ for all $\alpha \in \mathcal{A}$, $s = t$ (locality)

(ii) for each open covering $(U_\alpha)_{\alpha\in\mathcal{A}}$ of an open subset U of X and for each family of sections $(s_\alpha)_{\alpha\in\mathcal{A}}$ in $F(U)$ such that $\rho_{U_\alpha\cap U_\beta,U_\alpha}(s_\alpha) = \rho_{U_\beta\cap U_\alpha,U_\beta}(s_\beta)$ for all $\alpha \neq \beta$, there is a section s in $F(U)$ such that $\rho_{U_\alpha,U}(s) = s_\alpha$ for all $\alpha \in \mathcal{A}$ (gluing)

Example 1.27. *Let $X = \{p, q\}$ be a two-point space with discrete topology, whose open sets are $\{\emptyset\}$, $\{p\}$, $\{q\}$ and $\{p, q\}$. The constant sheaf F over X with value the integers $\mathbb{Z}$ consists of:*

i) the sets $F(\{\emptyset\}) = 0$, $F(\{p\}) = \mathbb{Z}$, $F(\{q\}) = \mathbb{Z}$ and $F(\{p, q\}) = \mathbb{Z} \oplus \mathbb{Z}$

ii) the restriction maps $\rho_{\emptyset,\{p,q\}} = 0$, $\rho_{\{p\},\{p,q\}} : \mathbb{Z}\oplus\mathbb{Z} \to \mathbb{Z}$, $\rho_{\{q\},\{p,q\}} : \mathbb{Z}\oplus\mathbb{Z} \to \mathbb{Z}$, $\rho_{\emptyset,\{p\}} = 0$ and $\rho_{\emptyset,\{q\}} = 0$, where $\mathbb{Z} \oplus \mathbb{Z} \to \mathbb{Z}$ are projection maps

$$
\begin{tikzcd}
 & F(\emptyset) = 0 & \\
F(\{p\}) = \mathbb{Z} \arrow[ur, "0"] & & F(\{q\}) = \mathbb{Z} \arrow[ul, "0"'] \\
 & F(\{p,q\}) = \mathbb{Z}\oplus\mathbb{Z} \arrow[uu, "0"] \arrow[ul, "\mathbb{Z}\oplus\mathbb{Z}\to\mathbb{Z}"] \arrow[ur, "\mathbb{Z}\oplus\mathbb{Z}\to\mathbb{Z}"'] &
\end{tikzcd}
$$

Chapter 2

The max-flow min-cut theorem

In this chapter, we review the classical max-flow min-cut (MFMC) theorem, which says that the maximum amount of flow between two vertices in a directed graph is equal to the capacity of the smallest bottleneck. The MFMC theorem is an important theorem in optimization theory, which has theoretical applications, such as in deriving Menger's theorem, König's theorem and many other graph-theoretical results, as well as practical applications, such as in operations research and image processing. The definitions, results and proofs are from Geir Dahl's *Network flows and combinatorial matrix theory* [4] and Alexander Schrivjer's *A course in combinatorial optimization* [1], but I have added structure and details to all the proofs and developed examples of all the concepts to better understand them.

In the first section. we define graphs, flows and capacities. In the second section, we consider the maximum flow problem, the minimum cut problem, and the MFMC theorem, which relates the two problems. In the third and final section, we consider two applications of the MFMC theorem, by proving Menger's theorem and König's theorem.

2.1 Graphs, flows and capacities

We need to define graphs, flows and capacities before we can formulate the max-flow min-cut theorem.

Definition 2.1 (Graph, vertex, edge)**.** *A (directed)* ***graph*** *is a pair $G = (V, E)$ such that V is a finite set and E is a set consisting of (ordered) pairs of elements from V. Elements of V are called* ***vertices*** *and elements of E are called* ***edges****.*

Notation. *For each vertex $v \in V$ and each vertex subset $S \subseteq V$, we have*

$$\begin{aligned}
\delta^+(v) &= \{e \in E : e = (v, w), w \in E\} && \textit{(outgoing edges from } v\textit{)}\\
\delta^-(v) &= \{e \in E : e = (w, v), w \in E\} && \textit{(incoming edges to } v\textit{)}\\
\delta^+(S) &= \{e \in E : e = (v, w), v \in S, w \notin S\} && \textit{(outgoing edges from } S\textit{)}\\
\delta^-(S) &= \{e \in E : e = (v, w), v \notin S, w \in S\} && \textit{(incoming edges to } S\textit{)}
\end{aligned}$$

A flow function on a graph assigns an amount of flow through each edge.

Definition 2.2 (Flow). *A **flow** on G is a function $f : E \to \mathbb{R}$.*

Notation. *For each flow f and each vertex $v \in V$, we have*

$$\begin{array}{ll} \sum_{e\in\delta^+(v)} f(e) & \textit{(total outflow from } v\textit{)} \\ \sum_{e\in\delta^-(v)} f(e) & \textit{(total inflow to } v\textit{)} \end{array}$$

The divergence of flow from a vertex is the net outflow.

Definition 2.3 (Divergence). *For each flow f and each vertex $v \in V$, the **divergence** in v is given by a function $div_f : V \to \mathbb{R}$ defined as*

$$div_f(v) = \sum_{e\in\delta^+(v)} f(e) - \sum_{e\in\delta^-(v)} f(e)$$

A circulation is a flow without divergence.

Definition 2.4 (Circulation, flow conservation). *A **circulation** is a flow with no divergence in any vertex, which is said to satisfy **flow conservation**.*

A capacity function on a graph assigns a constraint on the amount of flow through each edge.

Definition 2.5 (Capacity). *A **capacity** on G is a function $c : E \to \mathbb{R}_+$.*

2.2 The max-flow min-cut theorem

The maximum flow problem is to find an st-flow with maximum value (max-flow), and the minimum cut problem is to find an st-cut with minimum capacity (min-cut), while the max-flow min-cut (MFMC) theorem says that the value of a max-flow is equal to the capacity of a min-cut. This relation between the maximum flow problem and the minimum cut problem was found by Air Force researchers T.E. Harris and F.S. Ross while studying the rail network (a directed graph) between Russia and Eastern European countries during the 1950s. In a classified report [5] published in 1955 and declassified in 1999, they described a method for finding "the bottleneck" (a min-cut) of the maximum flow (a max-flow) of goods from Russia (the source) to nearby countries (the targets). The MFMC theorem was first proven for undirected graphs, in 1954, by Ford and Fulkerson [6], who referenced T.E. Harris' formulation of the maximum flow problem. In 1955, Dantzig and Fulkerson [7] proved that the theorem also holds for directed graphs, which will be considered here.

We need to formulate the maximum flow problem and the minimum cut problem before we can formulate the MFMC theorem. In the following, suppose G is a directed graph with capacity function c and vertices $s, t \in V$.

2.2.1 The maximum flow problem

We need to define st-flows and the value of an st-flow before we can formulate the maximum flow problem.

An st-flow is a flow from a source vertex to a target vertex that satisfies flow conservation in all intermediate vertices and capacity constraints on all edges.

Definition 2.6 (st-flow, source, target)**.** *An **st-flow** is a flow f such that*

i) for each vertex $v \in V \setminus \{s,t\}$, $\sum_{e \in \delta^+(v)} f(e) = \sum_{e \in \delta^-(v)} f(e)$

ii) for each edge $e \in E$, $0 \le f(e) \le c(e)$

*The vertex s is called the **source** and t is called the **target** of the st-flow.*

The value of an st-flow is the total outflow from the source vertex.

Definition 2.7 (Value of an st-flow)**.** *The **value** of an st-flow f is*

$$val(f) = \sum_{e \in \delta^+(s)} f(e)$$

Remark. *The total inflow to the target vertex equals the total outflow from the source vertex, since there is flow conservation in all intermediate vertices.*

We can now formulate the maximum flow problem.

Definition 2.8 (The maximum flow problem, maximum flow)**.** ***The maximum flow problem** is to find an st-flow f with maximal value $val(f)$. A solution to the problem is called a **maximum flow**.*

Example 2.1. *The flow on the graph in Figure 2.1 is an v_1v_6-flow, since (i) it satisfies flow conservation in all intermediate vertices; and (ii) it satisfies the capacity constraints on all edges. This v_1v_6-flow has value 4, since $val(f) = f(e_1) + f(e_2) = 3 + 1 = 4$, and it is a maximum flow, since no modification yields an v_1v_6-flow with higher value.*

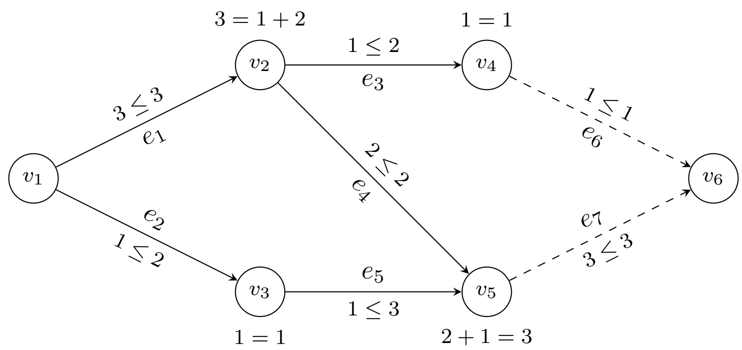

Figure 2.1: Vertices with inflow = outflow, edges with flow ≤ capacity.

The maximum flow problem can also be formulated as a standard linear programming maximum problem (see Definition 1.1), by having an edge from the source to the target and requiring flow conservation in all vertices.

Definition 2.9. *Suppose $G = (\{v_1, \ldots, v_m\}, \{e_1, \ldots, e_n\})$ is an enumerated directed graph with an edge $e_n = (v_m, v_1)$ whose capacity is*

$$c(e_n) = \min\left(\sum_{e \in \partial^+(v_1)} c(e), \sum_{e \in \partial^-(v_m)} c(e)\right)$$

*The **max-flow linear programming problem** is to*

maximize x_n

subject to $Ax \leq b$ and $x \in \mathbb{R}^n_+$

where $A \in \mathbb{R}^{(m+n)\times n}$ and $b \in \mathbb{R}^{m+n}$ are defined by

$$[a_{ij}]_{i=1:j}^{i=m;j} = \begin{cases} 1 & e_j \in \partial^+(v_i) \\ -1 & e_j \in \partial^-(v_i) \\ 0 & e_j \notin \partial^\pm(v_i) \end{cases}, [a_{ij}]_{i=m+1:j}^{i=m+n;j} = \begin{cases} 1 & i = m+j \\ 0 & i \neq m+j \end{cases}$$

$$[b_i]_{i=1}^{i=m} = 0, [b_i]_{i=m+1}^{i=m+n} = c(e_i)$$

Remark. *The matrix has a row for each vertex and edge and a column for each edge, where the upper part of the matrix represents the vertices and the lower part represents the edges. The constraint vector has an entry for each vertex and edge, where the upper part of the vector represents the vertex flow conservation constraints and the lower part represents the edge capacity constraints.*

Example 2.2. *The max-flow linear programming problem of the max-flow problem from the previous example is to*

maximize x_8

subject to $Ax \leq b$ and $x \in \mathbb{R}^8_+$

where

$$Ax = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 2 \\ 2 \\ 2 \\ 3 \\ 1 \\ 3 \\ 4 \end{bmatrix} = b$$

which has the solution $x = (3, 1, 1, 2, 1, 1, 3, 4)$ with value $x_8 = 4$, which also was the value of the max-flow in the previous example.

2.2.2 The minimum cut problem

We need to define st-cuts and the capacity of an st-cut before we can formulate the minimum cut problem.

An st-cut is an edge subset whose removal seperates the source vertex from the target vertex.

Definition 2.10 (st-cut)**.** *An* ***st-cut*** *is an edge subset $K = \delta^+(S) \subset E$ such that $S \subset V$ with $s \in S$ and $t \notin S$.*

The capacity of an st-cut is the sum of the capacities of its edges.

Definition 2.11 (Capacity of an st-cut)**.** *The* ***capacity of an st-cut*** *K is*

$$cap_c(K) = \sum_{e \in K} c(e)$$

We can now formulate the minimum cut problem.

Definition 2.12 (The minimum cut problem, minimum cut)**.** ***The minimum cut problem*** *is to find an st-cut K with minimal capacity $cap_c(K)$. A solution to the problem is called a* ***minimum cut****.*

Example 2.3. *The edge subset $K = \delta^+(S) = \{e_6, e_7\}$ (dashed edges in Figure 2.1) of the graph from the previous example is a v_1v_6-cut, since the vertex subset $S = \{v_1, v_2, v_3, v_4, v_5\}$ contains v_1 but not v_6. This v_1v_6-cut has capacity 4, since $cap_c(K) = c(e_6) + c(e_7) = 1 + 3 = 4$, and it is a minimum cut, since no other v_1v_6-cut has lower capacity.*

Remark. *The capacity of our minimum cut is equal to the value of our maximum flow, which exemplifies the max-flow min-cut theorem.*

The minimum cut problem can also be formulated as a standard linear programming minimum problem (see Definition 1.2).

Definition 2.13. *Suppose $G = (\{v_1, \dots, v_m\}, \{e_1, \dots, e_n\})$ is an enumerated directed graph with an edge $e_n = (v_m, v_1)$ whose capacity is*

$$c(e_n) = \min\left(\sum_{e \in \partial^+(v_1)} c(e), \sum_{e \in \partial^-(v_m)} c(e)\right)$$

The ***min-cut linear programming problem*** *is to*

minimize $b^T y$

subject to $A^T y \geq c$ and $y \in \{0,1\}^{m+n}$

where $A \in \mathbb{R}^{(m+n)\times n}$, $b \in \mathbb{R}_+^{m+n}$ and $c \in \mathbb{R}^n$ are defined by

$$[a_{ij}]_{i=1:j}^{i=m;j} = \begin{cases} 1 & e_j \in \partial^+(v_i) \\ -1 & e_j \in \partial^-(v_i) \\ 0 & e_j \notin \partial^\pm(v_i) \end{cases}, [a_{ij}]_{i=m+1:j}^{i=m+n;j} = \begin{cases} 1 & i = m+j \\ 0 & i \neq m+j \end{cases}$$

$$[b_i]_{i=1}^{i=m} = 0, [b_i]_{i=m+1}^{i=m+n} = c(e_i)$$

$$[c_i]_{i=1}^{i=n-1} = 0, [c_i]_{i=n}^{i=n} = 1$$

Remark. *The variable vector has an entry for each vertex and edge, where an entry equals one if the corresponding edge is part of the cut and zero otherwise. Furthermore, the min-flow linear programming problem is the dual (see Definition 1.3) of the max-flow linear programming problem.*

Example 2.4. *The min-cut linear programming problem of the min-cut problem from the previous example is to*

minimize $b^T y$

subject to $A^T y \geq c$ *and* $y \in \mathbb{R}^{14}_{+}$

where

$$b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 2 \\ 2 \\ 2 \\ 3 \\ 1 \\ 3 \\ 4 \end{bmatrix}, A^T y = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = c$$

which has the solution $y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0)$ *with value* $b^T y = 4$, *which also was the capacity of the min-cut in the previous example.*

2.2.3 The max-flow min-cut theorem

We can now formulate the max-flow min-cut theorem.

Theorem 2.1 (The max-flow min-cut theorem). *Suppose* G *is a directed graph with capacity function* c *and vertices* $s, t \in V$. *Then the value of a maximum flow is equal to the capacity of a minimum cut.*

We need to formulate Hoffman's circulation theorem and the weak MFMC theorem before we can prove the MFMC theorem.

Hoffman's circulation theorem says that there always exists a circulation that satisfy certain capacity conditions.

Theorem 2.2 (Hoffman's circulation theorem). *Suppose* G *is a directed graph and* $l, u : E \to \mathbb{R}$ *are capacity functions satisfying* $l \leq u$. *Then there exists a circulation* f *such that* $l \leq f \leq u$ *if and only if*

$$\sum_{e \in \delta^-(S)} l(e) \leq \sum_{e \in \delta^+(S)} u(e)$$

for all vertex subsets $S \subseteq V$.

The auxiliary graph is a modified graph whose edges satisfy certain capacity conditions, which will be used to prove Hoffman's circulation theorem.

Definition 2.14 (Auxiliary graph). *The* ***auxiliary graph*** *of* G *with respect to a flow* f *is the graph* $G_f = (V, E_f)$*, where* $E_f = \{e \in E : f(e) < u(e)\} \cup \{e^{-1} : e \in E, l(e) < f(e)\}$.

Proof of Hoffman's circulation theorem. Assume that f is a circulation such that $l \leq f \leq u$. Then

$$\sum_{e \in \delta^-(S)} l(e) \leq \sum_{e \in \delta^-(S)} f(e) = \sum_{e \in \delta^+(S)} f(e) \leq \sum_{e \in \delta^+(S)} u(e)$$

where the equality follows from circulations satisfying flow conservation.

Assume that

$$\sum_{e \in \delta^-(S)} l(e) \leq \sum_{e \in \delta^+(S)} u(e)$$

for all vertex subsets $S \subseteq V$. Let f be a flow such that $l \leq f \leq u$ and $\|div_f\|$ is minimized (existence by the extreme value theorem of analysis). We will show that f is actually a circulation. Define

$$V^- = \{v \in V : div_f(v) < 0\}, V^+ = \{v \in V : div_f(v) > 0\}$$

If $V^- \neq \emptyset$, we can deduce a contradiction. If the auxiliary graph $G_f = (V, E_f)$ contains a path P from a vertex in V^- to a vertex in V^+, then we can modify f by adding some small number ϵ to each edge along P, which leads to another flow g with $l \leq g \leq u$ and $\|div_g\| < \|div_f\|$. But this contradicts the assumption that f is minimizing, so we may assume that no such path P exists. Define S to be the set of vertices reachable in D_f from a vertex in V^-. Then for each $e \in \delta^+(S)$, we have $e \notin D_f$, so $x(e) = u(e)$; and for each $e \in \delta^-(S)$, we have $e^{-1} \notin D_f$, so $x(e) = l(e)$. This gives

$$\sum_{e \in \delta^+(S)} u(e) - \sum_{e \in \delta^-(S)} l(e) = \sum_{e \in \delta^+(S)} x(e) - \sum_{e \in \delta^-(S)} x(e)$$

$$= \sum_{v \in S} div_f(v) = \sum_{v \in V^-} div_f(v) < 0$$

which contradicts our assumption. Then we must have $V^- = \emptyset$, and so $V^+ = \emptyset$, and therefore $div_f = 0$, and thus f is a circulation.

□

The weak MFMC theorem says that the value of a a maximum flow is at least bounded by the capacity of a minimum cut.

Lemma 2.1 (The weak MFMC theorem). *Suppose* G *is a directed graph with capacity function* c *and vertices* $s, t \in V$*. Then the value of a maximum flow is bounded by the capacity of a minimum cut.*

Proof of the weak MFMC theorem. Let f be an st-flow and $K = \delta^+(S)$ be an st-cut. Then

$$val(f) = \sum_{v \in S} \left(\sum_{e \in \delta^+(v)} x(e) - \sum_{e \in \delta^-(v)} x(e) \right) = \sum_{e \in \delta^+(S)} x(e) - \sum_{e \in \delta^-(S)} x(e)$$

$$\leq \sum_{e\in\delta^+(S)} c(e) - \sum_{e\in\delta^-(S)} c(e) \leq \sum_{e\in\delta^+(S)} c(e) = cap_c(K)$$

where the first equality follows from flow conservation in $S \setminus \{s\}$. By taking the maximum over all st-flows and the minimum over all st-cuts, we obtain the desired inequality. □

We can now prove the MFMC theorem.

Proof of the MFMC theorem. Let M denote the capacity of a minimum cut. By the weak MFMC theorem, the value of a maximum flow is bounded by M, and we will show that there exists an st-flow with value equal to M. Let G' be the graph obtained from G by adding the edge (t, s), if it is not already there. We will use Hoffman's circulation theorem on G'. Define $l(t, s) = u(t, s) = M$, and, for each $e \in E$, let $l(e) = 0$ and $u(e) = c(e)$. If $s \in S$ and $t \in S$, then

$$\sum_{e\in\delta^-(S)} l(e) = M \leq M = \sum_{e\in\delta^+(S)} u(e)$$

which is obviously satisfied. If $s \in S$ and $t \notin S$, then

$$\sum_{e\in\delta^-(S)} l(e) = M \leq \sum_{e\in\delta^+(S)} u(e) = \sum_{e\in\delta^+(S)} c(e) = cap_c(\delta{+}(S))$$

which is satisfied, since M is the capacity of a minimum cut. In either case, Hoffman's circulation theorem implies that there exists a circulation f such that $l \leq f \leq u$, so $f(t, s) = l(t, s) = u(t, s) = M$. The restriction of f from G' to G is an st-flow with value equal to M, which is the capacity of a minimum cut. □

Remark. *The MFMC theorem also follows from the linear programming duality theorem (Theorem 1.2), since the max-flow linear programming problem and the min-cut linear programming problem are primal-dual problems that the theorem says have solutions with the same value, which was how Dantzig and Fulkerson proved the MFMC theorem in [7].*

2.3 Applications of the max-flow min-cut theorem

Menger's theorem says that the maximum amount of pairwise disjoint paths between two vertices in a graph is equal to the minimum amount of edges whose removal seperates the two vertices. König's theorem says that the amount of edges in a maximum matching of a bipartite graph is equal to the amount of vertices in a minimum vertex cover.

2.3.1 Menger's theorem

We only consider the edge-version of Menger's theorem for undirected graphs, while several other versions exist.

Theorem 2.3 (Menger's theorem)**.** *Suppose G is a finite undirected graph with distinct vertices $x, y \in V$. Then the maximum amount of pairwise edge-disjoint paths between x and y is equal to the minimum amount of edges whose removal seperates x and y.*

We need to formulate the following lemma before we can prove Menger's theorem using the MFMC theorem.

Notation. *Let G' denote the directed version of a undirected graph G obtained by replacing each undirected edge $\{u, v\} \in E$ with two directed edges (u, v) and (v, u).*

Lemma 2.2. *Suppose G is a finite undirected graph with distinct vertices $x, y \in V$. Then the amount of pairwise edge-disjoint paths between x and y in G and G' are the same.*

Proof of Lemma 2.2. There are at least as many pairwise edge-disjoint xy-paths in G' as there are in G, since if two xy-paths are edge-disjoint in G, then they are also edge-disjoint in G'.

We will show that there are at most as many pairwise edge-disjoint xy-paths in G' as there are in G. Let P_1 and P_2 be two edge-disjoint xy-paths in G'. If there is no edge $(a, b) \in E'$ such that $(a, b) \in P_1$ and $(b, a) \in P_2$, then the two paths are edge-disjoint in G too. If there exists such an edge, then we can transform the edge-joint xy-paths P_1 and P_2 into edge-disjoint xy-paths $P_1^{'}$ and $P_2^{'}$ in the following way: Let $P_1^{'}$ be the concatenation of the xa-path in P_1 and the ay-path in P_2; and let $P_2^{'}$ be the concatenation of the xb-path in P_2 and the by-path in P_1. By applying this transformation to each pair of edge-joint xy-paths in G', we obtain corresponding pairs of xy-paths that are edge-disjoint in G' as well as in G.

Since there are both at least and at most as many pairwise edge-disjoint xy-paths in G' as there are in G, the amounts must be equal. □

We can now prove Menger's theorem.

Proof of Menger's theorem. Let c be the capacity function that assigns unit capacity to all edges of G'. Then, by the max-flow min-cut theorem, the value of a maximum flow from x to y in G' is equal to the capacity of a minimum cut.

The value of a maximum flow is equal to the maximum amount of pairwise edge-disjoint xy-paths in G', since each unit value of flow must pass through its own xy-path when each edge has unit capacity. Furthermore, the amount of pairwise edge-disjoint xy-paths in G and G' are the same, by Lemma 2.2.

The capacity of a minimum cut in G' is equal to the cardinality of a minimum cut, since each edge has unit capacity, which in turn is equal to the minimum amount of edges whose removal disconnects x and y, by definition of a minimal cut. Furthermore, the corresponding minimum cut in G can clearly not contain more edges than the one in G', but it can neither contain fewer edges, since there must be at least one edge for each pairwise edge-disjoint xy-path in G.

Thus the maximum amount of pairwise edge-independent paths between x and y is equal to the minimum amount of edges whose removal disconnects x and y. □

2.3.2 König's theorem

We need to define bipartite graphs, matchings and vertex covers before we can formulate König's theorem.

Definition 2.15 (Bipartite graph). *A **bipartite graph** is a graph G whose vertices can be partioned into two disjoint sets such that there is no edge between vertices within each subset.*

Definition 2.16 (Matching, maximal). *For any bipartite graph G with vertex partition (X, Y), a **matching** is a set of pairwise vertex-disjoint edges $(x, y) \in E$ such that $x \in X$ and $y \in Y$. A matching is **maximal** if adding any edge yields a non-mathcing.*

Definition 2.17 (Vertex cover, minimal). *For any graph G, a **vertex cover** is a subset $W \subset V$ such that every edge in E has at least one vertex in W. A vertex cover is **minimal** if removing any vertex yields a non-vertex cover.*

We can now formulate König's theorem.

Theorem 2.4 (König's theorem). *Suppose G is a finite bipartite graph. Then the amount of edges in a maximal matching is equal to the amount of edges in a minimal vertex cover.*

Proof of König's theorem. Let G' be the graph obtained by adding two vertices s, t to V, and adding edges from s to each vertex in the first vertex cell X and edges from t to each vertex in the second vertex cell Y. Let c be the capacity function that assigns infinite capacity to the original edges and unit capacity to the added edges. Then, by the max-flow min-cut theorem, the value of a maximum flow from s to t in G' is equal to the capacity of a minimum cut.

For each matching in G with cardinality k, there is an integer flow in G' with value k, having unit flow along the paths (s, x, y, t), where (x, y) is an edge in the matching, and zero flow everywhere else. For each integer flow in G' with value k, there is a matching in G with cardinality k, consisting of the edges (x, y) that have non-zero flow. Thus the value of a maximum flow in G' is equal to the cardinality of a maximum covering in G.

For each vertex cover in G with cardinality k, let $W_X = W \cap X$ and $W_Y = W \cap Y$, let $X' = X \setminus W_X$ and $Y' = Y \setminus W_Y$, and let $S = s \cup W_Y \cup X'$ and $T = t \cup W_X \cup Y'$. There are no edges between X' and Y', since W covers G, so the cardinality/capacity of the cut $K = \delta^+(S)$ is k. For each cut $K = \delta^+(S)$ with finite cardinality/capacity k, each edge in K must be between s and X or Y and t, which have unit capacity, since edges between X and Y have infinite capacity. Then $W = \{x \in X : (s, x) \in S\} \cup \{y \in Y : (y, t) \in E \setminus S\}$ is a matching in G with cardinality k. Thus the capacity of a minimum cut in G' is equal to the cardinality of a minimum vertex cover in G.

Thus the cardinality of a maximum matching is equal to the cardinality of a minimum vertex cover. $\square$

Chapter 3

A generalized max-flow min-cut theorem

In this chapter, we review a generalization of Chapter 2's max-flow min-cut (MFMC) theorem from digraphs with numerical edge weights to digraphs with semimodule-valued edge weights, which are represented as partially ordered topological spaces with sheaves of partial semimodules over semirings. The generalized MFMC theorem can be used to solve optimization problems that are expressed as max-flow problems with semimodule-valued edge weights, like vectors, probability distributions and logical statements. The definitions, results and proofs are from Sanjeevi Krishnan's *Flow-cut dualities for sheaves on graphs* [8], but I have added structure and details to all the proofs, though they are otherwise unchanged, and developed examples of many of the concepts to better understand them.

In the first section, we define partial semimodules, digraphs represented as partially ordered topological spaces and sheaves of partial semimodules over semirings on digraphs. In the second section, we consider directed sheaf cohomology and homology with sheaves on digraphs as coefficients, orientation sheaves from local directed sheaf homology and a directed sheaf (co)homology duality. In the third section, we relate first directed sheaf cohomology and first directed sheaf homology to cut values and flows, respectively, and apply the directed sheaf (co)homology duality to obtain a generalized MFMC theorem. In the fourth and last section, we consider an application of the generalized MFMC theorem to logical statements.

3.1 Sheaves of partial semimodules on digraphs

Sheaves of partial semimodules on digraphs are assignments of objects from the category of partial semimodules to the vertices and edges of a digraph, which generalize numerical edge weights to semimodule-valued edge weights, and which will be used as coefficients of the directed sheaf (co)homology in the next section. Semimodules over semirings are useful because they can encode both ordinary numerical objects, like natural numbers, and more special types of objects, like vectors, probability distributions and logical propositions.

3.1.1 Partial semimodules

We need to define semirings and semimodules over semirings before we can define partial semimodules over semirings and generalize numerical edge weights to semimodule-valued edge weights.

A semiring is a generalization of a ring, whose elements are not required to have an additive inverse. Thus semirings are always commutative monoids, but not always abelian groups, like rings are.

Definition 3.1 (Semiring)**.** *A **semiring** is a set S with distrinct elements $0, 1 \in S$ and two associative operations $+_S, \times_S : S \times S \to S$ such that, for all $x, y, z \in S$,*

i) $x \times_S (y +_S z) = (x \times_S y) +_S (x \times_S z)$

ii) $0 \times_S x = 0$

iii) $0 +_S x = x$

iv) $x +_S y = y +_S x$

v) $1 \times_S x = x$

vi) $x \times_S 1 = x$

Example 3.1. *The extended non-negative natural numbers $\overline{\mathbb{N}_+} = \mathbb{N}_+ \cup \{\infty\}$ under addition and multiplication is a semiring, but not a ring, since not all elements have an additive inverse.*

A semimodule over a semiring is a generalization of a module over a ring, whose elements are not required to have an additive inverse. Thus semimodules over semirings are commutative monoids, but not always abelian groups, like modules over rings are.

Definition 3.2 (Semimodule)**.** *An **S-semimodule** M on a semiring S is a module over S.*

Example 3.2. *The module $\overline{\mathbb{N}_+^n}$ over the semiring $\overline{\mathbb{N}_+}$ with scalar multiplication is an $\overline{\mathbb{N}_+}$-semimodule, but not a proper module, since not all elements have an additive inverse.*

The set of all semimodules over a semiring together with certain homomorphisms between them form a closed monoidal category, where the tensor product is the categorical sum in the category of such semimodules.

Definition 3.3. *Let $\mathscr{M}_S = (\mathscr{M}_S, \otimes_S, S)$ denote the closed monoidal category of S-semimodules and S-homomorphisms between them whose closed structure $\hom_S(M, N)$ sends a pair (M, N) of S-semimodules to the S-semimodule of S-homomorphisms $M \to N$ with addition and scalar multiplication defined pointwise.*

A semimodule can have the property of flatness, which is necessary for many of the results about directed sheaf (co)homology with sheaves of semimodules on digraphs as coefficients.

Definition 3.4 (Flat). *An S-semimodule M is **flat** if*

$$- \otimes_S M : \mathscr{M}_S \to \mathscr{M}_S$$

preserves equalizer diagrams.

Example 3.3. *The module $\overline{\mathbb{N}_+}$ over the semiring $\overline{\mathbb{N}_+}$ is flat, while the module $\overline{\mathbb{Z}}$ over the semiring $\overline{\mathbb{N}_+}$ is not flat.*

The natural preorder on a semimodule can be used to describe some of its algebraic structure.

Definition 3.5 (Natural preorder). *The **natural preorder** on an S-semimodule M is the preorder $\leq_M$ on the underlying set of M such that $x \leq_M \lambda x + y$ for $x, y \in M$ and $\lambda \in S \setminus 0$.*

Example 3.4. *The natural preorder $\leq_{\overline{\mathbb{N}_+}}$ of the $\overline{\mathbb{N}_+}$-semimodule $\overline{\mathbb{N}_+^n}$ is the ordinary vector preorder $\leq$ on $\overline{\mathbb{N}_+^n}$.*

The additive ideals of a semimodule are the subsets that absorb all elements of the semimodule and their multiples under addition.

Definition 3.6 (Additive ideal). *An **additive ideal** in an S-semimodule M is a subset $I \subset M$ such that $(\lambda \times_M x) +_M y \in I$ for all $x \in M, y \in I$ and $0 \neq \lambda \in S$.*

Example 3.5. *The additive ideals of the $\overline{\mathbb{N}_+}$-semimodule $\overline{\mathbb{N}_+^n}$ are the quotients $\{0, \ldots, \infty\}^n / \{c+1, \ldots, \infty\}^n$ that identify all natural numbers greater than c with ∞.*

A semimodule can have two properties with respect to the natural preorder called naturally complete and naturally inf-semilattice ordered.

Definition 3.7. *An S-semimodule M is **naturally complete** if it contains all its unique infima and unique suprema with respect to the natural preorder and **naturally inf-semilattice ordered** if, for every pair $x, y \in M$, there is a unique greatest lower bound $x \wedge y$ with respect to the natural preorder and $x \wedge (y +_M z) = (x \wedge y) +_M z$ for all $x, y, z \in M$.*

Example 3.6. *The $\overline{\mathbb{N}_+}$-semimodule $\overline{\mathbb{N}_+^n}$ is naturally complete, since it contains all its unique infima and unique suprema with respect to the natural preorder, and naturally inf-semilattice ordered, since, for every pair $x, y \in \overline{\mathbb{N}_+^n}$, there is a unique greatest lower bound $x \wedge y \in \overline{\mathbb{N}_+^n}$ with respect to the natural preorder and $x \wedge (y + z) = (x \wedge y) + z$ for all $x, y, z \in \overline{\mathbb{N}_+^n}$. Thus $\overline{\mathbb{N}_+^n}$ is naturally complete inf-semilattice ordered.*

We can now define partial semimodules over semirings, which are generalizations of semimodules over semirings, whose operations are not required to be more than partially defined functions. Thus all semimodules are partial semimodules, but not all partial semimodules are semimodules. Furthermore, partial semimodules over semirings are always commutative monoids, but not always abelian groups, like modules over rings are.

Definition 3.8 (Partial semimodule). *A **partial S-semimodule** over a semiring S is a set M with a distinct element $0 \in M$ and two partial functions $+_M : M \times M \rightharpoonup M$ and $\times_M : S \times M \rightharpoonup M$ such that, for all $x, y, z, m \in M$ and $\lambda, \lambda_1, \lambda_2 \in S$, one side exists whenever the other side exists in the following equations* *i)* $0 +_M m = m$

ii) $1 \times_M m = m$

iii) $(x +_M y) +_M z = x +_M (y +_M z)$

iv) $x +_M y = y +_M x$

v) $(\lambda_1 +_S \lambda_2) \times_M x = (\lambda_1 \times_m x) +_M (\lambda_2 \times_M x)$

vi) $(\lambda_1 \times_S \lambda_2) \times_M x = \lambda_1 \times_M (\lambda_2 \times_M x)$

vii) $\lambda \times_M (x +_M y) = (\lambda \times_M x) +_M (\lambda \times_M y)$

viii) $0 \times_M m = 0$

ix) $\lambda \times_M 0 = 0$

Example 3.7. *The $\overline{\mathbb{N}_+}$-semimodule $\overline{\mathbb{N}_+^n}$ is also a partial $\overline{\mathbb{N}_+}$-semimodule, since all semimodules are partial semimodules.*

The set of all partial semimodules over a semiring together with certain homomorphisms between them also form a category.

Definition 3.9. *Let $\hat{\mathscr{M}}_S$ denote the category of partial S-semimodules and partial S-homomorphisms of the form $\psi : A \to B$ from a partial S-semimodule A to a partial S-semimodule B such that $\psi(0) = 0$ and the following equation holds whenever the left side exists:*

$$\psi((\lambda_1 \times_A x_1) +_A (\lambda_2 \times_A x_2)) = (\lambda_1 \times_B \psi(x_1)) +_B (\lambda_2 \times_B \psi(x_2))$$

A partial subsemimodule is a partial semimodule that is contained in another partial semimodule.

Definition 3.10 (Partial subsemimodule)**.** *A **partial S-subsemimodule** A of a partial S-semimodule B is an S-semimodule such that $A \subset B$ and addition and scalar multiplication on A are restrictions and corestrictions of addition and scalar multiplication on B.*

Example 3.8. *The additive ideals $\{0, \ldots, \infty\}^n / \{c+1, \ldots, \infty\}^n$ of the partial $\overline{\mathbb{N}_+}$-semimodule $\overline{\mathbb{N}_+^n}$ are partial $\overline{\mathbb{N}_+}$-subsemimodules, since they are contained in $\overline{\mathbb{N}_+^n}$ and their operations are restrictions and corestrictions.*

The direct sum of partial semimodules is also a partial semimodule.

Definition 3.11 (Direct sum)**.** *The **direct sum***

$$\oplus_{i \in \mathcal{I}} M_i$$

is the partial S-semimodule natural in an $\mathcal{I}$-indexed collection $\{M_I\}_{i \in \mathcal{I}}$ of partial S-semimodules, whose set consists of the elements of the Cartesian product of underlying sets whose projections onto all but finitely many factors are 0, and whose addition and scalar multiplication are defined coordinate-wise.

3.1.2 Digraphs

We need to define digraphs as partially ordered topological spaces before we can equip them with sheaves of partial semimodules over semirings that generalize numerical edge weights to semimodule-valued edge weights.

A digraph represented as a partially ordered topological space is a digraph whose vertices are ordered under the edges they are incident to.

Definition 3.12 (Digraph)**.** *A **digraph** $X = (V_X, E_X, \partial_-, \partial_+, \trianglelefteq)$ consists of*

*i) a set V_X of **vertices***

*ii) a set $E_X \subseteq V_X \times V_X$ of **edges***

iii) a preorder $\trianglelefteq$ such that the disjoint union $X = V_X \coprod E_X$ is partially ordered such that $v \trianglelefteq e$ if $e \in E_X$ and $v = \partial_- e$ or $v = \partial_+ e$

*iv) partial **source** and **target** functions $\partial_-, \partial_+ : E_X \trianglelefteq V_X$ such that $\partial_-((u,v)) = u$ and $\partial_+((u,v)) = v$ for all edges $(u,v) \in E_X$, where $u, v \in V_X$*

*A subset $C \subset X = (V_X, E_X)$ is **open** if $e \in C$ whenever $v \trianglelefteq e$ and $v \in C$; and a subset $C \subset X = (V_X, E_X)$ is **closed** if $v \in C$ whenever $v \trianglelefteq e$ and $e \in C$; and the **closure** $\langle C \rangle$ of a subset $C \subseteq X = (V_X, E_X)$ is the set $\langle C \rangle = C \cup \partial_+(C \cap E_X) \cup \partial_-(C \cap E_X)$.*

Remark. *This means that a digraph is a partially ordered topological space called an Alexandrov space.*

Example 3.9. *The digraph X from Example 2.1 in Figure 3.1 has vertices $V_X = (v_1, v_2, v_3, v_4, v_5, v_6)$, edges $E_X = (e_1, e_2, e_3, e_4, e_5, e_6, e_7)$, and a preorder $\trianglelefteq$ with, for instance, $v_1, v_2 \trianglelefteq e_1$, since the partial source and target functions $\partial_-, \partial_+ : E_X \trianglelefteq V_X$ have $\partial_-(e_1) = v_1$ and $\partial_+(e_1) = v_2$. The subset $\{v_1, e_1, e_2\}$ is open, since it contains all the edges $\{e_1, e_2\}$ incident to its one vertex v_1. The subset $\{v_1, v_2, v_3, e_1, e_2\}$ is closed, since it contains all the vertices $\{v_1, v_2, v_3\}$ incident to its two edges $\{e_1, e_2\}$. The closure of the open subset $\{v_1, e_1, e_2\}$ is the closed subset $\langle v_1, e_1, e_2 \rangle = \{v_1, v_2, v_3, e_1, e_2\}$, since it contains all the vertices $\{v_1, v_2, v_3\}$ incident to its two edges $\{e_1, e_2\}$.*

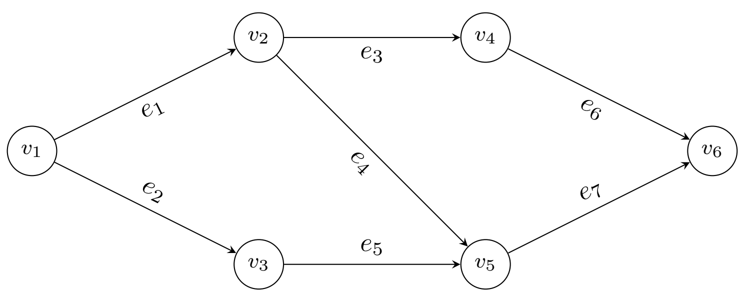

Figure 3.1: A digraph X.

A subdigraph is a digraph contained in another digraph.

Definition 3.13 (Subdigraph)**.** *A **subdigraph** C of a digraph X is a subset C of X considered as a digraph with $E_C = E_X \cap C$, $V_X = V_X \cap C$, and the source and target functions restricted to C.*

Example 3.10. *The closed subset $\{v_1, v_2, v_3, e_1, e_2\}$ of the digraph X in Figure 3.1 with restricted source and target functions is a subdigraph of X.*

A digraph is complete if it contains all the vertices incident to all its edges.

Definition 3.14 (Complete)**.** *A digraph is **complete** if its source and target functions are total functions.*

Example 3.11. *The digraph X in Figure 3.1 is complete, since its source and target functions are total functions, because X contains all the vertices $\{v_1, v_2, v_3, v_4, v_5, v_6\}$ incident to all its edges $\{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$.*

The positive and negative boundary of a digraph are the vertex subsets whose vertices have no edge entering or no edges leaving them, respectively.

Definition 3.15 (Positive boundary, negative boundary)**.** *The **positive boundary** and the **negative boundary** of a complete digraph are the vertex subsets $\partial_- = \partial_- E_X \setminus \partial_+ E_X$ and $\partial_+ = \partial_+ E_X \setminus \partial_- E_X$, respectively.*

Example 3.12. *The positive boundary of the complete digraph in Figure 3.1 is the vertex subset $\partial_- = \{v_1\}$, since no edge enters v_1, while the negative boundary is the vertex subset $\partial_+ = \{v_6\}$, since no edge leaves v_6.*

The in-degree and out-degree of a vertex in a digraph is the amount of edges leaving and entering it, respectively.

Definition 3.16 (In-degree, out-degree)**.** *The **in-degree** and the **out-degree** of a vertex $v \in V_X$ in a digraph are the cardinalities of the sets $\partial_-^{-1}(v)$ and $\partial_+^{-1}(v)$, respectively.*

Example 3.13. *The in-degree of the vertex v_1 in the digraph X in Figure 3.1 is 2, since 2 edges leaves v_1, while the out-degree is 0, since 0 edges enter v_1.*

A digraph is finite if it has a finite amount of vertices and edges, and is locally finite if each vertex is incident to a finite amount of edges.

Definition 3.17 (Finite, locally finite)**.** *A digraph is **finite** if V_X, E_X are finite and **locally finite** if each vertex has finite in-degree and finite out-degree.*

Example 3.14. *The digraph X in Figure 3.1 is finite, since it has a finite amount of vertices and edges, and locally finite, since each vertex in X is incident to a finite amount of edges.*

A digraph is compact if each edge has a source and target vertex.

Definition 3.18 (Compact)**.** *A digraph is **compact** if its source and target functions are total functions and X is finite.*

Remark. *All compact graphs are complete, since their source and target functions are total functions, but not all complete digraphs are compact, since a graph can be complete but not finite, and thus not compact.*

Example 3.15. *The complete digraph X in Figure 3.1 is compact, since its source and target functions are total functions and X is finite.*

A directed loop in a digraph is a subset whose elements form a path that starts and ends at the same vertex, while a simple directed loop is a directed loop that contains no other directed loops.

Definition 3.19 (Directed loop, simple)**.** *A* ***directed loop*** *in a digraph X is a compact subset C of X such that every vertex in C has in-degree and out-degree 1. A directed loop is called* ***simple*** *if there is no compact proper subset D of C that is a directed loop.*

Example 3.16. *The digraph X in Figure 3.1 contains no directed loops, since every compact subset has at least one vertex with in-degree or out-degree that is not 1. The subset $\{v_1, v_2, v_3, e_1, e_2, e_3\}$ of the digraph Y in Figure 3.2 is a directed loop, since it is compact and each of its vertices has in-degree and out-degree 1, and is also a simple directed loop, since it contains no compact proper subset that is a directed loop.*

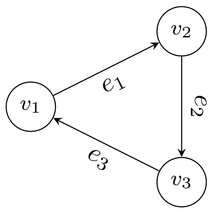

Figure 3.2: A digraph Y with a simple directed loop.

A directed acylic digraph is a digraph that contains no directed loops.

Definition 3.20 (Directed acyclic)**.** *A digraph is* ***directed acyclic*** *if it contains no directed loops.*

Example 3.17. *The digraph X in Figure 3.1 is directed acyclic, since it contains no directed loops.*

The subdivision of a digraph is another digraph obtained by subdividing each edge into two edges and replacing it with a vertex that the two edges enter and leave, respectively.

Definition 3.21 (Subdivision)**.** *The* ***subdivision*** *of a digraph X is the digraph $X_{sd} = (V_{sd,X}, E_{sd,X})$, where $V_{sd,X} = X$, $E_{sd,X} = \{e_- : e \in E_X\} \cup \{e_+ : e \in E_X\}$ and, for each $e \in E_X$, $\partial_- e_- = \partial_- e$, $\partial_+ e_+ = \partial_+ e$ and $\partial_+ e_- = \partial_- e_+ = e$.*

Example 3.18. *The digraph Z in Figure 3.3 has the subdivision Z_{sdZ}, where $V_{sdZ} = Z = \{v_1, v_2, e_1\}$ and $E_{sdZ} = \{e_{1,-}, e_{1,+}\}$.*

Figure 3.3: A digraph Z (left) and its subdivision $Z_{\mathrm{sd}Z}$ (right).

A weighted digraph is a digraph with edge weights in a commutative monoid.

Definition 3.22 (Weighted digraph)**.** *A **weighted digraph** $(X;\omega_M)$ is a digraph X together with a collection $\omega_M = \{\omega_e\}_{e \in E_X}$ of additive ideals $\omega_e \in M$ from a commutative monoid M.*

Example 3.19. *The weighted digraph $(X;\omega_{\overline{\mathbb{N}_+}})$ in Figure 3.4 consists of the digraph X together with the collection $\omega_{\overline{\mathbb{N}_+}} = \{\omega_{e_1}, \omega_{e_2}, \omega_{e_3}, \omega_{e_4}, \omega_{e_5}, \omega_{e_6}, \omega_{e_7}\}$ of additive ideals $\omega_{e_1} = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\}$, $\omega_{e_2} = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\}$, $\omega_{e_3} = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\}$, $\omega_{e_4} = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\}$, $\omega_{e_5} = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\}$, $\omega_{e_6} = \overline{\mathbb{N}_+}/\{2,\ldots,\infty\}$, $\omega_{e_17} = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\}$ from the commutative monoid $\overline{\mathbb{N}_+}$, where the edge weights satisfy the edge capacity constraints $c(e_1) = 3$, $c(e_2) = 2$, $c(e_3) = 2$, $c(e_4) = 2$, $c(e_5) = 3$, $c(e_6) = 1$, $c(e_7) = 3$ on the same digraph in Example 2.1.*

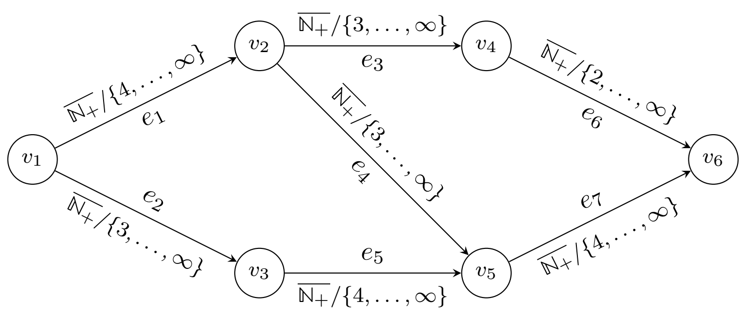

Figure 3.4: A weighted digraph $(X;\omega_{\overline{\mathbb{N}_+}})$.

3.1.3 Sheaves of partial semimodules on digraphs

We need to define cellular sheaves on digraphs and sheaves of partial semimodules on digraphs before be can define constant sheaves of partial semimodules on digraphs, which we will equip digraphs with to generalize numerical edge weights to semimodule-valued edge weights.

A cellular sheaf on a digraph is an assignment of objects from a category to the vertices and edges of the digraph.

Definition 3.23 (Cellular sheaf)**.** *Suppose X is a digraph and $\mathscr{C}$ is a category. A **cellular sheaf** $\mathcal{F}$ on X with values in $\mathscr{C}$ consists of*

i) for each vertex v and for each edge e, assignments of $\mathscr{C}$-objects $\mathcal{F}(v)$ to v and $\mathcal{F}(e)$ to e, which are sets

ii) for each vertex v and for each edge e such that $e \trianglelefteq v$, assignments of restriction maps $\mathcal{F}(v \trianglelefteq e) : \mathcal{F}(v) \to \mathcal{F}(e)$, which are $\mathscr{C}$-morphisms

The category of such sheaves is denoted $Sh_{X;\mathscr{C}}$.

A sheaf of (partial) semimodules on a digraph is a cellular sheaf of (partial) semimodules on the digraph, which is an assignment of objects from the category of (partial) semimodules to the vertices and edges of the digraph.

Definition 3.24 ((Partial) S-sheaf)**.** *An **S-sheaf** on X is an object in $Sh_{X;\mathscr{M}_S}$. A **partial S-sheaf** on X is an object in $Sh_{X;\hat{\mathscr{M}}_S}$.*

Remark. *A partial S-sheaf $\mathcal{A}$ is a partial S-subsheaf of an S-sheaf $\mathcal{B}$ such that $\mathcal{A}(c)$ is a partial S-subsemimodule of $\mathcal{B}(c)$ for each $c \in C \subset X$ and objectwise inclusion defines a natural transformation $\mathcal{A} \to \mathcal{B}$.*

The pullback functor from the category of sheaves on a digraph to the category of sheaves on the subsets of the digraph is the the functor that gives the sheaf on a subset of the digraph that corresponds to a sheaf on the whole digraph.

Definition 3.25 (Pullback functor)**.** *For each subset $C \subseteq X$), the **pullback functor** $(C \subseteq X)^* : \mathrm{Sh}_{X;\mathscr{C}} \to \mathrm{Sh}_{C;\mathscr{C}}$ is defined on objects $\mathcal{F}$ as restrictions.*

The pushforward functor from the category of sheaves on subsets of a digraph to the category of sheaves on the whole digraph is the functor that gives the sheaf on the whole digraph that corresponds to a sheaf on a subset of the digraph..

Definition 3.26 (Pushforward functor)**.** *For each subset $C \subseteq X$), the **pushforward functor** $(C \subseteq X)_* : \mathrm{Sh}_{C;\hat{\mathscr{M}}_S} \to \mathrm{Sh}_{X;\hat{\mathscr{M}}_S}$ is naturally defined on objects $\mathcal{F}$ by*

i) *for each $c \in C$, $(C \subseteq X)_*\mathcal{F}(c) = \mathcal{F}(c)$*

ii) *for each $c \in X \setminus C$, $(C \subseteq X)_*\mathcal{F}(c) = 0$*

iii) *for each edge-vertex pair $e, v \in C$ such that $v \trianglelefteq e$, $(C \subseteq X)_*\mathcal{F}(v \trianglelefteq E) = \mathcal{F}(v \trianglelefteq e)$*

The subdivision functor from the category of sheaves on a digraph to the category of sheaves on the subdivision of the digraph is the functor that gives the sheaf on the subdivision of the digraph that corresponds to a sheaf on the digraph.

Definition 3.27 (Subdivision functor)**.** *For each subset $C \subseteq X$), the **subdivision functor** $\mathrm{sd} : \mathrm{Sh}_{X;\mathscr{C}} \to \mathrm{Sh}_{\mathrm{sd}\,X;\mathscr{C}}$ is naturally defined on objects $\mathcal{F}$ by*

i) *for each $c \in X \subseteq V_{sd,X}$, $(\mathrm{sd}\,\mathcal{F})(c) = \mathcal{F}(c)$*

ii) *for each edge $e \in E_X \subseteq V_{\mathrm{sd},X}$, $(\mathrm{sd}\,\mathcal{F})(e_\pm) = \mathcal{F}(e)$*

iii) $(\mathrm{sd}\,\mathcal{F})(v \trianglelefteq e_\pm) = \mathcal{F}(v \trianglelefteq e)$

iv) $(\mathrm{sd}\,\mathcal{F})(e \trianglelefteq e_\pm) = 1_{\mathcal{F}(e)}$

Remark. *The action of $\mathscr{M}_S$ on $\hat{\mathscr{M}}_S$ defines an objectwise action*

$$\otimes_S : \mathrm{Sh}_{X;\mathscr{M}_S} \times \mathrm{Sh}_{X;\hat{\mathscr{M}}_S} \to \mathrm{Sh}_{X;\hat{\mathscr{M}}_S}$$

The direct sum operation $\oplus$ on $\hat{\mathscr{M}}_S$ defines an objectwise direct sum operation

$$\oplus_S : \mathrm{Sh}_{X;\hat{\mathscr{M}}_S} \times \mathrm{Sh}_{X;\hat{\mathscr{M}}_S} \to \mathrm{Sh}_{X;\hat{\mathscr{M}}_S}$$

A sheaf of partial semimodules on digraphs can have the property of flatness, which is necessary for many of the results about directed sheaf (co)homology with sheaves of partial semimodules on digraphs as coefficients.

Definition 3.28. *A partial S-sheaf on a digraph X is **flat** if the functor*

$$- \otimes_S \mathcal{F} : \mathrm{Sh}_{X;\mathscr{M}_S} \to \mathrm{Sh}_{X;\hat{\mathscr{M}}_S}$$

preserves equalizers.

Remark. *A sheaf of partial semimodules on a digraph is flat if its objectwise flat, while the edge weights of a weighted digraph are flat if they take values in a flat semimodule.*

A naturally inf-semilattice ordered sheaf of partial semimodules on a digraph is a sheaf of partial semimodules on the graph that is objectwise naturally inf-semilattice ordered and satisfies a condition concerning greater lower bounds.

Definition 3.29. *A partial S-sheaf $\mathcal{F}$ is called **naturally inf-semilattice ordered** if it is objectwise naturally inf-semilattice ordered and the restriction maps between cells of $\mathcal{F}$ preserves greatest lower bounds of finite subsets with respect to natural preorders.*

We can now define constant sheaves of partial semimodules on digraphs.

A constant sheaf of a partial semimodule on a digraph is a sheaf of the partial semimodules on the digraph, which is an assignment of objects from the partial semimodule to the vertices and edges of the digraph, and which digraphs will be equipped with to generalize numerical edge weights to semimodule-valued edge weights.

Definition 3.30 (Constant sheaf)**.** *The **constant sheaf** at M is the S-sheaf k_M that is constant on a partial S-semimodule M.*

We can now re-define weighted digraphs, by equiping digraphs with constant sheaves of partial semimodules that generalize numerical edge weights to semimodule-valued edge weights.

Definition 3.31 (Weighted digraph)**.** *A weighted digraph $(X;\omega_{k_M})$ is a weighted digraph $(X;\omega_M)$ whose edge weights are identified with the partial S-subsheaf on X of the constant partial S-sheaf k_M on X defined by*

i) for each edge $e \in E_X$, $\omega_e = \{x \in M : x \leq_M \omega_e\}$

ii) for each vertex $v \in V_X$, $\omega_v = M$

Example 3.20. *The weighted digraph $(X;\omega_{k_{\overline{\mathbb{N}_+}}})$ in Figure 3.5 is the weighted digraph from Example 3.18 with added vertex weights, where the edge and vertex weights are identified with the partial $\overline{\mathbb{N}_+}$-subsheaf of the constant partial $\overline{\mathbb{N}_+}$-*

sheaf $k_{\overline{\mathbb{N}_+}}$ *on* X*, whose sections and restrictions are*

$$
\begin{array}{c}
k_{\overline{\mathbb{N}_+}}(v_1 \trianglelefteq e_1) = k_{\overline{\mathbb{N}_+}}(v_2 \trianglelefteq e_1) = k_{\overline{\mathbb{N}_+}}(e_1) = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_1 \trianglelefteq e_2) = k_{\overline{\mathbb{N}_+}}(v_3 \trianglelefteq e_2) = k_{\overline{\mathbb{N}_+}}(e_2) = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_2 \trianglelefteq e_3) = k_{\overline{\mathbb{N}_+}}(v_4 \trianglelefteq e_3) = k_{\overline{\mathbb{N}_+}}(e_3) = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_2 \trianglelefteq e_4) = k_{\overline{\mathbb{N}_+}}(v_5 \trianglelefteq e_4) = k_{\overline{\mathbb{N}_+}}(e_4) = \overline{\mathbb{N}_+}/\{3,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_3 \trianglelefteq e_5) = k_{\overline{\mathbb{N}_+}}(v_5 \trianglelefteq e_5) = k_{\overline{\mathbb{N}_+}}(e_5) = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_4 \trianglelefteq e_6) = k_{\overline{\mathbb{N}_+}}(v_6 \trianglelefteq e_6) = k_{\overline{\mathbb{N}_+}}(e_6) = \overline{\mathbb{N}_+}/\{2,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_5 \trianglelefteq e_7) = k_{\overline{\mathbb{N}_+}}(v_6 \trianglelefteq e_7) = k_{\overline{\mathbb{N}_+}}(e_7) = \overline{\mathbb{N}_+}/\{4,\ldots,\infty\} \\
k_{\overline{\mathbb{N}_+}}(v_1) = \overline{\mathbb{N}_+} \\
k_{\overline{\mathbb{N}_+}}(v_2) = \overline{\mathbb{N}_+} \\
k_{\overline{\mathbb{N}_+}}(v_3) = \overline{\mathbb{N}_+} \\
k_{\overline{\mathbb{N}_+}}(v_4) = \overline{\mathbb{N}_+} \\
k_{\overline{\mathbb{N}_+}}(v_5) = \overline{\mathbb{N}_+} \\
k_{\overline{\mathbb{N}_+}}(v_6) = \overline{\mathbb{N}_+}
\end{array}
$$

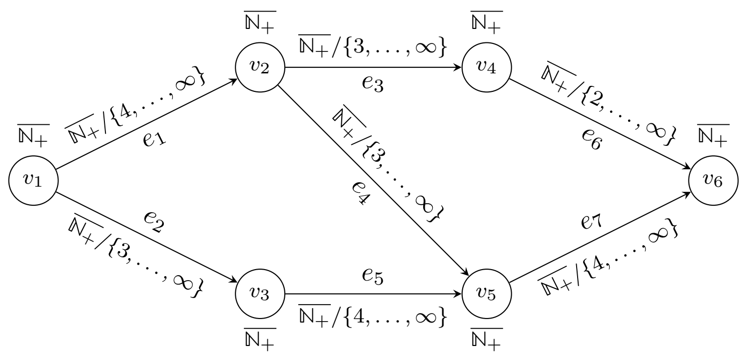

Figure 3.5: A weighted digraph $(X; \omega_{k_{\overline{\mathbb{N}_+}}})$.

3.2 Directed (co)homology with sheaves on digraphs as coefficients

Directed sheaf (co)homology is a generalization of Abelian sheaf (co)homology for semimodule-valued sheaves over digraphs represented as partially preordered topological spaces. Directed sheaf (co)homology is used instead of Abelian sheaf (co)homology, because Abelian sheaf (co)homology is indiscriminate to edge orientations in digraphs, while directed sheaf (co)homology is not. This section is quite theoretical, so we refer to the fourth and last section for an example that demonstrates the utility of the concepts.

3.2.1 Directed sheaf cohomology

Zeroth and first directed sheaf cohomology are defined as functors H^0_c, H^1_c that equalize and coequalize, respectively, coboundary operators from sheaf-valued 0-cochains to sheaf-valued 1-cochains.

We need to define a preliminary directed sheaf cohomology H^0, H^1 before we can define a compactly supported directed sheaf cohomology H^0_c, H^1_c.

Definition 3.32. *Let $H^\bullet(X;\mathcal{F})$ be defined by the equalizer and coequalizer diagrams*

$$H^0(X;\mathcal{F}) \dashrightarrow \oplus_{v\in V_X}\mathcal{F}(v) \underset{\oplus_{\partial_+ e=v}\mathcal{F}(v\trianglelefteq e)}{\overset{\oplus_{\partial_- e=v}\mathcal{F}(v\trianglelefteq e)}{\rightrightarrows}} \oplus_{e\in E_X}\mathcal{F}(e) \dashrightarrow H^1(X;\mathcal{F})$$

natural in partial S-sheaves $\mathcal{F}$ on compact digraphs X.

Remark. *Inclusions $A\subset B\subset X$ of digraphs induce inclusions between direct sums of stalks and thus induce partial S-homomorphisms*

$$H^0(A;(A\subset X)^*\mathcal{F}) \to H^0(B;(B\subset X)^*\mathcal{F})$$

natural in partial S-sheaves $\mathcal{F}$ on digraphs X.

We can now define a compactly supported directed sheaf cohomology.

Definition 3.33. *Let $H^\bullet_c(X;\mathcal{F})$ be defined by*

$$H^\bullet_c(X;\mathcal{F}) = \operatorname{colim}_{K\subset X} H^\bullet(K;(K\subset X)^*\mathcal{F})$$

natural in partial S-sheaves $\mathcal{F}$ on digraphs X, where $H^\bullet(K;\mathcal{F})$ is considered as a covariant functor in K and the colimit is over all compact subdigraphs $K\subset X$.

Zeroth relative cohomology is defined as an absolute cohomology.

Definition 3.34. *Suppose $\mathcal{F}$ is a sheaf of a S-semimodule on a digraph X. For each subset C of X on which $\mathcal{F}$ is defined, let $H^0_c(C;\mathcal{F}) = H^0_c(C;(C\subset X)^*\mathcal{F})$. For each pair of subdigraphs $A\subset B\subset X$, let $H^0_c(A\subset B;\mathcal{F})$ denote the induced partial S-homomorphism*

$$H^0_c(A\subset B;\mathcal{F}) : H^0_c(B;\mathcal{F}) \to H^0_c(A;\mathcal{F})$$

There are two connecting homomorphisms from zeroth cohomology to first cohomology, which we need to assign cut values from zeroth cohomology to cuts from first cohomology before we can generalize the MFMC theorem.

Definition 3.35. *Suppose $C\subset X$ is closed. Let the partial S-homomorphisms*

$$\delta_-, \delta_+ : H^0_c(C;\mathcal{F}) \to H^1_c((X,C);\mathcal{F})$$

be defined by each of the two commutative squares of the form

$$\begin{tikzcd}
H^0_c(C;\mathcal{F}) \arrow[r] \arrow[d, "\delta_\pm"'] & \oplus_{v\in V_C}\mathcal{F}(v) \arrow[r] & \oplus_{v\in V_{\partial_\pm C}}\mathcal{F}(v) \arrow[d, "\oplus_v \sum_e \mathcal{F}(v\trianglelefteq e)"] \\
H^1_c((X,C);\mathcal{F}) & \oplus_{e\in X\setminus C}\mathcal{F}(e) \arrow[l] & \oplus_{e\in\partial_\pm C}\mathcal{F}(e) \arrow[l]
\end{tikzcd}$$

where the left horizontal arrows are universal arrows and the right horizontal arrows are induced by projection and inclusion.

We need to define the c-sections of sheaves of partial semimodules on digraphs, c-sectionwise surjections and c-sectionwise coequalizer diagrams before we can formulate a result that we will need to prove the genrealized MFMC theorem.

Definition 3.36 (c-section). *The **c-sections** of a sheaf $\mathcal{F}$ of a partial S-semimodule on a digraph X are the elements in $H_c^0(U;\mathcal{F})$ for each open $U \subseteq X$.*

Definition 3.37. *A **c-sectionwise surjection** is a natural transformation ϵ of sheaves of partial S-semimodules on a digraph X such that $H_c^0(U;\epsilon)$ is a surjection for each open $U \subseteq X$.*

Definition 3.38. *A **c-sectionwise** coequalizer diagram is a diagram*

$$\mathcal{F}_1 \rightrightarrows \mathcal{F}_0 \longrightarrow \mathcal{F}$$

of sheaves of partial S-semimodules on a digraph X such that $H_c^0(U;\epsilon)$ is a surjection for each open $U \subseteq X$.

We can now formulate the result that we will need to prove the generalized MFMC theorem.

Lemma 3.1. *For each sheaf of S-semimodules on a digraph X, the universal natural transformation*

$$\epsilon : \bigoplus_{C \subset X} (C \subset X)_* k_{H_c^0(C;\mathcal{F})} \to \mathcal{F}$$

is a sectionwise surjection.

Proof. For each $B \subset X$, the partial S-homomorphism

$$H_c^0(B;\epsilon) : \bigoplus_{B \subset C} H_c^0(C;\mathcal{F}) \to H_c^0(B;\mathcal{F})$$

is a surjection, since its restriction to the B-indexed summand is the identity. Thus ϵ is a sectionwise surjection. □

3.2.2 Directed sheaf homology

Zeroth and first directed sheaf homology are defined as functors H_0^c, H_1^c that equalize and coequalize, respectively, boundary operators from partial semimodule-valued 1-chains to partial semimodule-valued 0-chains.

Definition 3.39. *Let $H_\bullet^c(X;\mathcal{F})$ be defined by the equalizer and coequalizer diagrams*

$$H_1^c(X;\mathcal{F}) \dashrightarrow \oplus_{e \in E_{sdX}} H_c^0(\langle e \rangle; sd\mathcal{F}) \underset{\oplus_e H_c^0(\partial_+ e \subset \langle e \rangle; sd\mathcal{F})}{\overset{\oplus_e H_c^0(\partial_- e \subset \langle e \rangle; sd\mathcal{F})}{\rightrightarrows}} \oplus_{v \in V_{sdX}} (sd\mathcal{F})(v) \dashrightarrow H_0^c(X;\mathcal{F})$$

natural in direct sums in $Sh_{X;\mathscr{M}_S}$ of pushforward functors of constant partial S-sheaves of projective partial S-semimodules.

Relative first homology is defined as follows.

Definition 3.40. *Let $H_1^c((X,U);\mathcal{F})$ denote the partial S-semimodule*

$$H_1^c((X,U);\mathcal{F}) = H_1^c(X;(X \setminus U)_* k_s \oplus_S \mathcal{F})$$

for a digraph X, an open subset $U \subset X$, and a partial S-sheaf $\mathcal{F}$ on X.

There are two connecting homomorphisms from first homology to zeroth homology, which we need to assign flow values from zeroth homology to flows from first homology before we can generalize the MFMC theorem.

Definition 3.41. *Suppose $C \subset X$ is open. Let the partial S-homomorphisms*

$$\delta_-, \delta_+ : H_1^c((X,U);\mathcal{F}) \to H_0^c(U;\mathcal{F})$$

be defined by each of the two commutative squares of the form

$$\begin{array}{ccccc}
H_1^c(X,U) & \longrightarrow & \oplus_{e \in E_{sd(X\setminus U)}} H_c^0(\langle e \rangle; sd\mathcal{F}) & \longrightarrow & \oplus_{v \in V_{\partial_\pm X \setminus U}} (sd\mathcal{F})(v) \\
{\scriptstyle \delta_\pm}\downarrow & & & & \downarrow \\
H_0^c(U;\mathcal{F}) & \longleftarrow & \oplus_{v \in X_{sdU}} (sd\mathcal{F})(v) & \longleftarrow & \oplus_{\partial_\mp e \in \partial_\pm (X\setminus U)} (sd\mathcal{F})(e)
\end{array}$$

where the left horizontal arrows are universal arrows and the right horizontal arrows are induced by projection and inclusion.

The following result will be used to prove the generalized MFMC theorem.

Lemma 3.2. *For a S-sheaf $\mathcal{F}$ on a digraph X and an open $U \subset X$,*

$$H_1^c(X;\mathcal{F}) \xrightarrow{H_1^c((X,\emptyset)\subset(X,U))} H_1^c((X,U);\mathcal{F}) \underset{\delta_-}{\overset{\delta_+}{\rightrightarrows}} H_0^c(U;\mathcal{F})$$

commutes for $X \setminus U$, U acyclic and is exact if $\mathcal{F}$ is naturally inf-semilattice ordered and the images of restriction maps between cells of $\mathcal{F}$ are down-sets with respect to the natural preorders.

Proof. (Sketch) Consider when X is compact, where the preorder $\leq_U$ on U generated by the relations of the form $\partial_- e \leq_U \partial_+ e$ for each $e \in E_U$ makes U a finite poset, by acyclicity of U. Then the diagram commutes by an inductive argument on the length of a maximal chain in U. □

3.2.3 Orientation sheaves from directed sheaf homology

An orientation sheaf is a local first directed sheaf homology, which generalizes directions, or edge orientations.

Definition 3.42 (Orientation sheaf)**.** *An **orientation sheaf** on a digraph X is the S-sheaf $\mathcal{O}_S$ defined by*

$$\mathcal{O}_S(x) = H_1^c((X, X \setminus x); k_s)$$

The following result will be used to prove the lemma in the next subsection.

Lemma 3.3. *Suppose X is a digraph. For each $v \in V_X$, the diagram*

$$\mathcal{O}_S \otimes_S M \dashrightarrow \oplus_{e \in E_X} (\langle e \rangle \subset X)_* k_S) \otimes_S M \underset{\partial_+ \otimes_S M}{\overset{\partial_- \otimes_S M}{\rightrightarrows}} \oplus_{v \in V_X} ((v \subset X)_* k_S) \otimes_S M$$

where the dotted arrow is induced by the natural inclusions $\mathcal{O}_S \to S$, is an equalizer diagram natural in partial S-semimodules M if M is flat.

Proof. Assume M is flat. Then the diagram is an equalizer diagram, since $M \times_S -$ preserves equalizer diagrams by definition. □

3.2.4 Directed sheaf (co)homology duality

The follownig result formulates a directed sheaf (co)homology duality, which relates zeroth and first directed sheaf (co)homology, and which we will use to prove the MFMC theorem.

Theorem 3.1. *Suppose X is a digraph and $U \subset X$ is open. Then there exists dotted arrows in*

$$\begin{tikzcd}
H^0_c(X \setminus U; \mathcal{O}_S \oplus_S \mathcal{F}) \arrow[r, dotted] \arrow[d, shift right, "\delta_+"'] \arrow[d, shift left, "\delta_-"] & H^c_1((X,U); \mathcal{F}) \arrow[d, shift right, "\delta_+"'] \arrow[d, shift left, "\delta_-"] \\
H^1_c((X, X \setminus U); \mathcal{O}_S \oplus_S \mathcal{F}) \arrow[r, dotted] & H^c_0(U; \mathcal{F})
\end{tikzcd}$$

natural in partial S-sheaves $\mathcal{F}$ on X, such that the diagram jointly commutes. The top arrow is an isomorphism and the bottom arrow is a surjection. The bottom arrow is an isomorphism if S is a ring and each vertex has positive total degree, or if each vertex in X has both positive in-degree and positive out-degree.

Proof. There exists a natural dotted monomorphism

$$\Delta^1_{\mathcal{F}} : H^0_c(X \setminus U; \mathcal{O}_S \otimes_S \mathcal{F}) \cong H^0_c(\mathrm{sd}(X \setminus U); \mathcal{O} \otimes_S \mathrm{sd}\,\mathcal{F}) \cong H^c_1((X,U); \mathcal{F})$$

in the above diagram, which defines an isomorphism if $\mathcal{F}$ is flat, by Lemma 3.4, and so the monomorphism is surjective even if $\mathcal{F}$ is not flat, since objectwise projective partial S-sheaves are flat. The diagrams in

$$\begin{tikzcd}
\oplus_{v \in V_U} \mathcal{O}_S(v) \otimes_S \mathcal{F}(v) \arrow[r, shift left, "\delta_-"] \arrow[r, shift right, "\delta_+"'] \arrow[d] & \oplus_{e \in E_U} \mathcal{O}_S(e) \otimes_S \mathcal{F}(e) \arrow[d] \\
\oplus_{e \in E_{\mathrm{sd}\,U}} H^0_c(\langle e \rangle; \mathcal{F}) \arrow[r, shift left, "\delta_-"] \arrow[r, shift right, "\delta_+"'] & \oplus_{x \in V_{\mathrm{sd}\,U}} (\mathrm{sd}\,\mathcal{F})(x)
\end{tikzcd}$$

where the left vertical arrow is induced by projections and the right vertical arrow is inclusion after identifiying $\mathcal{O}_S(e) \otimes_S \mathcal{F}(e) = \mathcal{F}(e)$ for each $e \in E_X$, commute. Thus the vertical arrows induce an arrow

$$\Delta^0_{\mathcal{F}} : H^0_c(U; \mathcal{O} \otimes_S \mathcal{F}) \to H^c_0(U; \mathcal{O} \otimes_S \mathcal{F})$$

which is surjective, since each element in $H^c_0(U; \mathcal{F})$ is represented by an element in $\oplus_{e \in E_U} (\mathrm{sd}\,\mathcal{F})(e)$. Then the diagram commutes by a diagram chase. □

The following result will be used to prove the next lemma.

Lemma 3.4. *For each compact digraph X, there exists a monic dotted arrow in*

$$H^0_c(X) \oplus_{e \in E_X} \dashrightarrow H^0_c(\langle e \rangle; \mathcal{F}) \underset{\oplus_v \sum_{\partial_- e = v} H^0_c(\partial_+ e \subset \langle e \rangle)}{\overset{\oplus_v \sum_{\partial_- e = v} H^0_c(\partial_- e \subset \langle e \rangle)}{\rightrightarrows}} \oplus_{v \in V_X} \mathcal{F}(v)$$

with $H^\bullet_c = H^\bullet_c(X; \mathcal{O}_S \oplus_S \mathcal{F})$, natural in partial S-sheaves $\mathcal{F}$ on X such that the diagram commutes. Furthermore, the diagram is an equalizer diagram if $\mathcal{F}$ is flat.

Proof. There exists a universal natural transformation i_S from $\mathcal{O}_S$ making the diagram

$$\oplus_{e \in E_X} (\langle e \rangle \subset X)_a stk_S \otimes_S \mathcal{F} \underset{\oplus_e k_{H^0_c(\partial_+ e \hookrightarrow \langle e \rangle)} \otimes_S 1_{\mathcal{F}}}{\overset{\oplus_e k_{H^0_c(\partial_- e \hookrightarrow \langle e \rangle)} \otimes_S 1_{\mathcal{F}}}{\rightrightarrows}} \oplus_{v \in V_X} (v \subset X)_* k_S \otimes_S \mathcal{F}(v)$$

an equalizer diagram for $\mathcal{F} = k_S$. Thus $H^0_c(X; i_S \otimes_S 1_{\mathcal{F}})$ induces a natural arrow to $H^c_1(X; \mathcal{F})$, and so the diagram commutes.

Assume that $\mathcal{F}$ is flat. Then $i_S \otimes_S \mathcal{F}$ equalizes the above diagram, by Lemma 3.3, and induces an arrow equalizing the diagram obtained by applying the equalizer-preserving functor $H^0_c(X; -)$ □

The following result will be used to prove a result that relates flows with first homology such that we can generalize the MFMC theorem.

Lemma 3.5. *For each compact digraph X, there exists a monic dotted arrow in*

$$H^c_1(X; \mathcal{F}) \dashrightarrow \oplus_{e \in E_X} H^0_c(\langle e \rangle; \mathcal{F}) \underset{\oplus_v \sum_{\partial_- e = v} H^0_c(\partial_+ e \subset \langle e \rangle)}{\overset{\oplus_v \sum_{\partial_- e = v} H^0_c(\partial_- e \subset \langle e \rangle)}{\rightrightarrows}} \oplus_{v \in V_X} \mathcal{F}(v)$$

natural in partial S-sheaves $\mathcal{F}$ on X such that the diagram commutes. Furthermore, the diagram is an equalizer diagram if, for each $v \in V_X$, $\mathcal{F}(v)$ is flat.

Proof. There exists a natural isomorphism

$$H^c_1(X; \mathcal{F}) \cong H^0_c(X; \mathcal{O}_S \otimes_S \mathcal{F})$$

by Theorem 3.1, and so the diagram commutes.

Assume that, for each $v \in V_X$, $\mathcal{F}(v)$ is flat. Then the diagram is an equalizer diagram, by Lemma 3.4. □

3.3 A generalized max-flow min-cut theorem

The generalized max-flow min-cut theorem is a generalization from numerical-weighted digraphs to semimodule-weighted digraphs with an additional edge from the target to the source vertex, which enables a decomposition of the values of flows. We need to define cut values from first directed sheaf cohomology, flow values from first directed sheaf homology, and a decomposition of flow values in terms of cut values before we can use all the previous results about directed sheaf (co)homology to formulate and prove the generalized MFMC theorem. This section is quite theoretical, so we refer to the next section for an example that utilizes many of the concepts.

3.3.1 Cut values from first directed sheaf cohomology

The value of an edge subset of a digraph is its total edge weight.

Definition 3.43. *The **value** of an edge subset $C \subset E_X$ in a weighted digraph $(X;\omega)$ is*

$$\sum_{c \in C} \omega_c$$

The following definition and proposition says that the values of edge subsets are contained in first cohomology.

Definition 3.44. *For each partial S-sheaf $\mathcal{F}$ on a digraph X, let*

$$[C]_{e,\mathcal{F}} = \operatorname{Im} H^1_c(C \subset X \setminus e) \circ \delta_- : H^0_c(C;\mathcal{F}) \to H^1_c(X \setminus e;\mathcal{F})$$

Proposition 3.1. *For each edge subset $C \subset X$ in a weighted digraph $(X;\omega)$,*

$$[C]_{c,\mathcal{O}_S \otimes_S \omega} = \sum_{c \in C} \omega_e$$

Notation. *For a pair of vertex subsets $A, B \subset V_X$ in a digrpah X, $A : B$ denotes the set $A : B = (\partial_-^{-1} A \cap \partial_-^{-1} B) \subset E_X$*

An st-cut is a partition of a digraph that seperates the source and the target vertex.

Definition 3.45. *For each pair of distinct vertices s, t in a digraph X, an **st-cut** is a partition $(A, V_X \setminus A)$ of the vertices V_X such that $s \in A$ and $t \notin A$.*

An e-cut is an edge subset that forms an st-cut with its complement, where e is the edge from the target to the source vertex.

Definition 3.46. *For each edge e in a digraph X, an **e-cut** is an edge subset $A : V_X \setminus A \subset E_X$ such that (A, V_X) is a $\partial_+ e \partial_- e$-cut.*

The following lemma will be used to prove a generalized sheaf-theoretical MFMC theorem.

Lemma 3.6. *Suppose $e \in E_X$ is an edge in a digraph X such that $X \setminus e$ is acyclic. Then the following statements are equivalent for finite subsets $C \subset X$*

i) C is an e-cut

ii) The element in $H^1_c(X;k_s)$ represented by $e \in S[e]$ is represented by $\sum_{c \in C} c$ in $S[C]$.

Proof. (Sketch) Consider the digraph $X \setminus e$ as a poset whose partial order $\leq_{X \setminus e}$ is generated by relations of the form $\partial_- e \leq_{X \setminus e} \partial_+$. Then the equivalence of the statements can be proved by induction on the length of maximal chains of $X \setminus e$. □

3.3.2 Flows from first directed sheaf homology

An ω-flow on a weighted digraph is a flow that satisfies edge capacity constraints and vertex flow conservation.

Definition 3.47. *An ω-**flow** on a digraph $(X;\omega)$ is a function*

$$\phi : E_X \to M$$

where M is the commutative monoid of edge weights satisfying the conditions

i) for each edge $e \in E_X$, $\phi(e) \leq_M \omega_e$

ii) for each vertex $v \in V_X$, $\sum_{e\in\partial_-^{-1}(v)} \phi(e) = \sum_{e\in\partial_+^{-1}(v)} \phi(e)$.

Remark. *ω-flows on weighted digraphs generalize to sheaf-valued flows in the sense that condition (i) generalizes to the structure of a partial S-sheaf and condition (ii) generalizes to an equalizer diagram.*

A $\mathcal{F}$-flow is a generalization of ω-flows, which is represented as an element in an equalizer that generalizes vertex flow conservation.

Definition 3.48. *An $\mathcal{F}$-**flow** is an element in the equalizer of the diagram*

$$\prod_{e\in E_X} H^0_c(\langle e\rangle;\mathcal{F}) \underset{H^0_c(\partial_+ e\subset\langle e\rangle)}{\overset{H^0_c(\partial_- e\subset\langle e\rangle)}{\rightrightarrows}} \prod_{v\in V_X} \mathcal{F}(v)$$

*where $\prod$ is the Cartesian product of underlying set with coordinate-wise operations, for each partial S-sheaf $\mathcal{F}$ on a locally finite digraph X. The **support** $|\phi|$ of an $\mathcal{F}$-flow is the union of $\langle e\rangle$ for all $e \in E_X$ with the e-indexed of ϕ in $H^0_c(\langle e\rangle;\mathcal{F})$ nonzero. An $\mathcal{F}$-flow ϕ is **finite** if $|\phi|$ is finite. An $\mathcal{F}$-flow ϕ is e-**acyclic** if $|\phi| \setminus e$ is acyclic. An $\mathcal{F}$-flow ϕ is **locally S-decomposable** if it lifts to an $\mathcal{F}_0$-flow for a natural transformation $\mathcal{F}_0 \to \mathcal{F}$ from a flat partial S-sheaf $\mathcal{F}_0$.*

The following proposition says that $\mathcal{F}$-flows are classified by first homology.

Proposition 3.2. *For each partial S-sheaf $\mathcal{F}$ on a digraph X,*

$$H^c_1(X;\mathcal{F}) = \textit{finite locally decomposable } \mathcal{F}\textit{-flows}$$

naturally, where the partial S-semimodule contains all finite $\mathcal{F}$-flows if, for each $v \in V_X$, $\mathcal{F}(v)$ is flat.

Proof. Assume that, for each $v \in V_X$, $\mathcal{F}(v)$ is flat. Then the partial S semimodule contains all finite $\mathcal{F}$-flows, by Lemma 3.5, in which the finite direct sums are the Cartesian products of underlying sets equipped with coordinate-wise operations in the definition of $\mathcal{F}$-flows. Thus the partial S semimodule is naturally the image of $\mathcal{F}_0$-flows under a natural partial S-homomorphism from $\mathcal{F}_0$-flows to $\mathcal{F}$-flows induced by a c-sectionwise surjection $\mathcal{F}_0 \to \mathcal{F}$, where $\mathcal{F}_0$ is objectwise projective, and thus flat, and so the $\mathcal{F}$-flows are locally S-decomposable. $\square$

3.3.3 A generalized max-flow min-cut theorem from directed sheaf (co)homology duality

We need to formulate a decomposition of the values of $\mathcal{F}$-flows in terms of $\mathcal{F}$-values of cuts and a generalized sheaf-theoretical MFMC theorem before we can formulate the generalized MFMC theorem.

The decomposition comes from the homotopy limit in the following proposition, where we refer to [8] for a proof, since it requires results about semimodules, which are not the focus of this thesis.

Proposition 3.3. *There exists a terminal natural transformation from a functor*

$$\mathrm{holim}_C[C]_{e,\check{}} : \mathrm{Sh}_{X;\hat{\mathscr{M}}_S} \to \hat{\mathscr{M}}_S$$

to $\bigcap_C[C]_{e,\check{}}$*, where* C *is an element in a collection of subsets of* x*, terminal among all such natural transformations from functors sending c-sectionwise surjections to surjections. For* $\mathcal{F}$ *a direct sum in* $Sh_{X;\hat{\mathscr{M}}_S}$ *of pushforward functors of constant sheaves,* $\mathrm{holim}_C[C]_{e,\mathcal{F}} = \bigcap_C[C]_{e,\mathcal{F}}$.

The generalized sheaf-theoretical MFMC theorem expresses a decomposition of the values of $\mathcal{F}$-flows in terms of $\mathcal{F}$-values of cuts.

Theorem 3.2 (The generalized sheaf-theoretical MFMC theorem)**.** *The equality*

$$\begin{array}{c}\textit{e-values of finite locally}\\ \textit{S-decomposable}\\ \mathcal{F}\textit{-flows}\end{array} = \mathrm{holim}_C[C]_{e,\mathcal{O}_S\otimes_S\mathcal{F}} \subset \bigcap_C[C]_{e,\mathcal{O}_S\otimes_S\mathcal{F}}$$

where the homotopy limit is taken over all minimal e-cuts C *and the lnclusion is an equality if* $H_1^c(-;\mathcal{F})$ *is exact at e, holds for*

i) a cellular sheaf $\mathcal{F}$ *of* S*-semimodules on* X

ii) an edge e *in* X *such that* $X \setminus e$ *is acyclic*

Proof. For each e-cut C, let $[C]$ denote $[C]_{e,\mathcal{O}_S\otimes_S\mathcal{F}}$. Then there exists a natural transformation $\mathcal{F}_0 \to \mathcal{F}$ from a direct sum $\mathcal{F}_0 \in \mathrm{Sh}_{X;\hat{\mathscr{M}}_S}$ of pushforwards of constant sheaves and dotted isomorphisms making the diagram

$$\begin{tikzcd} \bigcap_C[C]_{e,\mathcal{F}_0} \arrow[rr, dashed, "\cong"] \arrow[d] & & H_1^c(X;\mathcal{F}) \arrow[d, "H_0^c(X\setminus e\subset X)\circ\partial_-\circ H_c^0(e\subset X)"] \\ {[e]_{e,\mathcal{O}_S\otimes_S\mathcal{F}}} \arrow[r] & H_c^1(X;\mathcal{O}_S\otimes_S\mathcal{F}) = H_0^c(X;\mathcal{F}) & \end{tikzcd}$$

where the left vertical arrow is the composite of inclusions into $[e]_{e,\mathcal{F}_0}$ with $[e]_{e,e}$ and the bottom horizontal arrow is inclusion, commute, by Lemma 3.1. Furthermore, the left arrows have images the e-values of finite locally S-decomosable $\mathcal{F}$-flows and the right arrows have images $\mathrm{holim}_C[C]_{e,\mathcal{O}_S\otimes_S\mathcal{F}}$, by Proposition 3.2. Then the inclusion follows, since the natural transformation $\mathrm{holim}_C[C]_{e,-} \to \bigcap_C[C]_{e,-}$ is terminal, by Proposition 3.3.

Assume that $H_1^c(-;\mathcal{F})$ is exact at e. If $\lambda \in [e]\setminus\mathrm{Im}\, H_c^0(e \subset X;\mathcal{O}_S\otimes_S\mathcal{F})$, then $\delta_-\lambda \neq \lambda_+\lambda$, by exactness, and so there exists an e-cut C such that $\partial_-\lambda \notin [C]$ and $\partial_+\lambda \in [C]$, by Lemma 3.6; otherwise, $\partial_-\lambda = \partial_+\lambda$, by naturality. Thus the inclusion is an equality. ∎

We can now formulate the generalized MFMC theorem, which says that the generalized union over feasible flows of values of flows equals the generalized intersection over cuts of values of cuts.

Corollary 3.1 (The generalized MFMC theorem)**.** *The equality*

$$\bigvee_{\substack{\textit{finite}\\ S\textit{-decomposable}\\ \textit{flow } \phi}} \phi(e) = \bigwedge_{e\textit{-cut } C} \sum_{c\in C} \omega_c$$

holds for

i) a naturally complete inf-semilattice ordered S-semimodule M

ii) a digraph $(X;\omega)$ with edge weights in M

iii) an edge e in X such that $X \setminus e$ is acyclic

Remark. *There need not be a well-defined max-value among the flow values, since the semimodule of feasible flow values is not required to be totally ordered.*

Proof. For each e-cut C and flow ϕ on $(X;\omega)$,

$$\left(\bigwedge_{e\text{-cut } C} \sum_{c\in C} \omega_c\right) = \bigcap_C \left(\sum_{c\in C} \omega_e\right) = \bigcap_C [C]_{e,\mathcal{O}_S\otimes_S\omega} = \begin{array}{c}e\text{-values of}\\ \text{finite locally } S\text{-}\\ \text{decomposable}\\ \omega\text{-flows}\end{array} = \left(\bigvee_\phi \phi(e)\right)$$

where a term b enclosed in parenthesis (b) denotes the partial S-semimodule $\{a : a \leq_M b\}$ with respect to the natural preorder of M, and where the first equality follows from M being naturally inf-semilattice ordered, the second equality follows from Proposition 3.1, the third equality follows from Theorem 3.2 and Lemma 3.2, and the last equality follows from the naturality of the isomorphism in Proposition 3.2 and M being naturally complete. Then the result follows from the natural preorder on M being antisymmetric, since it is the natural preorder of a semilattice. $\square$

Example 3.21. *The equality in the generalized MFMC theorem holds for the weighted digraph $(X;\omega_{k_{\overline{\mathbb{N}_+}}})$ from Example 3.20 with an additional edge e_8 from the target to the source in Figure 3.6, since (i) $\overline{\mathbb{N}_+}$ is a naturally complete inf-semilattice ordered $\overline{\mathbb{N}_+}$-semimodule, (ii) the digraph has edge weights in $\overline{\mathbb{N}_+}$, and (iii) the edge e_8 is such that $X \setminus e_8$ is acylic. For each of the three flow paths from source to target, the feasible flow value is the intersection of the edge capacities, while, for each cut, the cut value is the union of the edge capacities. The generalized MFMC theorem says that the set of feasible flows equals the intersection over all cuts of the union of edge capacities over the cut, where the value 4 of the union over feasible flows of flow values is equal to the value 4 of the intersection over all cuts of the union of edge capacities over the cut. As confirmation, we see that the weighted digraph has edge weights that correspond to the capacity constraints in the max-flow problem in Example 2.1, which has the solution flow $f(E_X) = (3,1,1,2,1,1,3,4)$ with value 4, and the min-cut problem in Example 2.3, which has the min-cut $K = \delta^+(S) = \{e_6, e_7\}$ with capacity 4.*

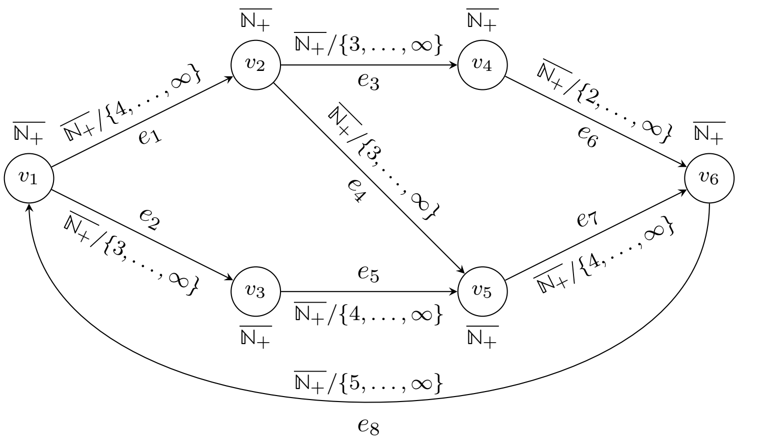

Figure 3.6: A weighted digraph $(X;\omega_{k_{\overline{\mathbb{N}_+}}})$.

3.4 Applications of the generalized max-flow min-cut theorem

3.4.1 Logical flows

We consider an application of the generalized MFMC theorem to logical statements from [9], which utilizes many of the concepts from the quite theoretical two previous sections

Consider the digraph X in Figure 3.7 with edge weights that are subsets of the alphabet $\mathcal{A} = \{A, B, C, D\}$, where we have a Boolean algebra generated by $\mathcal{A}$ under the operations $\{\cup, \cap\}$. The capacity sheaf over X is the semimodule of the powerset $\mathcal{P}$ of edge weights under union, where the semimodule defines which elements are allowed to flow through each edge. For each of the three flow paths, the feasible flow value is the intersection of the edge capacities, while, for each cut, the cut value is the union of the edge capacities. The generalized MFMC theorem says that the set of feasible flows equals the intersection over all cuts of the union of edge capacities over the cut. Thus the only feasible flow consists of A and $0 = \emptyset$, which is equal to the intersection of the smallest cut (dashed edges).

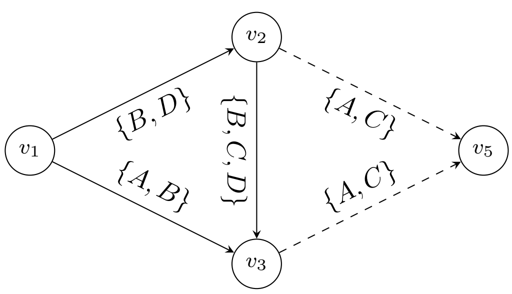

Figure 3.7: A weighted digraph X.

Chapter 4

A generalized linear programming duality theorem

In this chapter, we informally consider a generalization of Chapter 1's linear programming (LP) duality theorem from numerical variables to semimodule-valued variables for linear programs that correspond to Chapter 2 and 3's max-flow and min-cut problems. The generalized LP theorem can be used to tabulate and solve graph-related optimization problems that are expressed as max-flow problems with semimodule-valued variables, like vectors, probability distributions and logical statements, or used to tabulate and solve similar optimization problems that are not really graph-related, in which case using the generalized MFMC theorem makes less sense, though I have yet to actually find any such problem. The idea of a "topological approach to LP duality" was suggested in Robert Ghrist and Sarnjeevi Krishnan's *A Topological Max-Flow-Min-Cut Theorem* [9] as a possible application of Krishnan's generalized MFMC theorem, which we review in Chapter 3, and which I have used to develop and prove a generalized LP duality theorem for a special case.

In the first section, we define generalized max-flow and dual min-cut linear programming problems. In the second and last section, we formulate the generalized linear programming duality theorem for such linear programs.

4.1 The generalized max-flow and min-cut linear programming problems

We need to define the generalized max-flow and dual min-cut linear programming problems before we can formulate a generalized linear programming duality theorem for such linear programs. In the following, suppose that M is a naturally complete inf-semilattice ordered S-semimodule (see Definition 3.6).

The generalized max-flow linear programming problem is a generalization of the max-flow linear programming problem (see Definition 2.9) from numerical variables to semimodule-valued variables.

Definition 4.1. *Suppose* $b \in M^{m+n} = (0,\dots,0,b_{m+1},\dots,b_{m+n})$ *defines a constraint* $x_i \leq_M b_{m+i}$ *on each* $x_i \in x \in M^n$ *with respect to the natural preorder of* M. *The* ***generalized max-flow linear programming problem*** *is to*

maximize $c^T x = x_n$

subject to $Ax \leq_M b$ *and* $x \in M^n$

where $c \in \{0,1\}^n = (0,\dots,0,1)$ *and* $A \in \{-1,0,1\}^{(m+n)\times n} = \begin{bmatrix} G \\ I \end{bmatrix}$ *is such that*

i) G *is an* $m \times n$ *matrix*

ii) each column in G *either contains only zeros or exactly one* 1 *and one* -1.

iii) the first row in G *has* -1 *as its last compoent and has no other* -1*s, while the last row in* G *has* 1 *as its last component and has no other* 1*s.*

iv) I *is the* $n \times n$ *identity matrix*

Remark. *The matrix corresponds to a digraph with distinct source and target vertices and an edge between them whose removal makes the digraph acylic (see Definition 2.9), where multiplying the matrix with the variable vector makes the corresponding digraph a weighted digraph. However, multiplication of a negative entry with a semimodule-valued variable does not necessarily represent an additive inverse element, since semimodules are not required to contain additive inverse elements.*

Example 4.1. *The weighted digraph* $(X;\omega_{k_{\overline{\mathbb{N}_+}}})$ *from Example 3.21 in Figure 4.1 has the following generalized max-flow linear programming problem*

maximize x_8

subject to $Ax \leq b$ *and* $x \in \mathbb{R}^8_+$

where

$$Ax = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 3 \\ 2 \\ 2 \\ 2 \\ 3 \\ 1 \\ 3 \\ 4 \end{bmatrix} = b$$

which is actually the ordinary max-flow linear programming problem from Example 2.2, which has the solution $x = (3,1,1,2,1,1,3,4)$ *with value* $x_8 = 4$, *which in turn satisfies the single-variable constraints* $x_1 = 3 \leq 3 = b_7$, $x_2 = 1 \leq$

$2 = b_8$, $x_3 = 1 \leq 2 = b_9$, $x_4 = 2 \leq 2 = b_{10}$, $x_5 = 1 \leq 3 = b_{11}$, $x_6 = 1 \leq 1 = b_{12}$, $x_7 = 3 \leq 3 = b_{13}$, $x_8 = 4 \leq 4 = b_{14}$ *and the several-variable constraints* $x_1 + x_2 - x_8 = 3 + 1 - 4 = 0 = b_1$, $-x_1 + x_3 + x_4 = -3 + 1 + 2 = 0 = b_2$, $-x_3 + x_6 = -1 + 1 = 0 = b_3$, $-x_2 + x_5 = -1 + 1 = 0 = b_4$, $-x_4 - x_5 + x_7 = -2 - 1 + 3 = 0 = b_5$, $-x_6 - x_7 + x_8 = -1 - 3 + 4 = 0 = b_6$.

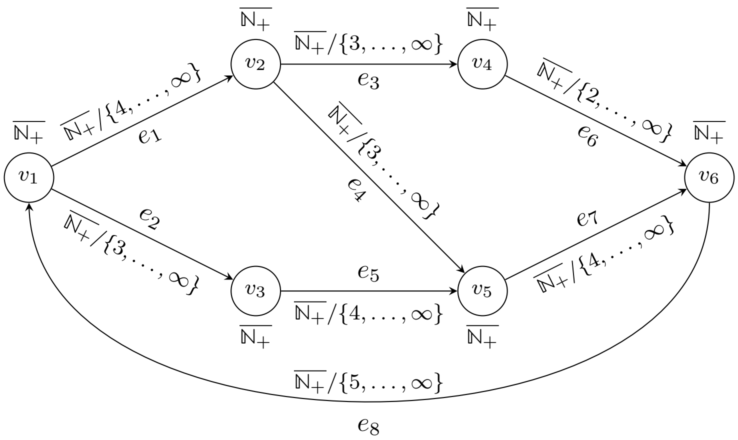

Figure 4.1: A weighted digraph $(X; \omega_{k_{\overline{\mathbb{N}_+}}})$.

The generalized dual min-flow linear programming problem is a generalization of the min-flow linear programming problem (see Definition 2.13) from numerical variables to semimodule-valued variables.

Definition 4.2. *The* ***generalized dual min-cut linear programming problem*** *of a generalized max-flow linear programming problem is to*

minimize $b^T y$

subject to $A^T y \geq c$ *and* $y \in \{0, 1\}^{m+n}$

Example 4.2. *The generalized max-flow linear programming problem from the previous example has the following generalized dual min-cut linear programming problem*

minimize $b^T y$

subject to $A^T y \geq c$ *and* $y \in \{0, 1\}^{14}$

where

$$b = \begin{bmatrix} 0\\0\\0\\0\\0\\0\\3\\2\\2\\2\\3\\1\\3\\4 \end{bmatrix}, A^T y = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_1\\y_2\\y_3\\y_4\\y_5\\y_6\\y_7\\y_8\\y_9\\y_{10}\\y_{11}\\y_{12}\\y_{13}\\y_{14} \end{bmatrix} \geq \begin{bmatrix} 0\\0\\0\\0\\0\\0\\0\\1 \end{bmatrix} = c$$

which is actually the ordinary min-cut linear programming problem from Example 2.4, which has the solution $y = (0,0,0,0,0,0,0,0,0,0,0,1,1,0)$ *with value* $b^T y = 4$.

4.2 A generalized linear programming duality theorem

We can now formulate the generalized linear programming duality theorem for the generalized max-flow and dual min-cut linear programming problems.

Theorem 4.1. *For each generalized max-flow linear programming problem and its generalized dual min-cut linear programming problem,*

$$\bigvee_{x \in X} x_n = \bigwedge_{y \in Y} b^T y$$

where $X \subseteq M^n$ *and* $Y \subseteq \{0,1\}^{m+n}$.

Proof. A generalized max-flow linear programming problem corresponds to a weighted digraph (X, ω_M), which, by construction, satisfies the three conditions of the generalized MFMC theorem (Corollary 3.1). Thus

$$\bigvee_{x \in X} x_n = \bigvee_{\substack{\text{finite } S\text{-}\\ \text{decomposable}\\ \text{flow } \phi}} \phi(e) = \bigwedge_{e_n\text{-cut } C} \sum_{c \in C} \omega_c = \bigwedge_{y \in Y} b^T y$$

where $X \subseteq M^n$ is defined to consist of the elements $x \in M^n$ that correspond to finite S-decomposable flows ϕ and $Y \subseteq \{0,1\}^{m+n}$ is defined to consist of the elements $y \in \{0,1\}^{m+n}$ that correspond to e_n-cuts, which makes the first and last equality hold, while the middle equality is Corollary 3.1. □

Example 4.3. *The value of our generalized max-flow linear programming problem is equal to the value of its generalized dual min-cut linear programming problem, which exemplifies the ordinary case of the generalized LP duality theorem.*

Remark. *While this generalized linear programming duality theorem only holds for special linear programs with variables in a semimodule, there exists a generalized linear programming duality theorem for general linear programs with variables in a semimodule over the real numbers [10], which might indicate that our generalization can be extended from special linear progams to general linear programs, but I have not found any obvious way to do so and any further investigation is unfortunately beyond the timeline of this thesis.*

Bibliography

[1] A. Schrijver, A course in combinatorial optimization. Amsterdam: CWI, 2008.

[2] S. Awodey, Category Theory. Oxford University Press, 2006.

[3] B.R. Tennison, Sheaf Theory. Cambridge University Press, 2007.

[4] G. Dahl, Network flows and combinatorial matrix theory. Oslo: University of Oslo, 2013.

[5] T.E Harris and F.S Ross, *Fundamentals of a method for evaluating rail net capacities*, RAND Corporation, 1955.

[6] L.R. Ford and D.R. Fulkerson, *Maximal Flow through a Network*, Santa Monica, California: RAND Corporation, 1954.

[7] G.B Dantzig and D.R. Fulkerson, *On the Max Flow Min Cut Theorem of Networks*, Santa Monica, California: RAND Corporation, 1955.

[8] S. Krishnan, *Flow-cut dualities for sheaves on graphs*, arXiv: 1409.6712, 2014.

[9] R. Ghrist and S. Krishnan, *A Topological Max-Flow-Min-Cut theorem*, Global Conference on Signal and Information Processing, 2013.

[10] U. Zimmermann, *Duality for Algebraic Linear Programming*, Köln: University of Köln, 1980.