Diplomarbeit

Topological and Generalized Metric Methods for Logic Programming Semantics

angefertigt am Institut für Informatik III

Dezember/2001

erstellt von
Roland Heinze

betreut von
Prof. Dr. Rainer Manthey
und
Dr. Anthony Karel Seda
Abstract

This diploma thesis essentially consists of three parts. These are: convergence structures for computation, generalized metric methods to obtain a unique declarative semantics of some classes of logic programs, and the investigation of the topological and metrical continuity of various operators used in logic programming semantics.

We define convergence spaces and convergence classes in net and in filter notation. The net notation of convergence spaces and the filter notation of convergence classes are new. We show that the class of topological convergence spaces in a particular notation is exactly the class of convergence classes in that notation. We will see how one can constructively change the representation of a convergence class into one either in net or in filter notation.

We extend the classes of $\Phi^+$- and $\Phi$-accessible programs. Instead of classes of programs with a unique supported model, that is, with a unique fixpoint of the (2-valued) immediate consequence operator $T_P : I_P \rightarrow I_P$, we are interested in classes of programs with a unique fixpoint of Fitting’s operator $\Phi_P$ which acts on the space of (3-valued) valuations. Extended $\Phi^+$- and extended $\Phi$-accessible programs allow fixpoints which are 3-valued but we will see that both classes coincide with the original classes of programs, that is, the $\Phi^+$- and $\Phi$-accessible programs, respectively. The classes of 3-valued $\Phi^+$- and 3-valued $\Phi$-accessible programs allow facts of the form $A \leftarrow u$ as program clauses, that is, the corresponding programs are no longer normal logic programs (NLPs) but so called 3-valued NLPs. We get proper supersets of the original program classes and there exist programs $P$ in these classes such that $\Phi_P$ has a unique fixpoint which is not 2-valued.

Finally we investigate the topological and metrical continuity of the operator $W_P$ of the well-founded model and the operators $S_P$ and $A_P$ of the alternating fixpoint model. We prove that the operator $W_P$ is not topologically continuous with respect to the topology $Q_2$ (see Definition 2.4.2) and $S_P$ and $A_P$ are not topologically continuous with respect to the atomic topology $Q$ which was defined in [38]. The only thing we can show is a kind of 'semi'-continuity (see Lemma 4.3.2, Lemma 4.4.1 and Corollary 4.3.5). We give counterexamples for the lack of continuity in each case.

The discrete quasi ultrametrics $d_k$ on $I_P \times I_P$ and $d_+$ on $I_P$ of Definition 4.5.1 are applied to the operator $W_P$ and the operators $S_P$ and $A_P$, respectively. The operator $W_P$ is not metrically continuous with respect to $d_k$ but $S_P$ and $A_P$ are metrically continuous with respect to $d_+$. 
Introduction

The definition of the declarative semantics of normal logic programs (NLPs) is an open goal of research in logic programming semantics (see e.g. [37]). The problem of defining a 'meaningful' semantics of a NLP lies already in the different possibilities to interpret the word 'meaningful'. Different researchers may define different semantics for the same NLP and there exist reasons why each of these semantics is meaningful from that viewpoint. To overcome these problems we are particularly interested in NLPs which possess a unique semantics in some way.

The semantics of a NLP $P$ is usually defined by means of a fixpoint of an operator acting on sets of possible interpretations of the program $P$. We will only consider Herbrand interpretations $I \in I_P$, partial (or 3-valued) interpretations $I \in I_P \times I_P$ (based on Herbrand interpretations) and (3-valued) valuations as a possible semantics of $P$. Particularly ground atoms of the language underlying $P$ can only have the truth values true or false in the 2-valued case and additionally the truth value undefined in the 3-valued case. We consider the immediate consequence operator $T_P : I_P \rightarrow I_P$, Fitting’s operator $\Phi_P : 2^B^P \rightarrow 2^B^P$, the well-founded operator $W_P : I_P \times I_P \rightarrow I_P \times I_P$ and the operators of the alternating fixpoint model $S_P, A_P : I_P \rightarrow I_P$.

Many different kinds of semantics were defined in the past. The supported model semantics, that is, fixpoints of $T_P$ are frequently used. They coincide with Clark’s completed program model semantics (see [29]). Other semantics such as the 2-valued and 3-valued stable model semantics (see [16,35]), the well-founded semantics (see [15]), the alternating fixpoint semantics (see [14]) and Fitting’s Kripke-Kleene semantics (see [9]) exist. The perfect model semantics (see [32,34]) is in fact a supported model semantics. The class of NLPs with a unique supported model can be further subdivided into locally hierarchical NLPs (see Definition 3.1.2), acceptable NLPs (see [18,19]), $\Phi^+$- and $\Phi$-accessible NLPs (see Definition 3.1.4).

Consider the NLP $P$ which consists only of the program clause $A \leftarrow A$. It is an example of the case where the well-founded semantics is different from Fitting’s Kripke-Kleene semantics. The well-founded semantics assigns the truth value false to the atom $A$ and Fitting’s Kripke-Kleene semantics assigns the truth value undefined to $A$. Particularly $P$ does not have a unique fixpoint with respect to the operators $W_P$ and $\Phi_P$.

The above classes of NLPs with unique supported models (locally hierarchical etc.) have one thing in common: One can use level mappings $l : B_P \rightarrow \gamma$ to describe the classes and one can use generalized metric methods to prove the existence of a unique fixpoint as well as to construct a transfinite sequence which converges with respect to the chosen generalized metric space to that fixpoint.

Metric methods were introduced to logic programming semantics by M. Fitting in [12]. Other authors developed and applied more general metric methods to logic programming semantics.
Applying further mathematical methods to logic programming semantics can never be wrong and has been done. H. Blair et al. showed in [3] how continuous semanticians can be used for logic programming semantics. Topological methods were e.g. used by A. K. Seda and P. Hitzler. A. K. Seda defined some helpful topologies as the positive atomic topology $Q^+$ (Scott topology) and the atomic topology $Q$ (Cantor topology) to characterize the continuity of the operator $T_P$, that is, to give necessary and sufficient conditions for the continuity of $T_P$ with respect to $Q$ and $Q^+$ (see [38]).

Mathematical methods often involve a kind of convergence structure such as topologies or generalized metric structures. Convergence spaces and convergence classes build an abstract mathematical framework to describe convergence and particularly convergence in computing. They can be used to combine discrete and continuous models of computation (see [4]).

In Chapter 0 we recall the basic terms and notations of first order logic, set theory, logic programming and topology used throughout the whole diploma thesis.

Chapter 1 treats convergence spaces and convergence classes in net and filter notation. The net notation of convergence spaces and the filter notation of convergence classes are new. We show that the class of topological convergence spaces in a particular notation is exactly the class of convergence classes in that notation. We will see how one can constructively change the representation of a convergence class into one either in net or in filter notation.

Some different mathematical methods used to determine the semantics of logic programs are described in Chapter 2. We subsume some important results from lattice theory e.g. Tarski’s theorem (Theorem 2.1.1) and give an overview of the different generalizations of metric spaces. Dislocated generalized ultrametrics (d-gums) and quasiultrametrics (particularly Rutten’s theorem, see Theorem 2.2.6) will play an important role in subsequent chapters. We describe the necessary theory. Domain theory is often used for programming language semantics. We show how a domain can be recast into a gum. The topologies $Q, Q^+, Q^-, Q_2$ and $Q_2^-$ are defined and characterized in the last section of Chapter 2.

In Chapter 3 the unique supported model semantics and 3-valued semantics using Fitting’s operator $\Phi_P$ are described in more detail. We will extend the classes of $\Phi^+$- and $\Phi$-accessible programs. Instead of classes of programs with a unique supported model, that is, with a unique fixpoint of the (2-valued) immediate consequence operator $T_P : \Phi_P \rightarrow \Phi_P$, we are interested in classes of programs with a unique fixpoint of Fitting’s operator $\Phi_P$ which acts on the space of (3-valued) valuations. Extended $\Phi^+$- and extended $\Phi$-accessible programs allow fixpoints which are 3-valued but we will see that both classes coincide with the original classes of programs, that is, the $\Phi^+$- and $\Phi$-accessible programs, respectively. The classes of 3-valued $\Phi^+$- and 3-valued $\Phi$-accessible programs allow facts of the form $A \leftarrow u$ as program clauses, that is, the corresponding programs are no longer normal logic programs (NLPs) but so called 3-valued NLPs. We get proper superset of the original program classes and there exist programs $P$ in these classes such that $\Phi_P$ has a unique fixpoint which is not 2-valued.

In Chapter 4 we investigate the topological and metrical continuity of the operator $W_P$ of the well-founded model (see Definition 4.1.4) and the operators $S_P$ and $A_P$ of the alternating fixpoint model (see Definition 4.2.1 and Definition 4.2.2). We prove that the operator $W_P$ is not topologically continuous with respect to the topology $Q_2$ (see Definition 2.4.2) and $S_P$ and $A_P$ are not topologically continuous with respect to the atomic topology $Q$ which was defined in [38]. The only thing we can show is a kind of ‘semi’-continuity (see Lemma 4.3.2, Lemma 4.4.1 and Corollary 4.3.5). We give counterexamples for the lack of continuity in each case. The discrete quasi ultrametrics $d_q$ on $I_P \times I_P$ and $d_+$ on $I_P$ of Definition 4.5.1 are applied to the operator $W_P$ and the operators $S_P$ and $A_P$, respectively. The operator $W_P$ is
not metrically continuous with respect to $d_k$ but $S\bar{p}$ and $A\bar{p}$ are metrically continuous with respect to $d_+$.

Acknowledgements

I want to thank Dr. Anthony Karel Seda for his support which led me to some different aspects of the recent research in logic programming semantics. Particularly my work on convergence spaces and convergence classes is inspired by him. Furthermore I want to thank Dr. Pascal Hitzler for reading, commenting and correcting some of my early postscript files. He gave me some hints and asked me some interesting questions which pushed my work. Last but not least I want to thank Prof. Dr. Manthey who made this diploma thesis possible and who supervised it in the whole.
INTRODUCTION
Contents

0 Preliminary Definitions and Notations ........................................... 1

1 Convergence in Computing .................................................. 5
   1.1 Models of Computation .............................................. 5
   1.2 Introduction to Convergence Structures .......................... 5
   1.3 Nets, Filters and Closure Operators .............................. 6
       1.3.1 Nets ..................................................... 6
       1.3.2 Filters .................................................. 7
       1.3.3 Relationship between Nets and Filters .................. 8
       1.3.4 Closure Operators ...................................... 9
   1.4 Convergence Spaces .................................................. 9
       1.4.1 Convergence Spaces in Net Notation ...................... 10
       1.4.2 Convergence Spaces in Filter Notation ................... 11
   1.5 Convergence Classes .............................................. 12
       1.5.1 Convergence Classes in Net Notation ..................... 12
       1.5.2 Convergence Classes in Filter Notation ................. 14
       1.5.3 Properties of Convergence Classes in Net Notation ... 17
       1.5.4 Properties of Convergence Classes in Filter Notation .. 18
       1.5.5 Constructively Changing the Notation of a Convergence Class .... 19

2 Mathematical Methods for Logic Programming Semantics ............. 21
   2.1 Lattice Theory ................................................... 21
   2.2 Generalized Metric Methods ...................................... 22
   2.3 Domain Theory ................................................... 26
   2.4 Topologies for Logic Programming Semantics .................... 27

3 Extensions to the Classes of $\Phi^*- \text{ and } \Phi^*$-accessible Programs ............. 31
   3.1 Some Classes of Logic Programs and their Semantics .......... 31
       3.1.1 Supported Model Semantics ................................ 31
       3.1.2 Three-Valued Semantics of NLPs ......................... 33
   3.2 The Extended Program Classes ................................... 34
       3.2.1 Preliminary Definitions .................................... 34
       3.2.2 Extensions of the Class of $\Phi^*- \text{ accessible Programs} ...... 35
       3.2.3 Extensions of the Class of $\Phi^*$-accessible Programs ....... 40
       3.2.4 Summary .................................................. 44
4 Topological Investigations of the Operators of the Well-Founded and AFP Model
4.1 The Well-Founded Model .................................................. 47
4.2 The Alternating Fixpoint Model ........................................ 48
4.3 Topological Investigations of the Well-Founded Operator $W_p$ ... 49
  4.3.1 A New Representation of the Operator $U_p$ ...................... 49
  4.3.2 Topological Investigations of the Operator $U_p$ ................. 51
  4.3.3 Topological Investigations of the Well-Founded Operator $W_p$ . . 52
4.4 Topological Investigations of the Operators of the Alternating Fixpoint Model 53
  4.4.1 Topological Investigations of the Operator $S_p$ .................. 53
  4.4.2 Topological Investigations of the AFP Operator $A_p$ ............ 54
4.5 Quasi-Metrics and the Well-Founded and AFP Model ................ 54

References ................................................................. 57
Chapter 0

Preliminary Definitions and Notations

We assume that the reader is familiar with first order logic, set theory and with the basic concepts of logic programming and topology. For basic information on logic programming see [29]. Basic information on general topology can be found in [26, 45].

We define \( \mathbb{N} = \{1, 2, 3, \ldots \} \) to be the natural numbers, \( \mathbb{N}_0 = \mathbb{N} \cup \{0\} \), \( \mathbb{R} \) to be the real numbers, \( \mathbb{R}_0^+ \) to be the non-negative real numbers and \( (a, b) = \{c \in \mathbb{R} \mid a < c < b\} \) to be an interval of \( \mathbb{R} \) for all \( a, b \in \mathbb{R} \cup \{-\infty, \infty\} \) with \( a < b \). Let \( \mathcal{O} \) be the class of all ordinal numbers and \( \omega \in \mathcal{O} \) be the first limit ordinal. Let \( \aleph_0 \) be the first infinite cardinal number.

We use \( \alpha, \beta \) and \( \gamma \) (i.e. lower-case Greek letters) to denote arbitrary ordinal numbers and write \( \text{Lim}(\alpha) \) iff \( \alpha \) is a limit ordinal number. The last upper-case Arabic letters \( (X, Y, Z) \) will often denote arbitrary sets and the last lower-case Arabic letters \( (x, y, z) \) will denote elements (or points) of sets. The lower-case Arabic letters \( f, g, h \) will be used to denote functions. If \( f : X \to Y \) is a function and \( A \subseteq X \), we define \( f(A) = \{f(x) \mid x \in A\} \) and \( \text{Im} f = f(X) \) to be the image of \( f \). In the case \( X = Y \) an element \( x \in X \) is a fixpoint of \( f \) if \( f(x) = x \). We denote the power set of a set \( X \) by \( \mathcal{P}(X) \). If \( A \subseteq X \) for a set \( X \), we mostly use \( A^c \) to denote the complement \( X \setminus A \) of \( A \) in \( X \). A transfinite sequence \( (x_\alpha)_{\alpha < \gamma} \) in a set \( X \) is a mapping \( f : \gamma \to X \) with \( f(\alpha) = x_\alpha \in X \) for all \( \alpha < \gamma \) and an ordinal number \( \gamma \). If \( \gamma = \omega \), we speak of the sequence \( (x_\alpha)_{\alpha < \gamma} \) and also write \( (x_n)_{n \in \mathbb{N}} \), instead.

Mostly \( \mathcal{T} \) will denote a topology on a set \( X \) and by \( (X, \mathcal{T}) \) we will denote a topological space. We use \( \overline{A} \) or \( \overline{A}^\mathcal{T} \) to denote the (topological) closure of a subset \( A \subseteq X \) with respect to the topology \( \mathcal{T} \) on the set \( X \). We denote by \( N(x) \) the neighbourhood system of a point \( x \in X \) with respect to the topological space \( (X, \mathcal{T}) \). The calligraphic letters \( \mathcal{A}, \mathcal{T}, \mathcal{G} \) and \( \mathcal{H} \) frequently denote filters on a set \( X \).

We expound the basic definitions and notations of the theory of logic programming used in the diploma thesis.

A term is a variable, constant or of the form \( f(t_1, \ldots, t_n) \) where \( f \) is a function symbol and each \( t_i \) is a term. An atom is of the form \( p(t_1, \ldots, t_n) \) where \( p \) is a predicate symbol and each \( t_i \) is a term. Formulas are built from atoms. If \( F \) and \( G \) are formulas, we get new formulas by negation \( (\neg F) \), conjunction \( (F \land G) \), disjunction \( (F \lor G) \), implication \( (F \rightarrow G) \), equivalence \( (F \leftrightarrow G) \) and quantification \( (\forall x F) \) or \( (\exists x F) \). A ground atom is a variable free atom. The first upper-case Arabic letters \( (A, B, C, \ldots) \) will frequently denote (ground) atoms of a logic program. A literal is an atom \( A \) or the negation \( \neg A \) of an atom \( A \) and will often
be denoted by \( L \). A **positive literal** is an atom and a **negative literal** is the negation of an atom. A **clause** is a formula of the form \( \forall x_1 \ldots \forall x_k (L_1 \lor \ldots \lor L_n) \) where each \( L_i \) is a literal, \( x_1, \ldots, x_k \) are the variables occurring in \( L_1 \lor \ldots \lor L_n \) and \( n \in \mathbb{N} \). A **program clause** \( R \) is a clause of the form

\[
\forall x_1 \ldots \forall x_k (A \lor \neg A_1 \lor \ldots \lor \neg A_n \lor B_1 \lor \ldots \lor B_m)
\]

where \( A, A_i, B_j \) are atoms for all \( 1 \leq i \leq n, 1 \leq j \leq m, m \in \mathbb{N}_0 \). We use the abbreviation

\[
A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m.
\]

In that case \( A \) is called the **head** of the clause and \( A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m \) the **body** of the clause. We define

\[
\begin{align*}
\text{head}(R) &= A, \\
\text{pos}(R) &= \{A_1, \ldots, A_n\}, \\
\text{neg}(R) &= \{B_1, \ldots, B_m\}.
\end{align*}
\]

A **unit clause** or **fact** is a program clause of the form \( A \leftarrow \). We will also write \( A \leftarrow t \) to denote facts. A **normal logic program** or **NLP** is a finite set of program clauses. A **definite logic program** or **positive logic program** is a finite set of program clauses each consisting only of positive literals in the body. We will mostly denote a definite or normal logic program by \( \mathbf{P} \). The constant, function and predicate symbols of the **alphabet** of a NLP \( \mathbf{P} \) are exactly the constant, function and predicate symbols occurring in \( \mathbf{P} \). If no constant symbols occur in the language of \( \mathbf{P} \) then we add a dummy constant symbol to the alphabet of \( \mathbf{P} \). The **underlying language** of \( \mathbf{P} \) consists of all formulas built from the alphabet of \( \mathbf{P} \).

We consider the **semantics** of a normal logic program. When we speak of the semantics of a NLP we always mean the **declarative semantics** in contrast to the **procedural semantics** (see [29]). The semantics is defined by **interpretations** or **valuations**. We only use interpretations based on **Herbrand pre-interpretations** i.e. the domain of each pre-interpretation is the set of all variable free terms and constant and function symbols are interpreted by themselves (see [29]). Therefore we define \( R' \) to be a **ground instance** of a program clause \( R \) iff one gets \( R' \) by replacing each occurrence of a variable in \( R \) by a variable free term. We denote by \( \text{ground}(\mathbf{P}) \) the set of all ground instances of clauses of a NLP \( \mathbf{P} \). A **ground atom** is an atom without variables i.e. each term of that atom contains no variable. We denote by \( Bp\mathbf{P} \) the set of all ground instances of atoms occurring in clauses of \( \mathbf{P} \). \( Bp\mathbf{P} \) is called the **Herbrand base** of \( \mathbf{P} \). We interpret \( I_\mathbf{P} = \mathcal{P}(Bp\mathbf{P}) = 2^{Bp\mathbf{P}} \) to be the set of all **Herbrand interpretations** of a NLP \( \mathbf{P} \). A **(2-valued) interpretation** of \( \mathbf{P} \) is an element of \( I_\mathbf{P} \). A **(3-valued or partial) interpretation** \( I \) of \( \mathbf{P} \) is given by \( I = (I^+, I^-) \in I_\mathbf{P} \times I_\mathbf{P} \). \( I \) is **consistent** iff \( I^+ \cap I^- = \emptyset \) and \( I \) is **total** iff \( I \) is consistent and \( I^+ \cup I^- = Bp\mathbf{P} \). For each \( I \in I_\mathbf{P} \) and each ground atom \( A \) of \( \mathbf{P} \) we write \( I \models A \) iff \( A \in I \) and write \( I \not\models \neg A \) iff \( A \notin I \). We continue that by \( I \models L_1 \land \ldots \land L_n \) iff \( I \models L_i \) for all \( 1 \leq i \leq n \) and ground literals \( L_1, \ldots, L_n \). We use the **truth values** \( t, f, u \) which are abbreviations for **true**, **false** and **undefined**, respectively. We use the following truth table for negation, conjunction and disjunction of our truth values:
A 3-valued logic based on the above table is called Kleene’s strong 3-valued logic (see [9]). We define an ordering $\leq_t$ on the truth values by $f \leq_t u \leq_t t$, see [10]. Let $\mathcal{B}$ be a set called set of truth values. Then a valuation is a mapping $v: B_\mathcal{P} \rightarrow \mathcal{B}$. We will only use the truth value set $\mathcal{B} = \{t, f, u\}$. If $\text{Im} \mathcal{v} \subseteq \{t, f\}$, we say that the valuation is 2-valued or $v$ is a (2-valued) valuation. We extend the definition of valuations to conjunctions and disjunctions of literals. For each ground atom $A$ and literals $L_1, \ldots, L_n$ of the language underlying a NLP $\mathcal{P}$ we define

$$v(\neg A) = \neg v(A),$$
$$v(L_1 \land \ldots \land L_n) = \min\{v(L_i) \mid 1 \leq i \leq n\},$$
$$v(L_1 \lor \ldots \lor L_n) = \max\{v(L_i) \mid 1 \leq i \leq n\}.$$
CHAPTER 0. PRELIMINARY DEFINITIONS AND NOTATIONS
Chapter 1

Convergence in Computing

1.1 Models of Computation

There exist many models of computation, e.g. Turing machines, recursion equations, register machines, finite state machines, logic programs, ordinary and partial differential equations. Some are discrete models of computation e.g. logic programs or finite state machines. Others are continuous as ordinary and partial differential equations. Sometimes it is possible to incorporate continuous models in computation where formerly only discrete models were used. E.g. in [3] continuous models of computation for logic programs are described. Continuous models sometimes open a door to a new and even better way to solve a problem. That is not surprising. Computation often involves models which are mathematically underdeveloped. As continuous mathematics is an old and well-developed science it frequently offers powerful methods. These methods are the strength of continuous models of computation.

E.g. if one only uses a discrete approximation method to solve a problem without seeing the underlying continuous model one might not be able to develop better approximations with less computational time to the solution of that problem. So sometimes a continuous computational model is more suitable and more natural to the problem under consideration even if the former discrete model seemed to be the natural one. Therefore finding suitable continuous models is a promising step to embed a problem in a more general context and to solve it with perhaps very different and more elegant (that means more exact, faster or less time or space consuming) methods.

Convergence spaces (see Definition 1.4.1 and Definition 1.4.3) are a way to describe hybrid computation models (see [4]), that is, they are able to combine discrete and continuous models of computation. They incorporate continuous mathematics for specifying, analysing and verifying models of computation.

1.2 Introduction to Convergence Structures

Nets and filters are a common way to define convergence in topological spaces. Sequences are a special kind of net which are adequate in first countable topological spaces (see [45, p. 71]) but not for general topological spaces. Filters are inspired by the fact that the neighbourhoods at every point of a topological space already describe the topology and therefore also convergence in the whole. Convergence spaces describe the convergence structure of a space at every point using nets or filters. Every topological space is embeddable into the class of convergence
spaces (see Lemma 1.4.2 and Lemma 1.4.5), but not every convergence space is the image of a topological space under that embedding (see [17]).

In practice one often has a space $X$ with some properties but one does not have a convergence structure or even a topology defined on the space $X$. In such cases it is helpful to know which properties are necessary and sufficient for the existence of a convergence structure on $X$. Convergence classes for $X$ in net notation and in filter notation are collections of pairs each consisting of a net or filter and a point of $X$ such that $X$ gets a topology, and a net or a filter converges to a point $x$ with respect to that topology iff the pair of net or filter together with the point $x \in X$ is an element of the convergence class. Each convergence class also corresponds to a convergence space with additional properties. The induced topologies of the convergence class and the corresponding convergence space are identical.

The notions of net and filter can be interchanged with respect to convergence classes in the sense that for every convergence class in net notation there exists a convergence class in filter notation (and vice versa) such that both have the same induced topology. One can construct the convergence class in one notation from the convergence class in the other notation. Every topological space can also be transformed into a convergence class in net or filter notation such that the induced topology of the convergence class is identical to the original topology.

1.3 Nets, Filters and Closure Operators

We intend to expound the parts of the theory of nets, filters and closure operators which are necessary and important in the following sections. For more information on nets and filters see e.g. [17, 26, 45].

1.3.1 Nets

Definition 1.3.1 (Directed Set) Let $D$ be a nonempty set and $\leq$ be a binary relation on $D$, that is, a subset of $D \times D$. We say $\leq$ directs $D$ and $D$ is a directed set if the following conditions are satisfied

(a) $\leq$ is reflexive, that is, $d \leq d$ for all $d \in D$.

(b) $\leq$ is transitive, that is, if $d_1 \leq d_2$ and $d_2 \leq d_3$ then $d_1 \leq d_3$.

(c) If $d_1, d_2 \in D$ then there exists $d_3 \in D$ with $d_1 \leq d_3$ and $d_2 \leq d_3$.

Definition 1.3.2 (Net and Subnet) Let $D$ be a directed set. A net in a set $X$ is a function $S : D \rightarrow X$. The point $S(n), n \in D$, is often denoted $S_n$ or $x_n$ and we frequently speak of "the net $(S_n)_{n \in D}$" or "the net $(x_n)_{n \in D}$" instead of the net $S$. If no confusion can arise we use $(x_n)$ as an abbreviation for $(x_n)_{n \in D}$.

Let $M$ be another directed set. A subnet $T$ of the net $S$ is given by a cofinal function $\varphi : M \rightarrow D$ such that $T = S \circ \varphi$. Thereby $\varphi$ is cofinal if for all $n \in D$ there exists $m \in M$ such that for all $p \in M$ with $p \geq m$ we have $n \leq \varphi(p)$. The point $S \circ \varphi(m)$ is often denoted $S_{n_m}$ or $x_{n_m}$, and we say "$(x_{n_m})_{m \in M}$ is a subnet of $(x_n)_{n \in D}$". If $(x_n)$ is a net in $X$ a set of the form $\{x_n | n \geq n_0\}, n_0 \in D$, is called tail of $(x_n)$.

A net $(x_n)$ in a set $X$ is defined for an arbitrary set $X$. If we want to define net convergence $X$ has to be a topological space.
1.3. NETS, FILTERS AND CLOSURE OPERATORS

Definition 1.3.3 (Net Convergence) Let $X$ be a topological space and $S : D \to X$ be a net in $X$. $(S_n)$ converges to $x \in X$, written $S_n \to x$ or $\lim S_n = x$, if for each neighbourhood $U$ of $x$ there exists $n_0 \in D$ such that $S_n \in U$ for each $n \geq n_0$. If $(S_n)$ converges to $x$ we say $(S_n)$ is residually (or eventually) in every neighbourhood $U$ of $x$.

So $(S_n)$ converges to $x$ if each neighbourhood $U$ of $x$ contains a tail of $(S_n)$. The following theorem will be used later. A proof can be found in [45, p. 75]

Theorem 1.3.1 Let $X$ be a topological space, $E \subseteq X$. We have $x \in \overline{E}$ iff there exists a net $(x_n)$ in $E$ with $x_n \to x$.

Proof. Let $x \in \overline{E}$. For every $U_1, U_2 \in N(x)$ we define $U_1 \subseteq U_2$ iff $U_2 \subseteq U_1$. Then $N(x)$ becomes a directed set by $\leq$. As $U \cap E \neq \emptyset$ for each $U \in N(x)$ we can choose $x_U \in U \cap E$ for each $U \in N(x)$ and $(x_U)$ is a net in $E$ which converges to $x$.

Now let $(x_n)$ be a net in $E$ with $x_n \to x$. Then for each $U \in N(x)$ there exists $m$ such that $x_n \in U$ for all $n \geq m$. Particularly we have $U \cap E \neq \emptyset$ for each $U \in N(x)$. So we get $x \in \overline{E}$.

1.3.2 Filters

Definition 1.3.4 (Filter, Filter Base and Ultrafilter) A filter $\mathcal{F}$ on a set $X$ is a non-empty collection of non-empty subsets of $X$ such that $\mathcal{F}$ is closed under finite intersections and under supersets, that is

(a) If $F_1, F_2 \in \mathcal{F}$ then $F_1 \cap F_2 \in \mathcal{F}$.

(b) If $F \in \mathcal{F}$ and $F \subseteq F' \subseteq X$ then $F' \in \mathcal{F}$.

$\mathcal{F}_0 \subseteq \mathcal{F}$ is a filter base of $\mathcal{F}$ if each element of $\mathcal{F}$ contains an element of $\mathcal{F}_0$. In that case we define $[\mathcal{F}_0] := \mathcal{F}$. If $\mathcal{F}_0 = \{A\}$, $A \subseteq X$, we abbreviate $[\mathcal{F}_0] = [\{A\}]$ by $[A]$. If $A = \{x_1, \ldots, x_n\}$ we sometimes write $[x_1, \ldots, x_n]$ instead of $[A]$. Particularly $[x]$ is used if $A = \{x\}$.

Let $\mathcal{F}_1$ and $\mathcal{F}_2$ be filters on a set $X$. We say $\mathcal{F}_1$ is finer than $\mathcal{F}_2$ and $\mathcal{F}_2$ is coarser than $\mathcal{F}_1$ if $\mathcal{F}_1 \supseteq \mathcal{F}_2$. A filter $\mathcal{F}$ is an ultrafilter if there does not exist a strictly finer filter than $\mathcal{F}$.

It is a direct consequence that every filter base $\mathcal{F}_0$ of a filter $\mathcal{F}$ is nonempty and we get

$\mathcal{F} = [\mathcal{F}_0] = \{F \subseteq X \mid \exists F_0 \in \mathcal{F}_0 : F_0 \subseteq F\}$

Clearly an arbitrary collection $C$ of nonempty subsets of $X$ is a filter base iff $C$ satisfies

$\forall C_1, C_2 \in C \exists C_0 \in C : C_0 \subseteq C_1 \cap C_2$.

Example 1.3.2 (a) Let $X$ be a topological space and $N(x)$ be the neighbourhood system at a point $x \in X$. Then $N(x)$ is a filter in $X$ which is called the neighbourhood filter at $x$.

(b) Let $X$ be an arbitrary nonempty set and $A \subseteq X$. Then $[A]$ is the filter of all supersets of $A$. It is called the principal filter.

(c) Let $X$ be an arbitrary nonempty set with cardinality greater or equal $\aleph_0$. The set $\mathcal{F} = \{F \mid \exists E \subseteq X : F = X \setminus E, E \text{ finite}\}$ is a filter on $X$. If $X = \mathbb{N}$ we call $\mathcal{F}$ the Fréchet filter on $\mathbb{N}$.

(d) Let $X = \mathbb{R}$ and $\mathcal{C} = \{(a, \infty) \mid a \in \mathbb{R}\}$. $\mathcal{C}$ is a filter base and $\mathcal{F} = [\mathcal{C}]$ is called the Fréchet filter on $\mathbb{R}$. 
Definition 1.3.5 (Filter Convergence) Let \( \mathcal{F} \) be a filter on a topological space \( X \). We say \( \mathcal{F} \) converges to \( x \in X \) if \( N(x) \subseteq \mathcal{F} \), that is, \( \mathcal{F} \) is finer than the neighbourhood filter \( N(x) \). In that case we write \( \mathcal{F} \to x \).

In the subsequent sections we will make use of the following simple statement. A proof can be found in [45, p. 79].

Theorem 1.3.3 If \( E \subseteq X \), then \( x \in \overline{E} \) iff there exists a filter \( \mathcal{F} \) with \( E \in \mathcal{F} \) and \( \mathcal{F} \to x \).

Proof. Let \( x \in \overline{E} \). Then \( \mathcal{B} = \{ U \cap E \mid U \in N(x) \} \) is the base of a filter \( \mathcal{F} \) with \( E \in \mathcal{F} \) and \( N(x) \subseteq \mathcal{F} \), that means, \( \mathcal{F} \to x \).

If \( \mathcal{F} \) is a filter on \( X \) with \( E \in \mathcal{F} \) and \( \mathcal{F} \to x \) we immediately get \( U \cap E \neq \emptyset \) for all \( U \in N(x) \), that is, \( x \in \overline{E} \).

1.3.3 Relationship between Nets and Filters

As one can get similar results for nets as for filters it is not surprising that there exists a tight relationship between these notations. E. g. every net in a topological space \( X \) can be mapped to a filter on \( X \) (and vice versa) such that the net converges to some \( x \in X \) iff the corresponding filter converges to \( x \). The mappings are defined by (see [45, p.81])

Definition 1.3.6 Let \( (x_n)_{n \in D} \) be a net in \( X \) and for each \( n \in D \) let \( B_n = \{ x_m \mid m \geq n \} \) be a tail of \( (x_n) \). We define \( C = \{ B_n \mid n \in D \} \). Then \( C \) is the base of a filter which is called the filter generated by \( (x_n) \).

Now let \( \mathcal{F} \) be a filter on \( X \) and \( D_\mathcal{F} = \{ (x, F) \mid x \in F \in \mathcal{F} \} \). We define a binary relation \( \leq \) on \( D_\mathcal{F} \) by \( (x_1, F_1) \leq (x_2, F_2) \) if \( F_2 \subseteq F_1 \). By means of \( \leq \) \( D_\mathcal{F} \) is a directed set. The mapping \( S \colon D_\mathcal{F} \to X \) with \( S(x, F) = x \) defines a net in \( X \) and is called the net based on \( \mathcal{F} \).

Theorem 1.3.4 Let \( X \) be a topological space and \( x \in X \).

(a) A filter \( \mathcal{F} \) on \( X \) converges to \( x \) iff the net based on \( \mathcal{F} \) converges to \( x \).

(b) A net \( (x_n) \) in \( X \) converges to \( x \) iff the filter generated by \( (x_n) \) converges to \( x \).

Proof. (a) Let \( x \in X, \mathcal{F} \to x \) and \( S \colon D_\mathcal{F} \to X \) be the net based on \( \mathcal{F} \). Let \( U \in N(x) \subseteq \mathcal{F} \) and \( p \in U \). For each \( (q, V) \geq (p, U) \) we get \( S(q, V) = q \in V \subseteq U \). Therefore the net \( S \) converges to \( x \).

Now let \( S \colon D_\mathcal{F} \to X \) be the net based on the filter \( \mathcal{F} \) with \( S_n \to x \in X \). Let \( U \in N(x) \). Then there exists \( (p, F_1) \in D_\mathcal{F} \) such that for each \( (q, F_2) \geq (p, F_1) \) we get \( q \in U \) and we conclude \( F_1 \subseteq U \) by means of the definition of \( D_\mathcal{F} \) and \( \geq \). Therefore we obtain \( U \in \mathcal{F} \) for each \( U \in N(x) \), that is, \( \mathcal{F} \to x \).

(b) Let \( D \) be the domain of \( (x_n) \) and \( x_n \to x \in X \). Let \( U \in N(x) \) be arbitrary. Then there exists \( n \in D \) such that for all \( m \geq n \) we have \( x_m \in U \). We conclude \( B_n \subseteq U \), that is, \( U \in \mathcal{F} \) and as \( U \in N(x) \) was arbitrary we obtain \( \mathcal{F} \to x \).

Let \( \mathcal{F} \) be the filter generated by \( (x_n) \) with \( \mathcal{F} \to x \). Let \( U \in N(x) \subseteq \mathcal{F} \) and \( D \) be the domain of \( (x_n) \). Then there exists \( n \in D \) such that \( B_n \subseteq U \), that is, for all \( m \geq n \) we have \( x_m \in U \). As \( U \in N(x) \) was arbitrary we get \( x_n \to x \).
1.3.4 Closure Operators

In the literature closure operators with different properties are given. We are only interested in so called Kuratowski (or topological) closure operators. There also exist Čech closure operators and so called closure operators which are not a generalization of the named preceding operators (see [43]).

**Definition 1.3.7 (Kuratowski Closure Operator)** Let $X$ be a nonempty set. An operator $^c: \mathcal{P}(X) \rightarrow \mathcal{P}(X)$ is called a Kuratowski (or topological) closure operator if $^c$ has the following properties

(a) $\emptyset^c = \emptyset$.

(b) $A \subseteq A^c$ for all $A \subseteq X$.

(c) $(A \cup B)^c = A^c \cup B^c$ for all $A, B \subseteq X$.

(d) $A^c = (A^c)^c$ for all $A \subseteq X$.

In the following we simply speak of closure operators but we will always mean the Kuratowski closure operator.

**Lemma 1.3.5** Let $X$ be a nonempty set and $^c: \mathcal{P}(X) \rightarrow \mathcal{P}(X)$ be a closure operator. Then $\mathcal{T} = \{X \setminus A | A \subseteq X, A = A^c\}$ defines a topology on $X$ with the property $\overline{A^c} = A^c$ for all $A \subseteq X$. So $^c$ is the topological closure of each subset of $X$ with respect to the topology $\mathcal{T}$.

**Proof.** Let $\mathcal{A} = \{A \subseteq X | A = A^c\}$. We have to show that arbitrary intersections and finite unions of elements of $\mathcal{A}$ are in $\mathcal{A}$ and that $\emptyset$ and $X$ are in $\mathcal{A}$. Because of (a) and (b) we have $\emptyset, X \in \mathcal{A}$. If $A \subseteq B \subseteq X$, we conclude $B = A \cup (B \setminus A)$ and using (c) we get $A^c \subseteq B^c$.

Now let $I$ be an index set and $A_i \in \mathcal{A}$ for each $i \in I$. Because of $\bigcap_{i \in I} A_i \subseteq A_j$ for each $j \in I$ we conclude $(\bigcap_{i \in I} A_i)^c \subseteq A_j^c$ for each $j \in I$, that is, $(\bigcap_{i \in I} A_i)^c \subseteq \bigcap_{i \in I} A_i^c = \bigcap_{i \in I} A_i$. Using (b) we get $(\bigcap_{i \in I} A_i)^c \in \mathcal{A}$.

Let $A_1, \ldots, A_n \in \mathcal{A}$. Using (c) we obtain $(A_1 \cup \ldots \cup A_n)^c = A_1 \cup \ldots \cup A_n$, that is, $A_1 \cup \ldots \cup A_n \in \mathcal{A}$.

So altogether $\mathcal{T}$ is a topology on $X$. Let $A \subseteq X$. We show $\overline{A^c} = A^c$. Because of (d) we know that $A^c \in \mathcal{A}$. As $\overline{A^c} = \bigcap \{B | A \subseteq B \in \mathcal{A}\}$ and for each $B \in \mathcal{A}$ with $A \subseteq B$ we have $A^c \subseteq B^c = B$ we immediately get what we claimed.

1.4 Convergence Spaces

Convergence spaces in net and filter notation are a generalization of topological spaces in the sense that every topological space can be represented as a convergence space and the induced topology (see below) of the corresponding convergence space is the original topology. The embedding used is not surjective, that is, there exist convergence spaces which are not in the image of that embedding (see [17]). We expound only the part of the theory of convergence spaces which is needed in subsequent sections. For more information on convergence spaces in filter notation see [4, 17, 25]. The net notation of convergence spaces is new and is inspired by the net notation of convergence classes (see next section). First we investigate convergence spaces (CS) in net notation.
1.4.1 Convergence Spaces in Net Notation

Definition 1.4.1 (Convergence Space in Net Notation) Let $X$ be a nonempty set. The pair $(X, (S_s)_{s \in X})$ is called a convergence space in net notation if for each $s \in X$, $S_s$ is a collection of nets with the following properties:

(a) If $S : D \rightarrow X$ is a constant net, that is, $S_n = s \in X$ for all $n \in D$, then $S \in S_s$.

(b) If $S \in S_s$ and $T$ is a subnet of $S$, then $T \in S_s$.

We will sometimes use another notation for convergence spaces in net notation. We write $S \downarrow s$ if $S \in S_s$ and speak of the convergence space $(X, \downarrow)$ instead of $(X, S)$. We interprete $\downarrow$ as a relation or a subset of the cartesian product of the set of all filters on $X$ and the set $X$.

If $S \in S_s$ we say $S$ converges to $s$.

A subset $O \subseteq X$ is open in the induced topology of a convergence space $(X, S)$ in net notation if $S \downarrow s \in O$ always implies that there exists $n \in D$ with $S_m \in O$ for all $m \geq n$. □

We prove that the induced topology really is a topology and that every topological space is representable as a convergence space in net notation such that the induced topology coincides with the original topology:

Lemma 1.4.1 The induced topology of a convergence space $(X, S)$ in net notation is a topology on $X$.

Proof. Let $(O_i)_{i \in I}$ be a family of open sets with respect to the induced topology of $(X, S)$. Let $O = \bigcup_{i \in I} O_i$ and $S \downarrow s \in O$. Then there exists $i \in I$ with $s \in O_i$ and there exists $n \in D$ with $S_n \in O_i \subseteq O$ for all $m \geq n$. As $s \in O$ was arbitrary $O$ is an element of the induced topology.

Now let $O_1, O_2 \subseteq X$ be open sets of the induced topology of $(X, S)$. Let $O = O_1 \cap O_2$ and $S \downarrow s \in O$. Then there exist $n_1, n_2 \in D$ such that $S_n \in O_i$ for all $n \geq n_i, i = 1, 2$. As $D$ is directed there exists $n_0 \in D$ with $n_0 \geq n_1, n_2$ and we have $S_n \in O$ for all $n \geq n_0$. Because $s \in O$ was arbitrary we conclude that $O$ is an element of the induced topology. ■

Lemma 1.4.2 Every topological space $(X, \mathcal{T})$ is representable as a convergence space $(X, S)$ in net notation such that the induced topology is $\mathcal{T}$.

Proof. We define $S \in S_s$ iff $S : D \rightarrow X$ is a net with $S_n \rightarrow s$ with respect to $\mathcal{T}$. Obviously $S_s, s \in X$, satisfies the conditions listed in the definition of a convergence space in net notation. So we can choose $S = (S_s)_{s \in X}$.

We show that the induced topology of $(X, S)$ is $\mathcal{T}$. First let $O$ be open with respect to $\mathcal{T}$ and let $S \downarrow s \in O$. By definition we have $S_n \rightarrow s$ with respect to $\mathcal{T}$ and as $O \in \mathcal{T}$ we conclude that there exists $n \in D$ with $S_m \in O$ for all $m \geq n$. Because $s \in O$ was arbitrary $O$ is an open set of the induced topology of $(X, S)$.

Next let $O$ be open with respect to the induced topology. Suppose $O \notin \mathcal{T}$, that is, $X \setminus O$ is not closed with respect to $\mathcal{T}$. Then there exists $s \in \overline{\bigcap \mathcal{T} \cap O}$. Using Theorem 1.3.1 we conclude that there exists a net $S : D \rightarrow X$ with $S_n \in X \setminus O$ for all $n \in D$ and $S_n \rightarrow s \in O$. But the last statement implies that there exists $n \in D$ with $S_m \in O$ for all $m \geq n$ which is a contradiction. So we conclude $O \in \mathcal{T}$. ■
**Definition 1.4.2** Let \((X, \mathcal{F})\) be a topological space. We define the induced convergence space in net notation by \(S \downarrow \emptyset \equiv s\) if \(S\) is a net in \(X\) such that \(S_n \to s\) with respect to \(\mathcal{F}\). An induced convergence space in net notation is sometimes also called a topological convergence space in net notation.

1.4.2 Convergence Spaces in Filter Notation

**Definition 1.4.3** (Convergence Space in Filter Notation) Let \(X\) be a nonempty set. The pair \((X, (\mathcal{F}_x)_{x \in X})\) is called a convergence space in filter notation if for each \(x \in X\) \(\mathcal{F}_x\) is a collection of filters with the following properties:

(a) \([x] \in \mathcal{F}_x\), that is, the principal filter \([x]\) is an element of \(\mathcal{F}_x\) (point filter axiom).

(b) If \(\mathcal{A} \in \mathcal{F}_x\) and \(\mathcal{B} \supseteq \mathcal{A}\) is a filter on \(X\) then \(\mathcal{B} \in \mathcal{F}_x\) (closure under superfilters).

One uses sometimes another notation for convergence spaces in filter notation. One writes \(\mathcal{A} \downarrow x\) if \(\mathcal{A} \in \mathcal{F}_x\) and speaks of the convergence space \((X, \downarrow)\) instead of \((X, \mathcal{F})\). We interpret \(\downarrow\) as a relation or a subset of the cartesian product of the set of all filters on \(X\) and the set \(X\). If \(\mathcal{A} \in \mathcal{F}_x\) we say \(\mathcal{A}\) converges to \(x\).

A convergence space \((X, \mathcal{F})\) in filter notation is pointed if \(\bigcap \mathcal{F}_x \subseteq \{x\}\) for each \(x \in X\).

A subset \(O \subseteq X\) is open in the induced topology of a convergence space \((X, \mathcal{F})\) in filter notation if \(\mathcal{A} \downarrow x \in O\) always implies \(O \in \mathcal{A}\) \((\mathcal{A} \supseteq \{O\})\).

We prove some properties of convergence spaces in filter notation. In the following let \((X, \mathcal{F})\) always be a convergence space in filter notation.

**Lemma 1.4.3** \(\bigcap \mathcal{F}_x\) is a coarser filter than each \(\mathcal{A} \in \mathcal{F}_x\). We have \(x \in F\) for each \(F \in \bigcap \mathcal{F}_x\).

**Proof.** Let \(\mathcal{A} = \bigcap \mathcal{F}_x\) and \(F, F_1, F_2 \in \mathcal{A}\). So \(F, F_1, F_2 \in \mathcal{B}\) for each \(\mathcal{B} \in \mathcal{F}_x\). We get \(F_1 \cap F_2 \in \mathcal{B}\) and \(G \in \mathcal{B}\) for each filter \(G \supseteq F\) on \(X\) and \(\mathcal{B} \in \mathcal{F}_x\). It follows that \(F_1 \cap F_2, G \in \mathcal{A}\) and we conclude that \(\mathcal{A}\) is a filter. Of course is \(\mathcal{A} \supseteq \mathcal{B}\) for each filter \(\mathcal{B} \in \mathcal{F}_x\). As \([x] \in \mathcal{F}_x\) we get \(\mathcal{A} \subseteq \mathcal{F}_x\) which implies \(x \in A\) for each \(A \in \mathcal{A}\).

**Lemma 1.4.4** The induced topology of a convergence space \((X, \mathcal{F})\) in filter notation is a topology on \(X\).

**Proof.** Let \((O_i)_{i \in I}\) be a family of open sets of the induced topology of the convergence space. Let \(O = \bigcup_{i \in I} O_i \) and \(\mathcal{A} \downarrow x \in O\). Then there exists \(i \in I\) with \(x \in O_i\) and we get \(O_i \in \mathcal{A}\). As \(\mathcal{A}\) is a filter we conclude \(O \in \mathcal{A}\) and as \(x \in O\) was arbitrary \(O\) is an element of the induced topology.

Now let \(O_1, O_2 \subseteq X\) be open sets of the induced topology of the convergence space. Let \(O = O_1 \cap O_2\) and \(\mathcal{A} \downarrow x \in O\). We conclude \(O_1, O_2 \in \mathcal{A}\) and as \(\mathcal{A}\) is a filter we get \(O \in \mathcal{A}\). So \(O\) is open in the induced topology.

**Lemma 1.4.5** Every topological space \((X, \mathcal{F})\) is representable as a convergence space \((X, \mathcal{F})\) in filter notation such that the induced topology is \(\mathcal{F}\).

**Proof.** Let \(\mathcal{F}_x\) be the set of all filters \(\mathcal{A} \supseteq N(x)\) where \(N(x)\) is the neighbourhood filter at \(x\) with respect to the topology \(\mathcal{F}\). Obviously \(\mathcal{F}_x\) has the properties listed in the definition of a convergence space in filter notation. So we can choose \(\mathcal{F} = (\mathcal{F}_x)_{x \in X}\).
First let $O \in \mathcal{T}$ and $A \downarrow x \in O$. We immediately get $O \in N(x) \subseteq A$ and as $x$ was arbitrary $O$ is open with respect to the induced topology.

Next let $O$ be open with respect to the induced topology. Then $N(x) \downarrow x \in O$ implies $O \in N(x)$. Therefore there exists an open set $O_x \in N(x) \cap \mathcal{T}$ with $O_x \subseteq O$ and we get $O = \bigcup_{x \in O} O_x \in \mathcal{T}$. So $O$ is open with respect to $\mathcal{T}$.

**Definition 1.4.4** Let $(X, \mathcal{T})$ be a topological space. We define the induced convergence space in filter notation by $A \downarrow O x$ iff $A \supseteq N(x)$ is a filter on $X$. An induced convergence space in filter notation is sometimes also called a topological convergence space in filter notation.

**1.5 Convergence Classes**

As mentioned in the introduction, one often has a set $X$ and a notion of which sequences, nets or filters should converge in $X$ to a point $x \in X$ but one does not know if there exists a topology on $X$ such that exactly these sequences, nets or filters converge with respect to the topology. Convergence classes in net or filter notation induce necessary and sufficient conditions on the set of nets or filters which are to converge to a given point $x \in X$ such that there exists a topology on $X$, and a net or filter converges to $x$ with respect to that topology iff the net or filter together with $x$ is an element of the corresponding convergence class.

First we will present convergence classes in net notation with some motivating statements and reasons for defining them in that way. The net notation of convergence classes is not new. The proofs of the theorems about convergence classes in net notation can also be found in [26].

After that we will present convergence classes in filter notation. This part of the presentation is new and will also be motivated by the reasons for defining these classes in that way. All proofs will be given.

Some questions arise after the main proofs on convergence classes in filter notation. We will see that all the questions can be nicely answered. We will be able to construct a convergence class in filter notation from a convergence class in net notation and vice versa. The induced topologies are identical (see below).

**1.5.1 Convergence Classes in Net Notation**

In what follows we need to direct a cartesian product of directed sets. So let $D$ be an arbitrary but nonempty set and $E_m$ be directed by $\leq_m$ for each $m \in D$. Elements of $\prod_{m \in D} E_m$ can be viewed as functions $f : D \rightarrow \bigcup_{m \in D} E_m$ with $f(m) \in E_m$ for all $m \in D$. We direct $f, g \in \prod_{m \in D} E_m$ by $f \leq g$ iff $f(m) \leq_m g(m)$ for all $m \in D$. Obviously $\prod_{m \in D} E_m$ becomes a directed set in that way.

The following theorem shows a characteristic property of net convergence in a topological space. The proof is from [26, p. 69].

**Theorem 1.5.1 (Iterated Limits)** Let $D$ be a directed set, let $E_m$ be a directed set for each $m \in D$, let $F = D \times \prod_{m \in D} E_m$, and let $F' = \{(n,m) \mid n \in D, m \in E_n\}$. Let $R : F \rightarrow F'$ be the function with $R(m,f) = (m,f(m))$ for each $(m,f) \in F$. Let $(X, \mathcal{T})$ be a topological space and let $S : F' \rightarrow (X, \mathcal{T})$ be a function. Then $S \circ R$ converges to $\lim_{m,n} S(m,n)$ whenever the iterated limit exists.
1.5. CONVERGENCE CLASSES

**Proof.** Let \( \lim_{n} \lim_{m} S(m, n) = s \) and \( O \in N(s) \cap \mathcal{T} \). We choose \( m \in D \) such that \( \lim_{n} S(p, n) \in O \) for all \( p \geq m \). For each \( p \geq m \) we choose \( f(p) \in E_p \) such that \( S(p, n) \in O \) for all \( n \geq f(p) \). For all other \( p \in D \) we choose \( f(p) \in E_p \) arbitrarily. So \( f \in \bigtimes_{m \in D} E_m \) and we get for each \( (p, g) \geq (m, f) \), that is, \( p \geq m \) and \( g(n) \geq f(n) \) for all \( n \in D \) that \( S \circ R(p, g) = S(p, g(p)) \in O \).

The iterated limits theorem is the motivation for condition (d) in the following definition of convergence classes in net notation. Conditions (a) and (b) are a direct consequence of well-known properties of net convergence in a topological space.

**Definition 1.5.1 (Convergence Class in Net Notation)** Let \( X \) be an arbitrary non-empty set. We say \( \mathcal{C} \) is a convergence class for \( X \) in net notation if it is a set of pairs each consisting of a net and a point of \( X \) such that the conditions listed below are satisfied. Instead of \( (S, s) \in \mathcal{C} \) we also use the notation \( S \) converges \((\mathcal{C})\) to \( s \) or \( \lim_{n} S_{n} = s(\mathcal{C}) \).

(a) If \( S: D \rightarrow X \) is a constant net in \( X \), that is, \( S_{n} = s \) for all \( n \in D \), then \((S, s) \in \mathcal{C}\).

(b) If \((S, s) \in \mathcal{C}\) and \( T \) is a subnet of \( S \), then \((T, s) \in \mathcal{C}\).

(c) If \((S, s) \notin \mathcal{C}\), then there exists a subnet \( T \) of \( S \) such that for each subnet \( R \) of \( T \) we have \((R, s) \notin \mathcal{C}\).

(d) Let \( D \) be a directed set, let \( E_{m} \) be a directed set for each \( m \in D \), \( F = D \times \bigtimes_{m \in D} E_{m} \), and let \( F' = \{(m, n) | n \in D, m \in E_{n}\} \). Let \( R: F \rightarrow F' \) by \( R(m, f) = (m, f(m)) \) for each \((m, f) \in F \) and let \( S: F' \rightarrow X \) be a function. If \( \lim_{n} S(m, n) = s(\mathcal{C}) \), then \((S \circ R, s) \in \mathcal{C}\).

If a net \( S: D \rightarrow X \) does not converge in the topological space \((X, \mathcal{T})\) to \( s \in X \) there must exist \( U \in N(s) \) and a cofinal subset \( D' \subseteq D \) such that \( S_{n} \in X \setminus U \) for all \( n \in D' \). This fact is the reason for stipulating condition (c) in the above definition.

We now formulate the main theorem for convergence classes in net notation. The proof is a corrected version of that given in [26, Theorem 9, p. 74], in that the final part of the proof in [26] contains an error.

**Theorem 1.5.2** Let \( \mathcal{C} \) be a convergence class in net notation for a nonempty set \( X \). For each \( A \subseteq X \) let \( A^{\mathcal{C}} = \{ s \in X \mid \exists \text{net } S \text{ in } A \colon (S, s) \in \mathcal{C} \} \). Then \( ^{\mathcal{C}} \) is a closure operator, that is, defines a topology \( \mathcal{T} \) on \( X \) and we have \((S, s) \in \mathcal{C} \) iff \( S_{n} \rightarrow s \) with respect to \( \mathcal{T} \).

**Proof.** First we show that \( ^{\mathcal{C}} \) is a closure operator.

(i) We have \( \emptyset^{\mathcal{C}} = \emptyset \) as there does not exist a net in \( \emptyset \) by definition.

(ii) We show \( A \subseteq A^{\mathcal{C}} \). Let \( s \in A \) and \( S \) the constant net with \( S_{n} = s \) for all \( n \). Using condition (a) in the definition of \( \mathcal{C} \) we get \((S, s) \in \mathcal{C}\) and conclude \( s \in A^{\mathcal{C}} \).

(iii) We show \((A \cup B)^{\mathcal{C}} = A^{\mathcal{C}} \cup B^{\mathcal{C}} \). Let \( s \in A^{\mathcal{C}} \). Then there exists a net \( S \) in \( A \) with \((S, s) \in \mathcal{C}\). As \( S \) is also a net in \( A \cup B \) we conclude \( s \in (A \cup B)^{\mathcal{C}} \). In the same way one proves \( B^{\mathcal{C}} \subseteq (A \cup B)^{\mathcal{C}} \). Now let \( s \in (A \cup B)^{\mathcal{C}} \). Then there exists a net \( S: D \rightarrow (A \cup B) \) with \((S, s) \in \mathcal{C}\). Let \( D_{A} = \{ n \in D \mid S_{n} \in A \} \) and \( D_{B} = \{ n \in D \mid S_{n} \in B \} \). As \( D = D_{A} \cup D_{B} \) at least one of \( D_{A} \) or \( D_{B} \) is cofinal in \( D \) and \( S_{A} = S|_{D_{A}} \) or \( S_{B} = S|_{D_{B}} \) is a subnet of \( S \). Using condition (b) in the definition of \( \mathcal{C} \) we get \((S_{A}, s) \in \mathcal{C} \) or \((S_{B}, s) \in \mathcal{C} \). So we have \( s \in A^{\mathcal{C}} \cup B^{\mathcal{C}} \).
(iv) We show \((A')^c = A'\). Using (ii) we have \(A' \subseteq (A')^c\). Let \(s \in (A')^c\). Then there exists a net \(T: D \rightarrow A'\) with \((T, s) \in \mathcal{C}\) and for each \(m \in D\) there exists a net \(S(m, \bullet): E_m \rightarrow A\) with \((S(m, \bullet), T_m) \in \mathcal{C}\). We apply condition (d) in the definition of \(\mathcal{C}\) and get a net \(S \circ R\) in \(A\) with \((S \circ R, s) \in \mathcal{C}\). So we can conclude \(s \in A'\).

Now we prove the equivalence statement in our claim.

(v) We show \((S, s) \in \mathcal{C}\) implies \(S_n \rightarrow s\) with respect to \(\mathcal{T}\). Suppose \(S_n \rightarrow s\) with respect to \(\mathcal{T}\) is not true and let \(D\) be the domain of \(S\), that is, a directed set. Then there exists \(O \in N(s) \cap \mathcal{T}\) and for each \(n \in D\) there exists \(m_n \in D\) with \(m_n \geq n\) and \(S_{m_n} \in X \setminus O\). We obtain that \(D' = \{m_n | n \in D\}\) is cofinal in \(D\) and \(T = S[D']\) is a subnet of \(S\) in \(X \setminus O\). Using condition (b) in the definition of \(\mathcal{C}\) we get \((T, s) \in \mathcal{C}\) and \(s \in (X \setminus O)^c = X \setminus O\) which contradicts \(s \in O\). We conclude \(S_n \rightarrow s\) with respect to \(\mathcal{T}\).

(vi) We show \(S_n \rightarrow s\) with respect to \(\mathcal{T}\) implies \((S, s) \in \mathcal{C}\). Suppose \((S, s) \notin \mathcal{C}\). Because of condition (c) in the definition of \(\mathcal{C}\) there exists a subnet \(T: D \rightarrow X\) of \(S\) such that for each subnet \(R \subseteq T\) we have \((R, s) \notin \mathcal{C}\). Let \(D_m = \{n \in D | n \geq m\}\) and \(A_m = T[D_m]\). As \(D_m\) is cofinal in \(D\) we have that \(T[D_m]\) is a subnet of \(T\) which converges to \(s\) with respect to \(\mathcal{T}\). Using Theorem 1.3.1 we get \(s \in (A'_m)^c\) for each \(m \in D\). Therefore we obtain for each \(m \in D\) a net \(U(m, \bullet): E_m \rightarrow A_m\) with \((U(m, \bullet), s) \in \mathcal{C}\). We apply condition (d) in the definition of \(\mathcal{C}\). Let \(F\) and \(R\) as defined in condition (d). Then we get \((U \circ R, s) \in \mathcal{C}\). Because we have \(U \circ R(m, f) \in A_m\) there exists \(n_{m,f} \in D_m\) with \(U \circ R(m, f) = T_{n_{m,f}}\) for all \((m, f) \in F\). We define \(\varphi: F \rightarrow D\) by \(\varphi(m, f) = n_{m,f}\) and get \(U \circ R = T \circ \varphi\). We obtain that \(\varphi\) is cofinal as for all \(m \in D\) and \(f \in F\) we have \(\varphi(m, f) = n_{m,f} \geq m\). So \(U \circ R\) is a subnet of \(T\) and \((U \circ R, s) \in \mathcal{C}\) which is a contradiction to our assumption. We conclude \((S, s) \in \mathcal{C}\).

\textbf{Remark 1.5.3} It is an immediate consequence of the definition that each convergence class in net notation is a convergence space in net notation.

\subsection{1.5.2 Convergence Classes in Filter Notation}

Instead of nets we will now use filters on a given nonempty set \(X\) to define classes which describe exactly which filters should converge to which point of \(X\). We will give necessary and sufficient conditions such that there exists a topology \(\mathcal{T}\) on \(X\) with the property that exactly those filters converge to a point \(x \in X\) with respect to \(\mathcal{T}\) which are stipulated to converge to \(x\) with respect to the given class.

Before we define convergence classes in filter notation we want to give a characteristic property of filters converging to a point in a topological space. It is the adequate replacement of the iterated limits theorem for nets.

\textbf{Theorem 1.5.4} Let \(D\) be an index set, \((\mathcal{F}_d)_{d \in D}\) a family of filters on a topological space \((X, \mathcal{T})\), \((A_d)_{d \in D}\) a family of subsets of \(X\) such that \(\{A_d | d \in D\}\) is a filter base and let \(S = \{s_d | d \in D\} \subseteq X, s \in X\). For all \(d \in D\) we require

\begin{equation}
A_d \in \mathcal{F}_d, \mathcal{F}_d \rightarrow s_d \text{ and } \forall s' \in S \exists d' \in D: A_{d'} \in \mathcal{F}_{d'}, \mathcal{F}_{d'} \rightarrow s'.
\end{equation}

Let \(\mathcal{F}\) be a filter with \(S \subseteq \mathcal{F}\) and \(\mathcal{F} \rightarrow s\). Then there exists a filter \(\mathcal{G}\) on \(X\) with \(A_d \in \mathcal{G}\) for all \(d \in D\) and \(\mathcal{G} \rightarrow s\).

\textbf{Proof.} Let the premises of our claim be satisfied. From (1.1) we conclude \(s_d \in S \subseteq \mathcal{F}_{d'}\) for all \(d \in D\). Because of \(S \subseteq \mathcal{F}\) and \(\mathcal{F} \rightarrow s\) we obtain \(s \in \mathcal{S}\). Particularly we have \(s \in \mathcal{F} \subseteq \mathcal{A}_d\).
1.5. CONVERGENCE CLASSES

for all $d \in D$. Let $\mathcal{B} = \{A_d \cap U \mid d \in D, U \in N(s)\}$. Then $\mathcal{B}$ is a filter base because for all $d, d' \in D, U, U' \in N(s)$ there exists $d'' \in D$ with $A_{d''} \subseteq A_d \cap A_{d'}$ and we obtain

$$\emptyset \neq A_{d''} \cap (U \cap U') \subseteq (A_d \cap A_{d'}) \cap (U \cap U') = (A_d \cap U) \cap (A_{d'} \cap U').$$

Let $\mathcal{G}$ be the filter with base $\mathcal{B}$. We get $N(s) \subseteq \mathcal{G}$ and $A_d \in \mathcal{G}$ for all $d \in D$. Particularly we obtain $\mathcal{G} \rightarrow s$.

Condition (d) in the definition of convergence classes in filter notation is inspired by Theorem 1.5.4. Condition (a) follows by considering the filter generated by a constant net. Condition (b) follows from the fact that every filter $\mathcal{G}$ finer than $\mathcal{F}$ with $\mathcal{F} \rightarrow x$ also converges to $x$. Because our aim is that $(\mathcal{F}, s)$ be an element of a convergence class $\mathcal{C}$ in filter notation iff $\mathcal{F} \rightarrow s$ with respect to an unknown topology $\mathcal{T}$ condition (c) is another necessary requirement on $\mathcal{C}$ to reach that aim (compare with the proof of Lemma 1.5.14). A similar definition of classes in filter notation can be found in [43]. There conditions (a) – (c) of the following definition are used only.

**Definition 1.5.2 (Convergence Class in Filter Notation)** Let $X$ be a nonempty set. A convergence class $\mathcal{C}$ for $X$ in filter notation is a subset of the cartesian product of all filters on $X$ and $X$ such that the conditions listed below are satisfied. We say $\mathcal{F}$ converges (belongs) to $s$ or $\mathcal{F} \rightarrow s(\mathcal{C})$ if $(\mathcal{F}, s) \in \mathcal{C}$.

(a) Let $s \in X$ and $\mathcal{F} = \{F \subseteq X \mid s \in F\} = [s]$ be a principal ultrafilter on $X$. Then $(\mathcal{F}, s) \in \mathcal{C}$.

(b) Let $(\mathcal{F}, s) \in \mathcal{C}$. For each filter $\mathcal{G} \supseteq \mathcal{F}$ on $X$ we demand $(\mathcal{G}, s) \in \mathcal{C}$.

(c) If $(\mathcal{F}, s) \notin \mathcal{C}$ there exists a filter $\mathcal{F}' \supseteq \mathcal{F}$ such that for each filter $\mathcal{G} \supseteq \mathcal{F}'$ we have $(\mathcal{G}, s) \notin \mathcal{C}$.

(d) Let $D$ be an index set, $(\mathcal{F}_d)_{d \in D}$ a family of filters on $X$, $(A_d)_{d \in D}$ a family of subsets of $X$ such that $\{A_d \mid d \in D\}$ is a filter base and let $S = \{s_d \mid d \in D\} \subseteq X, s \in X$. For all $d \in D$ we require

$$A_d \in \mathcal{F}_d, (\mathcal{F}_d, s_d) \in \mathcal{C} \quad \text{and} \quad A_d \subseteq S \exists d' \in D: A_d \in \mathcal{F}_{d'}, (\mathcal{F}_{d'}, s') \in \mathcal{C}.$$

Let $\mathcal{F}$ be a filter with $S \subseteq \mathcal{F}$ and $(\mathcal{F}, s) \in \mathcal{C}$. Then there exists a filter $\mathcal{G}$ on $X$ with $A_d \in \mathcal{G}$ for all $d \in D$ and $(\mathcal{G}, s) \in \mathcal{C}$.

The main theorem about convergence classes in filter notation is an adequate reformulation of Theorem 1.5.2.

**Theorem 1.5.5** Let $X$ be a nonempty set, $\mathcal{C}$ be a convergence class for $X$ in filter notation and for each $A \subseteq X$ let $A^c = \{s \in X \mid \exists \text{filter } \mathcal{F} \text{ on } X: A \in \mathcal{F}, (\mathcal{F}, s) \in \mathcal{C}\}$. Then $\mathcal{C}$ is a closure operator, that is, defines a topology $\mathcal{T}$ on $X$ and we have $(\mathcal{F}, s) \in \mathcal{C}$ iff $\mathcal{F} \rightarrow s$ with respect to $\mathcal{T}$.

**Proof.** First we show that $\mathcal{C}$ is a closure operator.

(i) We have $\emptyset^c = \emptyset$ as $\emptyset \in \mathcal{F}$ is false for each filter $\mathcal{F}$ on $X$.

(ii) We show $A \subseteq A^c$. Let $s \in A$. We have $A \in \mathcal{F} = \{F \subseteq X \mid s \in F\} = [s]$. Using condition (a) in the definition of $\mathcal{C}$ we get $(\mathcal{F}, s) \in \mathcal{C}$ and conclude $s \in A^c$. 


(iii) We show \((A \cup B)^c = A^c \cup B^c\). Let \(s \in A^c\). Then there exists a filter \(\mathcal{F}\) on \(X\) with \(A \in \mathcal{F}\) and \((\mathcal{F}, s) \in \mathfrak{C}\). As \(\mathcal{F}\) is a filter we conclude \((A \cup B) \in \mathcal{F}\) and therefore \(s \in (A \cup B)^c\). In the same way one proves \(B^c \subseteq (A \cup B)^c\). Now let \(s \in (A \cup B)^c\). Then there exists a filter \(\mathcal{F}\) on \(X\) with \((A \cup B) \in \mathcal{F}\) and \((\mathcal{F}, s) \in \mathfrak{C}\). Let \(\mathcal{B}_1 = \{F \cap A | F \in \mathcal{F}\}\) and \(\mathcal{B}_2 = \{F \cap B | F \in \mathcal{F}\}\). Assume \(\emptyset \in \mathcal{B}_1 \cap \mathcal{B}_2\). We obtain \(F_1, F_2 \in \mathcal{F}\) with \(F_1 \cap A = F_2 \cap B = \emptyset\) and conclude \((F_1 \cap F_2) \cap (A \cup B) = \emptyset\) which contradicts the fact that \((A \cup B), (F_1 \cap F_2) \in \mathcal{F}\). Therefore \(\mathcal{B}_1\) or \(\mathcal{B}_2\) is a base of a filter \(\mathcal{F}' \supseteq \mathcal{F}\) with \(A \in \mathcal{F}'\) or \(B \in \mathcal{F}'\). Using condition (b) in the definition of \(\mathfrak{C}\) we conclude \((\mathcal{F}', s) \in \mathfrak{C}\). So \(s \in A^c \cup B^c\).

(iv) We show \((A')^c = A^c\). Using (ii) we have \(A^c \subseteq (A')^c\). Let \(s \in (A')^c\). Then there exists a filter \(\mathcal{F}\) with \(A^c \in \mathcal{F}\) and \((\mathcal{F}, s) \in \mathfrak{C}\). For each \(a \in A^c\) there exists a filter \(\mathcal{F}_a\) with \(A \in \mathcal{F}_a\) and \((\mathcal{F}_a, a) \in \mathfrak{C}\). We use condition (d) in the definition of \(\mathfrak{C}\). Let \(D = A^c, A_n = A, s_n = a\) for all \(a \in D\) as well as \(S = A^c\). Then the premises of condition (d) are satisfied and there exists a filter \(\mathfrak{G}\) on \(X\) with \(A \in \mathfrak{G}\) and \((\mathfrak{G}, s) \in \mathfrak{C}\). We conclude \(s \in A^c\).

Now we prove the equivalence statement in our claim.

(v) We show \((\mathfrak{F}, s) \in \mathfrak{C}\) implies \(\mathcal{F} \rightarrow s\) with respect to \(\mathfrak{F}\). Suppose \(\mathcal{F} \rightarrow s\) with respect to \(\mathfrak{F}\) is not true. Then there exists \(U \in N(s) \cap \mathcal{F}\) with \(U \notin \mathfrak{F}\). Let \(\mathcal{B} = \{F : F \cap (X \setminus U) \neq \emptyset | F \in \mathcal{F}\}\).

As \(F \cap (X \setminus U) \neq \emptyset\) for all \(F \in \mathcal{F}\) (otherwise there would exist \(F \in \mathcal{F}\) with \(F \subseteq U\), that is, \(U \in \mathcal{F}\), which is a contradiction) \(\mathcal{B}\) is a base of a filter \(\mathcal{B}' \supseteq \mathcal{F}'\). Using condition (b) in the definition of \(\mathfrak{C}\) it follows \((\mathfrak{F}', s) \in \mathfrak{C}\). Because of \(B \subseteq X \setminus U\) for all \(B \in \mathcal{B}\) we get \(s \in (X \setminus U)^c\) in contradiction to \(X \setminus U = (X \setminus U)^c\) and \(s \in U\). So \(\mathcal{F} \rightarrow s\) with respect to \(\mathfrak{F}\) is true.

(vi) We show \(\mathcal{F} \rightarrow s\) with respect to \(\mathfrak{F}\) implies \((\mathfrak{F}, s) \in \mathfrak{C}\). Suppose \((\mathfrak{F}, s) \notin \mathfrak{C}\). Because of condition (c) in the definition of \(\mathfrak{C}\) there exists a filter \(\mathcal{F}' \supseteq \mathfrak{F}\) such that for all filters \(\mathfrak{G} \supseteq \mathcal{F}'\) we have \((\mathfrak{G}, s) \notin \mathfrak{C}\). We have \(N(s) \subseteq \mathfrak{F} \subseteq \mathcal{F}'\). Therefore we obtain \(s \in \bigcap_{F \in \mathcal{F}} \overline{F}\). The definition of \(\overline{\cdot}\) yields for each \(F \in \mathcal{F}'\) a filter \(\mathcal{F}_F\) with \(F \in \mathcal{F}_F\) and \((\mathcal{F}_F, s) \in \mathfrak{C}\). We use condition (d) in the definition of \(\mathfrak{C}\). Let \(D = \mathcal{F}', A_F = F, s_F = s\) for each \(F \in \mathcal{F}'\). Due to condition (a) in the definition of \(\mathfrak{C}\) we get for the principal ultrafilter \(\mathfrak{F}' = [s]\) the property \((\mathfrak{F}' = [s], s) \in \mathfrak{C}\). So the premises of condition (d) are satisfied and we obtain a filter \(\mathfrak{G}\) on \(X\) with \(F \in \mathfrak{G}\) for all \(F \in \mathcal{F}'\) and \((\mathfrak{G}, s) \in \mathfrak{C}\). Particularly \(\mathfrak{G} \supseteq \mathcal{F}'\) which is a contradiction to our assumption. So we conclude \((\mathfrak{F}, s) \in \mathfrak{C}\).

\[\square\]  

\textbf{Remark 1.5.6} It is an immediate consequence of the definition that each convergence class in filter notation is a convergence space in filter notation.

After one has proved the above theorem some questions directly arise.

- Every convergence class induces a topology on the underlying space. On the other hand this topology induces a convergence space (see Lemma 1.4.2 and Lemma 1.4.5). Is this convergence space once again a convergence class in net or filter notation ? If that is true, is this convergence class identical to the original convergence class ?

- Is every convergence class in filter notation a pointed convergence space ? Is the induced topology of the convergence space identical to the induced topology of the convergence class ?

- Can one transform each convergence class in net notation in a convergence class in filter notation (and vice versa) such that both induce the same topology ?

To all these questions, we will give satisfactory answers in the subsequent subsections.
1.5. CONVERGENCE CLASSES

In the following we denote the induced topology of a convergence space \((X, S)\) or \((X, \mathcal{I})\) by \(TX\). Each convergence class will also be interpreted as a convergence space with special properties. Therefore if we speak of a convergence class we will sometimes use the notations of convergence spaces to denote elements of the convergence class.

1.5.3 Properties of Convergence Classes in Net Notation

Lemma 1.5.7 Let \(X\) be a nonempty set and \(\mathcal{C}\) a convergence class for \(X\) in net notation. Then the induced topology \(\mathcal{I}\) of the convergence class coincides with \(TX\), that is, \(TX = \mathcal{I}\).

Proof. Let \(O \in TX\) be an open set of the underlying convergence space in net notation. We show that \(X \setminus O\) is closed with respect to \(\mathcal{I}\). Suppose that there exists \(s \in (X \setminus O)^c \cap O\). Then there exists a net \(S: D \to X \setminus O\) with \(S \downarrow s \in O\). As \(O \in TX\) there exists \(n \in D\) such that \(S_n \in O\) for all \(m \geq n\) which is a contradiction. We conclude \((X \setminus O)^c = X \setminus O\), that is, \(O \in \mathcal{I}\).

Now let \(O \in \mathcal{I}\) and \(S \downarrow s \in O\), that is, \(S: D \to X\) is a net and \(S_n \to s \in O\) with respect to \(\mathcal{I}\). We conclude that there exists \(n \in D\) with \(S_m \in O\) for all \(m \geq n\). As \(s \in O\) was arbitrary and because of the definition of \(TX\) we get \(O \in TX\).

Lemma 1.5.8 (Associated Convergence Space) Let \(X\) be a nonempty set and \(\mathcal{C}\) a convergence class for \(X\) in net notation. Let \(\mathcal{I}\) be the induced topology of \(\mathcal{C}\). Let \((X, \downarrow_0)\) be the induced convergence space (see Definition 1.4.2) with respect to the topology \(\mathcal{I}\) and let \(S \downarrow_0 s\) iff \((S, s) \in \mathcal{C}\). Then we get \(\downarrow_0 = \downarrow\).

Proof. Applying Definition 1.4.2 and using Theorem 1.5.2 we conclude for each net \(S\) in \(X\) and \(s \in X\)

\[ S \downarrow_0 s \iff S_n \to s \text{ with respect to } \mathcal{I} \iff S \downarrow s. \]

Lemma 1.5.9 Let \(X\) be a nonempty set and \((X, \mathcal{I})\) a topological space. Let \((X, \downarrow_0)\) be the induced convergence space in net notation with respect to the topology \(\mathcal{I}\). We define \((S, s) \in \mathcal{C}\) iff \(S \downarrow_0 s\). Then \(\mathcal{C}\) is a convergence class in net notation.

Proof. As \((X, \downarrow_0)\) is a convergence space in net notation condition (a) and (b) in the definition of convergence class in net notation are already satisfied. We have to prove conditions (c) and (d). We have

\[ (S, s) \in \mathcal{C} \iff S \downarrow_0 s \iff S_n \to s \text{ with respect to } \mathcal{I}. \]

First we prove condition (c). Let \(S: D \to X\) be a net and let \((S, s) \notin \mathcal{C}\), that is, \((S_n)\) does not converge to \(s\) with respect to \(\mathcal{I}\). Then there exists \(U \in N(s)\) and for all \(n \in D\) there exists \(m_n \geq n\) with \(S_{m_n} \in X \setminus U\). Let \(D' = \{m_n \mid n \in D\} \subseteq D\) and \(T: D' \to X\) with

\[ T_n = S_n \text{ for all } n \in D'. \]

\(T\) is a subnet of \(S\) because \(D'\) is cofinal in \(D\). We have that \(T\) and every subnet \(R\) of \(T\) does not converge to \(s\) with respect to \(\mathcal{I}\) as \(T_n \in X \setminus U\) for all \(n \in D'\) and \(s \in U \in N(s)\). So \((T, s) \notin \mathcal{C}\) and \((R, s) \notin \mathcal{C}\) for each subnet \(R\) of \(T\).

Condition (d) follows immediately because we defined \((S, s) \in \mathcal{C}\) iff \(S_n \to s\) with respect to \(\mathcal{I}\). We have only to apply Theorem 1.5.1.
Remark 1.5.10 Lemma 1.5.8 and Lemma 1.5.9 imply that for every convergence class \( \mathcal{C} \) for \( X \) in net notation there exists a topology \( \mathcal{T} \) on \( X \) such that the induced convergence space is the convergence class \( \mathcal{C} \) and that every topology \( \mathcal{T} \) on \( X \) induces a topological convergence space in net notation which is a convergence class in net notation, that is, the class of topological convergence spaces in net notation is exactly the class of all convergence classes in net notation.

1.5.4 Properties of Convergence Classes in Filter Notation

Lemma 1.5.11 Let \( X \) be a nonempty set and \( \mathcal{C} \) a convergence class for \( X \) in filter notation. Then the induced topology \( \mathcal{T} \) of the convergence class coincides with \( TX \), that is, \( TX = \mathcal{T} \).

Proof. Let \( O \subseteq TX \) be an open set of the underlying convergence space in filter notation. We show that \( X \setminus O \) is closed with respect to \( \mathcal{T} \). Assume that there exists \( s \in (X \setminus O)^c \cap O \). Then there exists a filter \( \mathcal{F} \) on \( X \) with \( X \setminus O \in \mathcal{F} \) and \( \mathcal{F} \downarrow s \). As \( O \subseteq TX \) we conclude \( O \in \mathcal{F} \) which contradicts \( X \setminus O \in \mathcal{F} \). So we get \( X \setminus O = (X \setminus O)^c \), that is, \( O \) is open with respect to \( \mathcal{T} \).

Now let \( O \in \mathcal{F}, O \neq \emptyset \). Let \( \mathcal{A} \) be a filter on \( X \) with \( \mathcal{A} \downarrow x \in O \). We assume that \( O \notin \mathcal{A} \). Then for all \( B \subseteq O \) we have \( B \notin \mathcal{A} \) and conclude that for all \( B \in \mathcal{A} \) we have \( B \cap (X \setminus O) \neq \emptyset \). We define \( \mathcal{B} = \{ B \cap (X \setminus O) \mid B \in \mathcal{A} \} \). Then \( \mathcal{B} \) is a base of a filter \( \mathcal{G} \supseteq \mathcal{A} \). Therefore we obtain \( \mathcal{G} \downarrow x \in O \) and \( X \setminus O \in \mathcal{G} \). That means \( x \in (X \setminus O)^c = X \setminus O \) as \( O \in \mathcal{F} \) and we get a contradiction. So \( O \in \mathcal{A} \) and as \( x \in O \) was arbitrary we conclude \( O \in TX \). ■

Lemma 1.5.12 (Associated Convergence Space) Let \( X \) be a nonempty set and \( \mathcal{C} \) a convergence class for \( X \) in filter notation. Let \( \mathcal{T} \) be the induced topology of \( \mathcal{C} \). Let \( (X, \downarrow_{C}) \) be the induced convergence space (see Definition 1.4.4) with respect to the topology \( \mathcal{T} \) and let \( \mathcal{F} \downarrow_{C} x \) iff \( (\mathcal{F}, x) \in \mathcal{C} \). Then we get \( \downarrow_{O} = \downarrow_{C} \).

Proof. Applying Definition 1.4.4 and using Theorem 1.5.5 we conclude for each filter \( \mathcal{F} \) on \( X \) and \( x \in X \)

\[ \mathcal{F} \downarrow_{C} x \iff \mathcal{F} \rightarrow x \] with respect to \( \mathcal{T} \iff \mathcal{F} \downarrow_{C} x. \]

Corollary 1.5.13 Let \( X \) be a nonempty set. Every convergence class \( \mathcal{C} \) for \( X \) in filter notation is a pointed convergence space in filter notation.

Proof. Let \( x \in X \). We have \( \bigcap \mathcal{F}_x = \bigcap \{ A \mid A \downarrow_{C} x \} = \bigcap \{ A \mid A \downarrow_{O} x \} \) and the definition of an induced convergence space in filter notation (Definition 1.4.4) yields \( \bigcap \mathcal{F}_x = N(x) \in \mathcal{F}_x \). ■

Lemma 1.5.14 Let \( X \) be a nonempty set and \( (X, \mathcal{T}) \) a topological space. Let \( (X, \downarrow_{C}) \) be the induced convergence space in filter notation with respect to the topology \( \mathcal{T} \). We define \( (\mathcal{F}, x) \in \mathcal{C} \) iff \( \mathcal{F} \downarrow_{O} x \). Then \( \mathcal{C} \) is a convergence class in filter notation.

Proof. As \( (X, \downarrow_{C}) \) is a convergence space in filter notation condition (a) and (b) in the definition of convergence class in filter notation are already satisfied. We have to prove conditions (c) and (d). We have

\[ (\mathcal{F}, x) \in \mathcal{C} \iff \mathcal{F} \downarrow_{O} x \iff \mathcal{F} \supseteq N(x) \iff \mathcal{F} \rightarrow x \] with respect to \( \mathcal{T} \).
1.5. CONVERGENCE CLASSES

First we prove condition (c). Let \((\mathcal{F}, s) \notin \mathcal{C}\). There exists \(U \in N(s)\) with \(U \notin \mathcal{F}\). As \(\mathcal{F}\) is a filter for all \(B \subseteq U\) we have \(B \notin \mathcal{F}\) and we can conclude that for all \(F \in \mathcal{F}\) we have \(F \cap (X \setminus U) \neq \emptyset\). Let \(\mathcal{B} = \{F \cap (X \setminus U) \mid F \in \mathcal{F}\}\). Then \(\mathcal{B}\) is a base of a filter \(\mathcal{F}' \supseteq \mathcal{F}\). Now let \(\mathcal{G} \supseteq \mathcal{F}'\) be a filter. We assume that \(U \notin \mathcal{G}\). Because of \(\mathcal{B} \subseteq \mathcal{G}\) there exists \(F \in \mathcal{F}\) with \(F \cap (X \setminus U) \in \mathcal{B} \subseteq \mathcal{F}' \subseteq \mathcal{G}\) and \(\emptyset = U \cap F \cap (X \setminus U) \in \mathcal{G}\) which is a contradiction. So \(U \notin \mathcal{G}\) and for all filters \(\mathcal{G} \supseteq \mathcal{F}'\) we get \((\mathcal{G}, s) \notin \mathcal{C}\), that is, condition (c) is satisfied.

Condition (d) follows immediately because we defined \((\mathcal{F}, s) \in \mathcal{C}\) iff \(\mathcal{F} \rightarrow s\) with respect to \(\mathcal{F}\). We have only to apply Theorem 1.5.4. \(\blacksquare\)

**Remark 1.5.15** Lemma 1.5.12 and Lemma 1.5.14 imply that for every convergence class \(\mathcal{C}\) for \(X\) in filter notation there exists a topology \(\mathcal{T}\) on \(X\) such that the induced convergence space is the convergence class \(\mathcal{C}\) and that every topology \(\mathcal{T}\) on \(X\) induces a topological convergence space in filter notation which is a convergence class in filter notation, that is, the class of topological convergence spaces in filter notation is exactly the class of all convergence classes in filter notation.

1.5.5 Constructively Changing the Notation of a Convergence Class

In the following we will denote the closure operator of a convergence class in net notation by \(\mathcal{C}_1\) and the closure operator of a convergence class in filter notation by \(\mathcal{C}_2\). Let \(X\) be a nonempty set and \(\mathcal{C}\) be a convergence class on \(X\) in net notation. Because of Theorem 1.5.2 \(\mathcal{C}_1\) defines a topology \(\mathcal{T}\) on \(X\) with the property \((\mathcal{S}, s) \in \mathcal{C}\) iff \(S_n \rightarrow s\) with respect to \(\mathcal{T}\). As we have just seen in Lemma 1.5.14 \(\mathcal{T}\) defines a convergence class \(\mathcal{C}'\) on \(X\) in filter notation with the property

\[
(\mathcal{F}, s) \in \mathcal{C}' \iff \mathcal{F} \uparrow_0 s \iff \mathcal{F} \supseteq N(s) \iff \mathcal{F} \rightarrow s \text{ with respect to } \mathcal{T}.
\]

So the induced topologies of \(\mathcal{C}\) and \(\mathcal{C}'\) are equal, that is, we have \(\mathcal{A}^{\mathcal{C}_1} = \mathcal{A}^{\mathcal{C}_2}\) for all \(A \subseteq X\). In other words we have for all \(A \subseteq X\)

\[
\mathcal{A}^{\mathcal{C}_1} = \{s \in X \mid \exists \text{net } S \text{ in } A: (S, s) \in \mathcal{C}\}
\]

\[
= \{s \in X \mid \exists \text{filter } \mathcal{F}: A \in \mathcal{F}, (\mathcal{F}, s) \in \mathcal{C}'\} = \mathcal{A}^{\mathcal{C}_2}.
\]

We can construct \(\mathcal{C}'\) from \(\mathcal{C}\) by

**Lemma 1.5.16** Let \(X, \mathcal{C}\) and \(\mathcal{C}'\) as defined above. Then we have

\[
(\mathcal{F}, s) \in \mathcal{C}' \iff \exists (S, s) \in \mathcal{C}, S: D \rightarrow X \text{ net:}
\]

\[\mathcal{B} = \{B_n \mid n \in D\}\] base of \(\mathcal{F}\) with \(B_n = \{S_m \mid m \geq n\}\).

**Proof.** "\(\Leftarrow\)" Let \((S, s) \in \mathcal{C}\) and \(O \in N(s)\) be open. Then there exists \(n \in D\) with \(S_m \in O\) for all \(m \geq n\) and we have \(B_n \subseteq O\). As \(\mathcal{F}\) is a filter we get \(O \in \mathcal{F}\) and therefore we have \(N(s) \subseteq \mathcal{F}\). So we can conclude \((\mathcal{F}, s) \in \mathcal{C}'\) by the definition of \(\mathcal{C}'\).

"\(\Rightarrow\)" Let \((\mathcal{F}, s) \in \mathcal{C}'\). By definition \(\mathcal{F} \rightarrow s\) with respect to \(\mathcal{F}\). Let \(S: D_\mathcal{F} \rightarrow X\) be the net based on \(\mathcal{F}\) (see Definition 1.3.6). From Theorem 1.3.4 (a) it follows \(S_n \rightarrow s\) with respect to \(\mathcal{T}\). We obtain \((S, s) \in \mathcal{C}\) by means of the definition of \(\mathcal{C}\). The construction of \(S\) yields \(B_{(x,F)} = \{S_{(x',F')} \mid (x',F') \geq (x,F)\} = F\) for all \(x \in F \in \mathcal{F}\). Particularly we have \(\mathcal{B} = \{B_{(x,F)} \mid (x,F) \in D_\mathcal{F}\}\). \(\mathcal{F}\) is a base of \(\mathcal{F}\). \(\blacksquare\)
Next we start with a convergence class \( \mathcal{C}' \) for \( X \) in filter notation. Let \( \mathcal{T} \) be the induced topology such that we have

\[
(\mathcal{F}, s) \in \mathcal{C}' \iff \mathcal{F} \rightarrow s \quad \text{with respect to } \mathcal{T} \iff \mathcal{F} \supseteq \mathcal{N}(s).
\]

As we have seen in Lemma 1.5.9 \( \mathcal{T} \) defines a convergence class \( \mathcal{C} \) for \( X \) in net notation and by means of Lemma 1.5.7 the induced topologies of \( \mathcal{C} \) and \( \mathcal{C}' \) coincide, that is, are equal to \( \mathcal{T} \). Furthermore \( \mathcal{C} \) has the property

\[
(S, s) \in \mathcal{C} \iff \exists \text{net } S \text{ in } X : S_n \rightarrow s \quad \text{with respect to } \mathcal{T}.
\]

Once again Lemma 1.5.16 holds. Additionally we show

**Lemma 1.5.17** Let \( X, \mathcal{C}, \mathcal{C}' \) and \( \mathcal{T} \) be defined as above. Then we have

\[
(S, s) \in \mathcal{C} \iff \exists (\mathcal{F}, s) \in \mathcal{C}' : D_{\mathcal{F}} = \{(x, F) \mid x \in F \in \mathcal{F}\}, T : D_{\mathcal{F}} \rightarrow X \text{ with } T(x, F) = x \text{ and } S \text{ is a subnet of } T.
\]

**Proof.** "\( \Rightarrow \)" Let \( (S, s) \in \mathcal{C} \), that is, let \( S : D \rightarrow X \) be a net with \( S_n \rightarrow s \) with respect to \( \mathcal{T} \). Let \( B_n = \{S_m \mid m \geq n\} \) and \( \mathcal{B} = \{B_n \mid n \in D\} \) be the base of the filter \( \mathcal{F} \) generated by the net \( S \). Using Theorem 1.3.4 (b) we get \( \mathcal{F} \rightarrow s \) with respect to \( \mathcal{T} \). We conclude \( (\mathcal{F}, s) \in \mathcal{C}' \). Let \( D_{\mathcal{F}} \) and \( T \) as defined in our claim. Then \( T \) is a net as \( D_{\mathcal{F}} \) is directed by \((x_1, F_1) \leq (x_2, F_2)\) iff \( F_2 \subseteq F_1 \). Let \( \varphi : D \rightarrow D_{\mathcal{F}} \) be defined by \( \varphi(n) := (S_n, B_n) \) for all \( n \in D \). For all \( n, m \in D \) we have

\[
n \leq m \implies B_m \subseteq B_n \iff (S_n, B_n) \leq (S_m, B_m).
\]

So \( \varphi \) is a monotonic mapping. As for all \( x \in F \in \mathcal{F} \) there exists \( n \in D \) such that \( F \supseteq B_n \) we obtain \((x, F) \leq (S_n, B_n)\), that means, \( \text{Im } \varphi \) is cofinal in \( D_{\mathcal{F}} \). Therefore \( T \circ \varphi \) is a net with \( T(\varphi(n)) = T((S_n, B_n)) = S_n = S(n) \), that is, \( S = T \circ \varphi \). So \( S \) is a subnet of \( T \).

"\( \Leftarrow \)" Let \( (\mathcal{F}, s) \in \mathcal{C}' \). By definition of \( \mathcal{C}' \) we get \( \mathcal{F} \rightarrow s \) with respect to \( \mathcal{T} \). Because \( T \) is the net based on \( \mathcal{F} \) using Theorem 1.3.4 (a) we conclude \( T_n \rightarrow s \) with respect to \( \mathcal{T} \). As \( S \) is a subnet of \( T \) we obtain as well \( S_n \rightarrow s \) with respect to \( \mathcal{T} \), that is, \( (S, s) \in \mathcal{C} \) by definition of \( \mathcal{C} \).

\[\square\]
Chapter 2

Mathematical Methods for Logic Programming Semantics

In this chapter we discuss some frequently used mathematical methods for finding a declarative semantics of a logic program. These methods include set theoretic, generalized metric, domain theoretic and common topological aspects. The aim of each method is to find meaningful fixpoints of various operators, including the $T_P$ operator, which arise in logic programming semantics and which we describe in chapters 3 and 4.

Tarski’s theorem uses set theoretic methods to find the least fixpoint of an operator on a complete lattice (see below). These results are old and well-established. In the subsequent chapters we will often use this theorem.

Generalized metric methods for logic programming are rather new. M. Fitting used in [12] metric methods for finding the semantics of some logic programs. S. Priess-Crampe and P. Riebenboim found some useful fixpoint theorems for generalized ultrametrics (gums) (see [30, 31]) which are applicable in logic programming. They and M. Khamsi and D. Misane [27] also considered multivalued functions and their fixpoints but that is out of the scope of the diploma thesis. A. K. Seda and P. Hitzler extended the results of Priess-Crampe and Riebenboim and applied them to find the semantics of some classes of logic programs (see e.g. [20–22]).

Domain theory is a well-established science which is used to define the semantics of programming languages (not only LPs). A rather new book on that topic is [44]. A. K. Seda and P. Hitzler used domain theory and recast a Scott-Ershov domain into a generalized ultrametric (gum) [6, 20, 40]. The result was applicable to find the semantics of logic programs.

A. K. Seda also defined some useful topologies to investigate the topological continuity of the immediate consequence operator (see next chapter) of some classes of logic programs. One gets a simple characterisation of the nets which converge with respect to these topologies to a point (see [38]).

2.1 Lattice Theory

At first we need some definitions

**Definition 2.1.1** Let $L$ be a partially ordered set and $X \subseteq L$. We denote by $\inf(X)$ the greatest lower bound and by $\sup(X)$ the least upper bound of $X$ in $L$. We say $X \subseteq L$ is
directed if for all \(a, b \in L\) there exists \(c \in L\) with \(a \leq c\) and \(b \leq c\). Furthermore \(L\) is a complete lattice if \(\inf(X)\) and \(\sup(X)\) exist for all \(X \subseteq L\). We define \(\top = \sup(L)\) to be the top element of \(L\) (if it exists) and \(\bot = \inf(L)\) to be the bottom element of \(L\) (if it exists). We have that \(L\) is a complete semilattice if \(\inf(X)\) exists for all \(X \subseteq L\) and \(\sup(X)\) exists for all directed subsets \(X \subseteq L\). Let \(T : L \rightarrow L\) be a mapping. We say \(T\) is monotonic if for all \(x, y \in L, x \leq y\) we have \(T(x) \leq T(y)\) and \(T\) is (Scott) continuous if for each directed set \(X \subseteq L\) we have \(T(\sup(X)) = \sup(T(X))\). The least fixpoint of \(T\) (if it exists) is denoted \(\text{lfp}(T)\) and the greatest fixpoint of \(T\) (if it exists) is denoted \(\text{gfp}(T)\).

The following result can be found in [29].

**Theorem 2.1.1 (Tarski’s Theorem)** Let \(L\) be a complete lattice, \(T : L \rightarrow L\) a monotonic mapping. Then \(\text{lfp}(T)\) and \(\text{gfp}(T)\) exist and we have \(\text{lfp}(T) = \inf\{a \in L \mid T(a) = a\}\) and \(\text{gfp}(T) = \sup\{a \in L \mid T(a) = a\}\).

**Definition 2.1.2 (Ordinal Powers)** Let \(L\) be a complete lattice and \(T : L \rightarrow L\) be a mapping. We define

\[
T \uparrow 0 = \bot, \\
T \uparrow (\alpha + 1) = T(T \uparrow \alpha), \\
T \uparrow \alpha = \sup\{T \uparrow \beta \mid \beta < \alpha\}, \text{Lim}(\alpha), \\
T \downarrow 0 = \top, \\
T \downarrow (\alpha + 1) = T(T \downarrow \alpha), \\
T \downarrow \alpha = \inf\{T \downarrow \beta \mid \beta < \alpha\}, \text{Lim}(\alpha).
\]

There exists the following relationship between ordinal powers of \(T\) and \(\text{lfp}(T)\) and \(\text{gfp}(T)\).

**Proposition 2.1.2** Let \(L\) be a complete lattice and \(T : L \rightarrow L\) be a mapping. If \(T\) is monotonic, then for each ordinal \(\alpha \in \text{On}\) we have \(T \uparrow \alpha \leq \text{lfp}(T)\) and \(T \downarrow \alpha \geq \text{gfp}(T)\). Additionally there exist ordinals \(\beta_1, \beta_2 \in \text{On}\) with \(T \uparrow \beta_1 = \text{lfp}(T)\) and \(T \downarrow \beta_2 = \text{gfp}(T)\). If \(T\) is a (Scott) continuous mapping we have \(\text{lfp}(T) = T \uparrow \omega\).

**Proposition 2.1.3** Let \(L\) be a complete semilattice and \(T : L \rightarrow L\) be a monotonic mapping. Then \(\text{lfp}(T)\) exists and there exists an ordinal \(\alpha \in \text{On}\) with \(\text{lfp}(T) = T \uparrow \alpha\).

**Definition 2.1.3** In the situation of Proposition 2.1.2 and Proposition 2.1.3 the least ordinal \(\alpha\) such that \(\text{lfp}(T) = T \uparrow \alpha\) is called the closure ordinal of \(T\).

### 2.2 Generalized Metric Methods

In many situations the strong premises in the definition of a metric are not satisfied but a weaker structure is still given. Therefore at first we define some generalizations of metric spaces.

**Definition 2.2.1 (Generalized Metric Structures)** Let \(X\) be a nonempty set, \(\Gamma\) a partially ordered set with least element \(0\) and \(d : X \times X \rightarrow \Gamma\) a mapping. We call \(d\) a distance function on \(X\). Consider the following axioms.
2.2. GENERALIZED METRIC METHODS

(a) For all \( x \in X \) we have \( d(x, x) = 0 \) (zero self-distance).

(b) For all \( x, y \in X \) we have \( d(x, y) = d(y, x) = 0 \) implies \( x = y \).

(c) For all \( x, y \in X \) we have \( d(x, y) = d(y, x) \) (symmetry).

(d) There exists an Abelian semigroup structure \((\Gamma, +)\) such that \( 0 + u = u \) for all \( u \in X \) and for all \( x, y, z \in X \) we have \( d(x, y) \leq d(x, z) + d(z, y) \) (triangle inequality).

(e) For all \( x, y, z \in X \) we have \( d(x, z), d(z, y) \leq \gamma \in \Gamma \) implies \( d(x, y) \leq \gamma \) (strong triangle inequality).

(f) \( \Gamma = \mathbb{R}^+_0 \).

We characterize generalizations of a metric by the above axioms. At first let \( \Gamma = \mathbb{R}^+_0 \), that is, let (f) be true.

<table>
<thead>
<tr>
<th>Name</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>metric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>quasimetric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pseudometric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>quasi-pseudo metric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>quasi ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pseudo ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>quasi-pseudo ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Now let \( \Gamma \) be arbitrary. We distinguish the following generalizations of metrics.

<table>
<thead>
<tr>
<th>Name</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
</tr>
</thead>
<tbody>
<tr>
<td>generalized ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dislocated generalized ultrametric</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For each generalization of metric in the above tables we use \(*\) to indicate axioms which must be satisfied for that structure and \(\circ\) for axioms which are implications of the \(*\) axioms of the same structure.

If \( \Gamma \) is arbitrary and \( d \) a \((d-)gum\), we say \((X, d, \Gamma)\) is a \((d-)ultrametric space\). In all other cases we say \((X, d)\) is the whatever metric space, e.g. if \(d\) is a quasimetric we say \((X, d)\) is a quasimetric space.

In the following we are mainly investigating metrics, quasimetrics, ultrametrics, gums and dislocated gums \((d-gums)\). All other generalizations of a metric will not be used here.

We are interested in conditions on functions \( f : X \rightarrow X \) operating on a generalized version of a metric space such that \( f \) has a unique fixpoint. We also want to be able to get that fixpoint constructively, that is, with an iteration construction. At first we consider the metrical case.

**Theorem 2.2.1 (Banach Contraction Mapping Theorem)** Let \((X, d)\) be a complete metric space, \( 0 < c < 1 \) and \( f : X \rightarrow X \) be a function with \( d(f(x), f(y)) \leq c d(x, y) \) for all \( x, y \in X \) (that means \( f \) is a contraction mapping). Then \( f \) has a unique fixpoint and the sequence \((f^n(x))_{n\in\mathbb{N}}\) converges with respect to \((X, d)\) to the unique fixpoint. \( \square \)
A proof of this theorem can be found in most (numerical) analysis books. M. Fitting used Banach’s theorem in [12] to show that metric methods can be applied to find the semantics of logic programs. S. Priess-Crampe and P. Ribenboim showed some more general results for gums. We immediately prove an extended version for d-gums. It was published by A. K. Seda and P. Hitzler in [21] and will be used in the following chapter. We need some more definitions and two helpful lemmas. The first lemma can be found in [30, 31].

**Definition 2.2.2** Let $(X,d,\Gamma)$ be a d-ultrametric space, $0 \neq \gamma \in \Gamma$ and $x \in X$. Then $B_\gamma(x) = \{ y \in X \mid d(x,y) \leq \gamma \}$ is called a $\gamma$-ball in $X$ with center $x$ and radius $\gamma$. We say $X$ is spherically complete if we have for each chain $(\mathcal{C},\subseteq)$ of nonempty balls in $X$, that is, each set $\mathcal{C}$ of nonempty balls in $X$ which is linear ordered by $\subseteq$, that $\bigcap \mathcal{C} \neq \emptyset$. Let $f: X \rightarrow X$ be a function. We call $f$

(a) non-expanding if $d(f(x), f(y)) \leq d(x, y)$ for all $x,y \in X$.

(b) strictly contracting if $d(f(x), f(y)) < d(x, y)$ for all $x, y \in X$ with $x \neq y$. \hfill \Box

**Lemma 2.2.2** Let $(X,d,\Gamma)$ be a d-ultrametric space. Then for all $\alpha, \beta \in \Gamma$ and $x, y \in X$ we have

(i) If $\alpha \leq \beta$ and $B_\alpha(x) \cap B_\beta(y) \neq \emptyset$, then $B_\alpha(x) \subseteq B_\beta(y)$.

(ii) If $x \in B_\alpha(x) \subset B_\beta(y)$, then $\beta \leq \alpha$.

(iii) If $B_\alpha(x) \cap B_\beta(y) \neq \emptyset$, then $B_\alpha(x) = B_\alpha(y)$.

(iv) $B_{d(x,y)}(x) = B_{d(x,y)}(y)$.

**Proof.** (i) Let $a \in B_\alpha(x)$ and $b \in B_\beta(x)$. We get $d(a,x) \leq \alpha$ and $d(b,x) \leq \alpha$ as well as $d(b,y) \leq \beta$. Using the strong triangle inequality we conclude $d(a,b) \leq \alpha \leq \beta$ and finally $d(a,y) \leq \beta$. So we obtain $a \in B_\beta(y)$.

(ii) Assume $\beta \leq \alpha$. We conclude $B_\alpha(x) \subseteq B_\beta(y) = B_\beta(x) \subseteq B_\alpha(x)$ by means of (i) but that is a contradiction.

(iii) follows from (i) by symmetry.

(iv) We have $y \in B_{d(x,y)}(x)$ and $d(x,y), d(y,x) \leq d(x,y)$. The strong triangle inequality yields $d(y,x) \leq d(x,y)$ and we obtain $y \in B_{d(x,y)}(y)$. Now (iv) follows from (iii). \hfill \Box

**Lemma 2.2.3** Let $(X,d,\Gamma)$ be a spherically complete d-ultrametric space and $f : X \rightarrow X$ be a non-expanding function without fixpoint. Let $\mathcal{B} = \{ B_{d(x,f(x))}(x) \mid x \in X \}$. Then for each chain $\mathcal{C} \subseteq \mathcal{B}$ and $z \in \bigcap \mathcal{C}$ we have $B_{d(z,f(z))}(z) \subseteq \bigcap \mathcal{C}$.

**Proof.** As $f$ has no fixpoint we get from Definition 2.2.1 (b) $d(x, f(x)) \neq 0$ for all $x \in X$. So $\mathcal{B}$ is well-defined. As $f(x) \in B_{d(x,f(x))}(x)$ for all $x \in X$ the balls in $\mathcal{B}$ are nonempty and as $X$ is spherically complete $z \in \bigcap \mathcal{C}$ exists. Let $B_{d(x,f(x))}(x) \in \mathcal{C}$. We obtain $z \in B_{d(x,f(x))}(x) = B_{d(x,f(x))}(f(x))$ by Lemma 2.2.2 (iv). We conclude $d(z, f(z)) \leq d(x, f(x))$ and because $f$ is non-expanding we get $d(f(z), f(x)) \leq d(z, x) \leq d(x, f(x))$. Using the strong triangle inequality we have $d(z, f(z)) \leq d(x, f(x))$. Because $f(z) \in B_{d(z,f(z))}(z) \cap B_{d(x,f(x))}(f(x))$ we can use Lemma 2.2.2 (i) and obtain $B_{d(z,f(z))}(z) \subseteq B_{d(x,f(x))}(x)$. So we are finished. \hfill \Box

The following theorem is due to A. K. Seda and P. Hitzler. We prove it for completeness. The proof can also be found in [21].
2.2. GENERALIZED METRIC METHODS

Theorem 2.2.4 Let \((X,d,\Gamma)\) be a spherically complete \(d\)-ultrametric space and \(f: X \to X\) be strictly contracting. Then \(f\) has a unique fixpoint.

**Proof.** We assume that \(f\) has no fixpoint and apply Lemma 2.2.3. Let \(B\) be as defined in Lemma 2.2.3, \(C\) be a maximal chain in \(B\) and \(z \in \bigcap C\). Then \(B_{d(z,f(z))}\) is the smallest ball in \(C\). As \(f\) is strictly contracting we have \(d(f(z), f^2(z)) < d(z, f(z))\) and therefore \(z \notin B_{d(f(z), f^2(z))}(f(z)) \subseteq B_{d(z,f(z))}(f(z)) = B_{d(z,f(z))}(z)\) by Lemma 2.2.2 which contradicts the maximality of the chain \(C\).

We show that \(f\) has only one fixpoint. Assume \(x, y \in X\) are fixpoints of \(f\). Then we get \(d(x,y) = d(f(x), f(y)) < d(x, y)\) because \(f\) is strictly contracting and we get a contradiction. 

\[\forall \beta, \gamma \leq \gamma: d(f^\beta(x), f^\gamma(x)) \leq 2^{-\min\{\beta, \gamma\}}.\] (2.1)

**Proof.** We prove our claim by transfinite induction over \(\gamma' \leq \gamma\). Let \((f^\gamma(x))_{\gamma' \leq \gamma}\) be the beginning of our transfinite sequence which already satisfies the above properties. We show that \((f^\gamma(x))_{\gamma' \leq \gamma'}\) also satisfies the above properties.

(i) Let \(\gamma' = 0\). As \(d(x, x) \leq 2^{-\alpha}\) for all \(x \in X\) we are finished.

(ii) Let \(\gamma' = \alpha + 1 \leq \gamma\). Equation (2.1) is true for \(\beta, \gamma' < \gamma\). So let \(\beta = \gamma' = \alpha + 1\). If \(\beta\) is a successor ordinal, we get \(d(f^\beta(x), f^\gamma(x)) < d(f^\alpha(x), f^{\beta-1}(x)) \leq 2^{-\beta-1}\) and conclude \(d(f^\beta(x), f^\gamma(x)) \leq 2^{-\min\{\beta, \gamma'\}} = 2^{-\beta}\). If we have \(\lim(\beta')\), we get \(\beta' < \beta = \gamma'\). Let \(\delta + 1 < \beta\) arbitrary. Because of our induction hypothesis and the first part of (ii) we can conclude \(d(f^\beta(x), f^{\delta+1}(x)) \leq 2^{-\delta+1}\) and \(d(f^{\delta+1}(x), f^\gamma(x)) \leq 2^{-\delta+1}\). Therefore we obtain \(d(f^\beta(x), f^\gamma(x)) \leq 2^{-\delta+1}\) for all \(\delta + 1 < \beta\), that is, \(d(f^\beta(x), f^\gamma(x)) \leq 2^{-\beta}\).

(iii) We prove the case \(\lim(\gamma')\). Let \(\alpha \leq \alpha' < \gamma\) and \(y \in B_{2^{-\alpha'}}(f^\gamma(x))\). We get \(d(y, f^\alpha(x)) \leq 2^{\alpha'} \leq 2^{-\alpha}\) and \(d(f^\alpha(x), f^\gamma(x)) \leq 2^{-\alpha}\). Using the strong triangle inequality we conclude \(d(y, f^\gamma(x)) \leq 2^{-\alpha}\), that is, \(y \in B_{2^{-\alpha}}(f^\gamma(x))\). So we proved \(B_{2^{-\alpha}}(f^\gamma(x)) \subseteq B_{2^{-\alpha}}(f^\gamma(x))\), that is, \((B_{2^{-\alpha}}(f^\gamma(x)))_{\alpha \leq \gamma}\) is a decreasing chain. As \(X\) is spherically complete there exists \(f^\gamma(x) \cap B_{2^{-\alpha}}(f^\gamma(x)) \neq \emptyset\) and we get \(d(f^\gamma(x), f^\gamma(x)) \leq 2^{-\alpha}\) for all \(\alpha < \gamma\). The strong triangle inequality yields \(d(f^\gamma(x), f^\gamma(x)) \leq 2^{-\gamma}\). So we proved (2.1).

(iv) We remark that for each \(\beta < \gamma\) the ball \(B_{2^{-\beta}}(f^\beta(x))\) is nonempty by means of (2.1). The proof in (iii) is also true for arbitrary \(\alpha \leq \alpha' < \gamma\). So \((B_{2^{-\alpha}}(f^\beta(x)))_{\beta \leq \gamma}\) is a decreasing chain of balls.

Theorem 2.2.5 shows that one can get the unique fixpoint of \(f\) by transfinite iteration of \(f\) starting from an arbitrary point \(x \in X\) and that one has reached the fixpoint in no more than \(\gamma\) steps. In detail we get \(d(f^{\gamma+1}(x), f^{\delta+1}(x)) < d(f^\gamma(x), f^\delta(x)) \leq 2^{-\delta}\) for all \(\delta + 1 \leq \gamma\), that is, \(d(f^{\gamma+1}(x), f^\gamma(x)) \leq 2^{-\gamma} = 0\) (if \(\lim(\gamma)\), we use the strong triangle inequality) which means \(f^{\gamma+1}(x) = f^\gamma(x)\).

Next we consider quasiisometric spaces. We need the following definitions
**Definition 2.2.3** Let \((X, d)\) be a quasimetric space. A sequence \((x_n)\) in \(X\) is called forward Cauchy if for all \(\varepsilon > 0\) there exists \(k \in \mathbb{N}\) such that \(d(x_n, x_m) \leq \varepsilon\) for all \(m \geq n \geq k\). A point \(x \in X\) is a limit of the forward Cauchy sequence \((x_n)\) if for all \(y \in X\) we have 
\[
d(x, y) = \lim_{n \to \infty} d(x_n, y).
\]
We write \(x = \lim_{n \to \infty} x_n\) or simply \(x = \lim x_n\). Limits in that meaning are sometimes also called metric limits. The space \(X\) is complete if every forward Cauchy sequence in \(X\) has a limit. Let \((Y, d')\) be another quasimetric space and \(f : X \to Y\) be a function. We say \(f\) is Continuous if for each forward Cauchy sequence \((x_n)\) in \(X\) and \(x \in X\) we have \(\lim x_n = x\) implies \(\lim f(x_n) = f(x)\).

In the literature not only forward Cauchy sequences are used in connection with quasimetric spaces. One gets backward- or bi-Cauchy sequences by replacing the condition on \(k\) through \(n \geq m \geq k\) or \(n, m \geq k\) respectively. If \(d\) is a quasi ultrametric on \(X\) we get that \((x_n)\) is a forward Cauchy sequence iff for all \(\varepsilon > 0\) there exists \(k \in \mathbb{N}\) with \(d(x_n, x_{n+1}) < \varepsilon\) for all \(n \geq k\). This is a direct consequence of the strong triangle inequality property of \(d\). We will always use \(\text{lim}\) (instead of \(\lim\)) when we mean metric limits. As well we say a function \(f\) is Continuous (instead of continuous) when we mean continuity in the metric sense. We remark that the limit of a forward Cauchy sequence \((x_n)\) in \(X\) is unique. Suppose \(x, y \in X\) are metric limits of \(f : X \to X\). By definition we get 
\[
d(x, z) = \lim d(x_n, z) = d(y, z)\quad\text{for all } z \in X.
\]
Because \(d(x, x) = d(y, y) = 0\) we conclude \(d(y, x) = d(x, y) = 0\), that is, \(x = y\) by axiom (b) in Definition 2.2.1.

In chapter 4 we will need the following theorem. It is due to Rutten (see [36]). For a quasimetric space \((X, d)\) we define a partial order \(\leq_X\) on \(X\) by \(x \leq_X y\) iff \(d(x, y) = 0\) for all \(x, y \in X\).

**Theorem 2.2.6 (Rutten’s Theorem)** Let \((X, d)\) be a complete quasi ultrametric space and \(f : X \to X\) a non-expanding and Continuous function such that there exists \(x \in X\) with 
\[
d(x, f(x)) = 0.
\]
Then \(f\) has a fixedpoint which is the least fixedpoint above \(x\) with respect to the partial order \(\leq_X\).

**Proof.** The sequence \((f^n(x))_{n \in \mathbb{N}}\) is Cauchy because \(f\) is non-expanding (by induction 
\[
d(f^n(x), f^{n+1}(x)) \leq d(x, f(x)) = 0.
\]
As \(X\) is complete it has a limit \(y \in X\) and as \(f\) is Continuous we get 
\[
f(y) = \lim f(f^n(x)) = y.
\]
Because \(d\) satisfies the strong triangle inequality we obtain 
\[
d(x, f^n(x)) = 0\quad\text{and as } y = \lim f^n(x) \text{ we get } d(y, y) = 0 = \lim d(f^n(x), y).
\]
So let \(\varepsilon > 0\). Then there exists \(m \in \mathbb{N}\) such that 
\[
d(f^n(x), y) < \varepsilon\quad\text{for all } n \geq m.
\]
Using the strong triangle inequality we conclude 
\[
d(x, y) \leq \max\{d(x, f^n(x)), d(f^n(x), y)\} \leq \varepsilon\quad\text{for all } n \geq m.
\]
As \(\varepsilon > 0\) was arbitrary we have 
\[
d(x, y) = 0\quad\text{that is, } x \leq_X y.
\]
Now let \(y' \in X\) be another fixedpoint of \(f\) with \(x \leq_X y'\), that is, 
\[
d(x, y') = 0.
\]
We conclude 
\[
d(f^n(x), y') = d(f^n(x), f^n(y')) \leq d(x, y') = 0\quad\text{as well as } d(y, y') = \lim d(f^n(x), y') = 0\quad\text{by means of } y = \lim f^n(x).
\]
So we have \(y \leq_X y'\), that is, \(y\) is the least fixedpoint of \(f\) above \(x\). ■

### 2.3 Domain Theory

We show how a Scott-Enshov domain can be cast into a generalized ultrametric space and apply our result to the set of valuations. At first we need some definitions (see [6, 20, 44]).

**Definition 2.3.1** Let \((D, \sqsubseteq)\) be a partially ordered set (poset). We say \(D\) is directed if for all \(a, b \in D\) there exists \(c \in D\) with \(a \leq c\) and \(b \leq c\) (compare with Definition 1.3.1). We define \((D, \sqsubseteq)\) to be a complete partial order (cpo) if it has a bottom element \(\bot \in D\) and for
each directed set \( M \subseteq D \) the least upper bound \( \sqcup M \) of \( M \) in \( D \) exists. An element \( x \in D \) is called compact if for each directed set \( M \subseteq D \) with \( x \sqsubseteq \sqcup M \) there exists \( a \in M \) with \( x \sqsubseteq a \). The set of compact elements of \( D \) is denoted by \( D_C \).

A subset \( A \subseteq D \) is consistent if there exists an upper bound of \( A \) in \( D \), that is, there exists \( x \in D \) with \( a \sqsubseteq x \) for all \( a \in A \).

**Definition 2.3.2 (Scott-Ershov Domain)** A poset \( (D, \sqsubseteq) \) is called a Scott-Ershov domain or simply domain if \( (D, \sqsubseteq) \) satisfies

1. \( (D, \sqsubseteq) \) is a cpo.
2. For each \( x \in D \), the set \( \text{approx}(x) = \{ a \in D_C \mid a \sqsubseteq x \} \) is directed and \( x = \sqcup \text{approx}(x) \) (algebraicity of \( D \)).
3. For each consistent \( A \subseteq D \), the supremum, \( \sqcup A \) exists in \( D \) (consistent completeness of \( D \)).

**Definition 2.3.3** Let \( P \) be a normal logic program, \( B \) be a set of truth values and \( \leq \) a poset on \( B \) such that \( (B, \leq) \) is a complete lattice with bottom element \( \bot \). A valuation \( v \in B_P \) is finite if the set \( \{ A \in B_P \mid v(A) \neq \bot \} \) is finite. We define \( v_\bot \in B_P \) by \( v_\bot(A) = \bot \) for all \( A \in B_P \).

The proof of the following theorem can be found in [6, Theorem 3.4].

**Theorem 2.3.1** Let \( P, B \) and \( \leq \) as defined in Definition 2.3.3. Then \( (B_P, \leq) \) is a Scott-Ershov domain with bottom element \( v_\bot \) and the set of compact elements is identical with the set of finite valuations.

Now we want to cast a Scott-Ershov domain into a generalized ultrametric (gum). Let \( \gamma \in \text{On} \) be countable, that is, \( \gamma \) has cardinality \( 8_0 \) and let once again \( \Gamma = \{ 2^{-\alpha} \mid \alpha \leq \gamma \} \) be ordered by \( 2^{-\alpha} < 2^{-\beta} \) iff \( \beta < \alpha \) and denote \( 2^{-\gamma} \) by 0.

**Definition 2.3.4 (Rank Function)** Let \( (D, \sqsubseteq) \) be a domain with compact elements \( D_C \). A function \( r_C : D_C \rightarrow \Gamma \) is called a rank function. We define a distance function \( d_r : D \times D \rightarrow \Gamma \) by \( d_r(x, y) = \inf \{ 2^{-\alpha} \mid c \sqsubseteq x \text{ iff } c \sqsubseteq y \text{ for all } c \in D_C \text{ with } r_C(c) < \alpha \} \).

One can simply check that \( d_r \) in the above definition is an ultrametric on \( D \). One can also show that \( d_r \) is spherically complete but we omit the proof as we do not need it in the subsequent chapters. A proof can be found in [20, Lemma 4.6] and [40, Theorem 3.5].

### 2.4 Topologies for Logic Programming Semantics

In chapter 4 we will investigate the (topological) continuity or lack of (topological) continuity of some mappings used for logic programming semantics with respect to topologies defined in this section. These mappings operate on the set of all two- or three-valued interpretations of a normal logic program (NLP). So we have to define topologies on the set of all interpretations with respect to a program \( P \).

The benefit of doing that is not always clear in advance. Suppose we are considering an operator \( T : X \rightarrow X \) and can define a topology \( \mathcal{F} \) on \( X \). If \( T \) is continuous with respect to \( \mathcal{F} \) one can e.g. construct a net such that \( x_n \rightarrow x \in X \) with respect to \( \mathcal{F} \) and find \( T(x) \) by
iterating $T(x_n)$ which is perhaps much easier than trying to get $T(x)$ directly. If one can find additional properties of the topology $\mathcal{T}$ one may be able to express $T(x)$ in a short and precise way. The properties of the topology may also give some informations about the operator $T$. In practice one may be able to compute the solution of $T(x)$ in less time.

A. Batarekh and V. S. Subrahmanian investigated topologies in logic programming semantics in [2] and these were extended and simplified by A. K. Seda in [38]. Based on the topologies in [38] we define the following topologies for two-valued interpretations.

**Definition 2.4.1** Let $\mathbf{P}$ be a NLP. For each ground atom $A \in \mathit{IP}$ we define $\mathcal{S}(A) = \{ I \in \mathit{IP} \mid I \models A \} = \{ I \in \mathit{IP} \mid A \in I \}$ and $\mathcal{S}(\neg A) = \{ I \in \mathit{IP} \mid I \models \neg A \} = \{ I \in \mathit{IP} \mid A \notin I \}$. Let $\mathcal{S}^+ = \{ \mathcal{S}(A) \mid A \in \mathit{IP} \}$ and $\mathcal{S}^- = \{ \mathcal{S}(\neg A) \mid A \in \mathit{IP} \}$. The positive atomic topology $\mathcal{S}^+$ on $\mathit{IP}$ is the collection of all subsets of $\mathit{IP}$ obtained by taking $\mathcal{S}^+$ as subbasis of $\mathbf{P}$. The negative atomic topology $\mathcal{S}^-$ on $\mathit{IP}$ is the collection of all subsets of $\mathit{IP}$ obtained by taking $\mathcal{S}^+ \cup \mathcal{S}^-$ as subbasis of $\mathbf{P}$.

As the Herbrand base of each NLP $\mathbf{P}$ is countable we immediately get that the sets $\mathcal{S}^+$ and $\mathcal{S}^-$ as well as $\mathcal{S}$ are countable. Therefore $\mathcal{S}^+, \mathcal{S}^-$ and $\mathcal{S}$ have a countable basis i.e. the corresponding topological space is second countable (see [45, Definition 16.1]). We conclude that these topological spaces are also first countable i.e. have a countable neighbourhood basis at each point $I \in \mathit{IP}$. Therefore we can reduce convergence with respect to the topologies $Q, Q^+$ and $Q^-$ to convergence of sequences (see [45, Corollary 10.5 c]).

We have the following characterisations of convergence in the topologies $Q^+, Q^-$ and $Q$.

**Proposition 2.4.1** Let $(I_n)_{n \in \mathbb{N}}$ be a sequence in $\mathit{IP}$ and $I \in \mathit{IP}$. Then the following statements are satisfied

(a) $I_n \rightarrow I$ in $Q$ iff for all $A \in I$ there exists $m \in \mathbb{N}$ with $A \in I_n$ for all $n \geq m$ and for all $A \notin I$ there exists $m \in \mathbb{N}$ with $A \notin I_n$ for all $n \geq m$.

(b) $I_n \rightarrow I$ in $Q^+$ iff for all $A \in I$ there exists $m \in \mathbb{N}$ with $A \in I_n$ for all $n \geq m$.

(c) $I_n \rightarrow I$ in $Q^-$ iff for all $A \notin I$ there exists $m \in \mathbb{N}$ with $A \notin I_n$ for all $n \geq m$.

**Proof.** (a) The proof of statement (a) follows from the proofs of statements (b) and (c).

(b) Let $I_n \rightarrow I$ in $Q^+$ and $A \in I$. We have $I \in \mathcal{S}(A)$ by construction. As $\mathcal{S}(A)$ is open in $Q^+$ we conclude that there exists $m \in \mathbb{N}$ with $I_n \in \mathcal{S}(A)$ for all $n \geq m$, that is, $A \in I_n$ for all $n \geq m$. Now let $U$ be a neighbourhood of $I$. Without loss of the generality we can assume that $U$ is an element of the base of $Q^+$, that is, has the form $\mathcal{S}(A_1) \cap \ldots \cap \mathcal{S}(A_k)$. As $I \in U$ we conclude $A_1, \ldots, A_k \in I$. Using our premise there exists $m \in \mathbb{N}$ such that $A_1, \ldots, A_k \in I_n$ for all $n \geq m$. So we obtain $I_n \in \mathcal{S}(A_1) \cap \ldots \cap \mathcal{S}(A_k)$ for all $n \geq m$ and are finished.

(c) Let $I_n \rightarrow I$ in $Q^-$ and $A \notin I$. We have $I \notin \mathcal{S}(\neg A)$ by construction. As $\mathcal{S}(\neg A)$ is open in $Q^-$ we conclude that there exists $m \in \mathbb{N}$ with $I_n \in \mathcal{S}(\neg A)$ for all $n \geq m$, that is, $A \notin I_n$ for all $n \geq m$. Now let $U$ be a neighbourhood of $I$. Without loss of the generality we can assume that $U$ is an element of the base of $Q^-$, that is, has the form $\mathcal{S}(\neg A_1) \cap \ldots \cap \mathcal{S}(\neg A_k)$. As $I \in U$ we conclude $A_1, \ldots, A_k \notin I$. Using our premise there exists $m \in \mathbb{N}$ such that $A_1, \ldots, A_k \notin I_n$ for all $n \geq m$. So we obtain $I_n \in \mathcal{S}(\neg A_1) \cap \ldots \cap \mathcal{S}(\neg A_k)$ for all $n \geq m$ and are finished.

We use Definition 2.4.1 to construct topologies on the set of three-valued interpretations.
Definition 2.4.2 Let $\mathcal{P}$ be a NLP. On the set of three-valued interpretations $\mathcal{P} \times \mathcal{P}$ we define $Q_2$ to be the topology with basis $Q \times Q$, that is, $Q_2$ is the product topology. The topology $Q'_2$ on $\mathcal{P} \times \mathcal{P}$ has the basis $Q \times Q^-$.

As $(\mathcal{P}, Q)$ and $(\mathcal{P}, Q^-)$ are first countable spaces we immediately get that the spaces corresponding to the topologies $Q_2$ and $Q'_2$ are also first countable. We can reduce continuity in $Q_2$ and $Q'_2$ to continuity in $Q$ and $Q^-$ by the following characterization

Lemma 2.4.2 Let $(I_n)_{n \in \mathbb{N}}$ be a sequence in $\mathcal{P} \times \mathcal{P}$ and $I \in \mathcal{P} \times \mathcal{P}$. Then the following statements are satisfied

(a) $I_n \to I$ in $Q_2$ iff $I^+_n \to I^+$ in $Q$ and $I^-_n \to I^-_n$ in $Q$.

(b) $I_n \to I$ in $Q'_2$ iff $I^+_n \to I^+$ in $Q$ and $I^-_n \to I^-_n$ in $Q^-$.

Proof. The proof is trivial.

Corollary 2.4.3 Let $f : \mathcal{P} \times \mathcal{P} \to \mathcal{P} \times \mathcal{P}$ be a function. $f$ is continuous in $Q_2$ iff for all sequences $(I_n)_{n \in \mathbb{N}}$ in $\mathcal{P} \times \mathcal{P}$ and $I \in \mathcal{P} \times \mathcal{P}$ with $I^+_n \to I^+$ and $I^+_n \to I^-$ in $Q$ we have $f(I^+_n) \to f(I^+)$ and $f(I^-_n) \to f(I^-)$ in $Q$. □
Chapter 3

Extensions to the Classes of $\Phi^*$— and $\Phi$—accessible Programs

3.1 Some Classes of Logic Programs and their Semantics

First we need some definitions.

**Definition 3.1.1** Let $\mathbf{P}$ be a normal logic program. The (two-valued) immediate consequence operator $T_{\mathbf{P}} : I_{\mathbf{P}} \rightarrow I_{\mathbf{P}}$ is defined for each $I \in I_{\mathbf{P}}$ by

$$T_{\mathbf{P}}(I) = \{ A \in B_{\mathbf{P}} \mid \exists R \in \text{ground}(\mathbf{P}) : \text{head}(R) = A, \text{pos}(R) \subseteq I, \text{neg}(R) \cap I = \emptyset \}.$$  

An interpretation $I \in I_{\mathbf{P}}$ is a model of $\mathbf{P}$ if $T_{\mathbf{P}}(I) \subseteq I$. We say $I$ is supported if we have $T_{\mathbf{P}}(I) \supseteq I$.

Let $\gamma$ be a countable ordinal number. A level mapping for $\mathbf{P}$ is a mapping $l : B_{\mathbf{P}} \rightarrow \gamma$ and for each $A \in B_{\mathbf{P}}$ we say $l(A)$ is the level of $A$. We extend the domain of $l$ to the set of all literals by $l(L) = l(A)$ iff $L = A$ or $L = \neg A$. We call $l$ an $\omega$-level mapping if $\gamma = \omega$. We define $L_\alpha = \{ A \in B_{\mathbf{P}} \mid l(A) < \alpha \}$, that is, $L_\alpha$ is the set of all ground atoms with level less than $\alpha$.

So a ground atom $A \in B_{\mathbf{P}}$ is an element of $T_{\mathbf{P}}(I)$ for $I \in I_{\mathbf{P}}$ iff there exists a ground instance $A \leftarrow L_1, \ldots, L_n$ of a clause in $\mathbf{P}$ such that $I \models L_i$ for all $1 \leq i \leq n$. Furthermore $I \in I_{\mathbf{P}}$ is a supported model iff $I$ is a fixpoint of $T_{\mathbf{P}}$.

The two-valued semantics of normal logic programs is declared by two-valued interpretations. In the following subsection the semantics of a NLP is defined to be a supported model of $T_{\mathbf{P}}$. In each case the supported model of $\mathbf{P}$ will be unique. Therefore there will never exist an ambiguity concerning the choice of supported model.

3.1.1 Supported Model Semantics

We distinguish between some classes of NLPs using level mappings. We have

**Definition 3.1.2** Let $\mathbf{P}$ be a NLP. We call $\mathbf{P}$ locally hierarchical if for each clause $(A \leftarrow L_1, \ldots, L_n)$ in $\text{ground}(\mathbf{P})$ we have $l(A) > l(L_i)$ for all $1 \leq i \leq n$. If additionally $l$ is an $\omega$-level mapping, $\mathbf{P}$ is called acyclic. Locally hierarchical programs are sometimes also called strictly level decreasing.
We generalize the class of locally hierarchical NLPs and will see that each NLP in these classes has a unique supported model. We need the following definition (see [1]).

**Definition 3.1.3** Let \( P \) be a NLP and \( p, q \) be predicate symbols occurring in \( P \).

(a) \( p \) refers to \( q \) if there is a clause in \( P \) with \( p \) in its head and \( q \) in its body.

(b) \( p \) depends on \( q \) if \( (p, q) \) is in the reflexive, transitive closure of the relation refers to.

(c) \( \text{Neg}_P \) is the set of all predicate symbols in \( P \) which occur in a negative literal in the body of a clause of \( P \).

(d) \( \text{Neg}_P^+ \) is the set of all predicate symbols in \( P \) on which the predicatce symbols in \( \text{Neg}_P \) depend.

(e) \( P^- \) is the set of all clauses in \( P \) whose head contains a predicate symbol from \( \text{Neg}_P^+ \).

To define \( \Phi^* \)-accessible programs we need the following subset of \( B_P \).

\[
N = \{ p(t_1, \ldots, t_k) \in B_P \mid p \in \text{Neg}_P^+ \text{ k-ary predicate}, \ k \in \mathbb{N}_0, \ t_i \text{ ground term}, \ 1 \leq i \leq k \}.
\]

**Definition 3.1.4 (\( \Phi^* \)– and \( \Phi \)–accessible Programs)** A normal logic program \( P \) is called \( \Phi^* \)-accessible if there exists a level mapping \( l \colon B_P \to \gamma \) for \( P \) and a model \( I \in I_P \) of \( P \) such that \( I \cap N \) is a supported model of \( P^- \) and for each clause \( (A \leftarrow L_1, \ldots, L_k) \in \text{ground}(P) \) we either have

\[
I \models L_1 \land \ldots \land L_k \text{ and } l(A) > l(L_i), \ i = 1, \ldots, k \quad \text{or} \quad \exists l \leq i \leq k : I \not\models L_i \text{ and } l(A) > l(L_i).
\]

A normal logic program \( P \) is called \( \Phi \)-accessible if there exists a level mapping \( l \colon B_P \to \gamma \) for \( P \) and a model \( I \in I_P \) of \( P \) such that each \( A \in B_P \) satisfies either (i) or (ii).

(i) \( \exists (A \leftarrow L_1, \ldots, L_k) \in \text{ground}(P) : I \models L_1 \land \ldots \land L_k \text{ and } l(A) > l(L_i) \text{ for all } 1 \leq i \leq k. \)

(ii) \( I \not\models A \text{ and } \forall (A \leftarrow L_1, \ldots, L_k) \in \text{ground}(P) \exists 1 \leq i \leq k : I \not\models L_i \text{ and } l(A) > l(L_i). \)

A proof of the following theorem can be found in [21, Theorem 9], [22, Proposition 6.4] and [23, Theorem 4.3].

**Theorem 3.1.1** Let \( P \) be a \( \Phi^* \)-accessible or \( \Phi \)-accessible program. Then \( P \) has a unique supported model. \( \square \)

Later in this chapter we will give another proof of the above theorem. We will also see that one can get the fixpoint of the operator \( T_P \) constructively for each program \( P \) in that class.

Obviously locally hierarchical NLPs are also \( \Phi^* \)-accessible and it is not difficult to prove that \( \Phi^* \)-accessible programs are \( \Phi \)-accessible. We omit the proof.
3.1. SOME CLASSES OF LOGIC PROGRAMS AND THEIR SEMANTICS

3.1.2 Three-Valued Semantics of NLPs

It is generally known that the operator \( T_P \) need not have a supported model. Consider e.g. the NLP

\[ p(t) \leftarrow \neg p(t). \]

Whenever \( A \) is a ground instantiation of \( p(t) \) and \( A \in I_P \) we get \( A \not\in T_P(I) \). So \( T_P \) has no supported model.

There exist many such NLPs and if one wants to give these programs a meaningful semantics in general one extends the number of truth values and looks for the fixpoints of operators different from \( T_P \) which are acting on three-valued interpretations or many-valued valuations. We will use three-valued valuations in this chapter to express the semantics of programs. In the next chapter we will make a change in the notation and will use three-valued interpretations.

In the literature more than three truth values are not very often used. Bilattices are a way to generalize the set of truth values to nearly arbitrary sets e.g. one can choose an interval of \( \mathbb{R} \) to be the set of truth values (see [10]). It can be meaningful to choose such arbitrary sets (see e.g. [3]). One example of a bilattice with four truth values is the bilattice \( \mathcal{F}_4 \) which is explained in [10] and applied in [6]. We will only use three truth values.

Let \( \mathcal{B} = \{ t, u, f \} \) be our set of truth values where obviously \( t(u, f) \) are abbreviations for true (undefined, false), respectively. On \( \mathcal{B} \) we define two partial relations.

**Definition 3.1.5 (see also [10])** The partial relation \( \leq_k \) on \( \mathcal{B} \) is called knowledge direction and is defined by \( u \leq_k t \) and \( u \leq_k f \). The partial relation \( \leq_t \) on \( \mathcal{B} \) is called truth direction and is defined by \( f \leq_t u \leq_t t \).

One easily checks that \( (\mathcal{B}, \leq_k) \) is a complete semilattice and that \( (\mathcal{B}, \leq_t) \) is a complete lattice (see Definition 2.1.1). We can extend the definition of the relations \( \leq_k \) and \( \leq_t \) to valuations \( v: B_P \rightarrow \mathcal{B} \). For all valuations \( v, w \in \mathcal{B}^{B_P} \) we define

\[
\begin{align*}
v \leq_k w & \iff v(A) \leq_k w(A) \text{ for all } A \in B_P, \\
v \leq_t w & \iff v(A) \leq_t w(A) \text{ for all } A \in B_P.
\end{align*}
\]

The property of being a complete semilattice or lattice extends to \( (\mathcal{B}^{B_P}, \leq_k) \) and \( (\mathcal{B}^{B_P}, \leq_t) \), respectively. We now define an operator acting on the space of three-valued valuations.

**Definition 3.1.6 (The Operator \( \Phi_P \))** Let \( P \) be a NLP. The operator \( \Phi_P : \mathcal{B}^{B_P} \rightarrow \mathcal{B}^{B_P} \) is defined by

\[
\Phi_P(v)(A) = \bigvee \{ v(L_1 \land \ldots \land L_k) \mid \exists R \in \text{ground}(P): R = (A \leftarrow L_1, \ldots, L_k) \}.
\]

for all \( v \in \mathcal{B}^{B_P} \) and \( A \in B_P \). We use the convention \( v(t) = t \) for all \( v \in \mathcal{B}^{B_P} \) and \( v(\emptyset) = f \). The operator \( \Phi_P \) is called Fitting’s operator \( \Phi_P \).

Let \( P \) be a NLP and \( v_\perp: B_P \rightarrow \mathcal{B} \) be a valuation defined by \( v_\perp(A) = u \) for all \( A \in B_P \). Obviously \( v_\perp \) is the least element in \( (\mathcal{B}^{B_P}, \leq_k) \). One can check that \( \Phi_P \) is a monotonic operator with respect to \( \leq_k \). Therefore Proposition 2.1.3 can be applied, that is, there exists an ordinal number \( \alpha \) such that \( \text{Ifp}(\Phi_P) = \Phi_P \uparrow \alpha \) and \( \alpha \) is the closure ordinal of \( \Phi_P \).
34 CHAPTER 3. EXT. TO THE CLASSES OF $\Phi^+-$ AND $\Phi-$ACCESSIBLE PROGRAMS

Other operators which use more than two truth values exist. We mention the well-founded operator $W_p$ (see next chapter) and the three-valued stable model operator (see [35]) which are both defined for all NLPS. We have that $W_p$ is a monotonic operator on the set of three-valued interpretations and Theorem 2.1.1 is used to find the least fixpoint $\text{lp}(W_p)$ of $W_p$. The latter operator is an extension of the two-valued stable model operator (see [16]).

Some other techniques exist to get a three-valued model of a NLP. E.g., the alternating fixpoint operator $A_p$ (see next chapter) is defined on two-valued interpretations though the resulting model is three-valued and coincides with the well-founded model.

3.2 The Extended Program Classes

We will extend the results of [21] and [22]. Instead of 2-valued interpretations we consider 3-valued valuations. We define the class of extended $\Phi^+-$accessible, the class of 3-valued $\Phi^+-$accessible, the class of extended $\Phi-$accessible and the class of 3-valued $\Phi-$accessible programs. We extend the definition of normal logic program (NLP) to 3-valued NLP and generalize the immediate consequence operator $T_p$ to an extended version of Fitting’s operator $\Phi_p$ on 3-valued valuations where $P$ is a 3-valued NLP. The extended operator will also be denoted by $\Phi_p$. We will see that the operator $\Phi_p$ has a unique fixpoint for all the above classes of programs (except 3-valued NLPS). We prove that the class of $\Phi^+-$accessible programs coincides with the class of extended $\Phi^+-$accessible programs which implies that the unique fixpoints are 2-valued in that case. If $P$ is in the class of 3-valued $\Phi^+-$accessible or 3-valued $\Phi-$accessible programs $\Phi_p$ can have an unique fixpoint which is not 2-valued. We will give examples in each case.

3.2.1 Preliminary Definitions

**Definition 3.2.1 (3-valued NLP)** A 3-valued normal logic program $P$ consists of a finite set of normal logic program clauses (no facts, intensional part of the program) and a finite set of 3-valued facts (extensional part of the program). Each 3-valued fact has the form $A \leftarrow t$ or $A \leftarrow u$ where $A$ is an atom of the language underlying $P$.

Now let $P$ be a (3-valued) normal logic program.

**Definition 3.2.2 (Model)** A valuation $v : B_P \longrightarrow B$ of a (3-valued) normal logic program $P$ is a model of $P$ if for all $R \in \text{ground}(P)$ with head($R$) = $A$ we either have $R = (A \leftarrow L_1, \ldots, L_k)$ and $v(A) \geq 1, v(L_1 \land \ldots \land L_k)$ or $R = (A \leftarrow \alpha)$ and $v(A) \geq 1$ with $\alpha \in \{t, u\}$.

We are now in a position to extend Fitting’s operator $\Phi_p$ to 3-valued normal logic programs. We will denote the extended version again by $\Phi_p$.

**Definition 3.2.3 (The Operator $\Phi_p$)** We split the operator $\Phi_p : B^{B_p} \longrightarrow B^{B_p}$ in an extensional part $\Phi_p^e : B_p \longrightarrow B$ and an intensional part $\Phi_p^i : B^{B_p} \longrightarrow B^{B_p}$ by

$$
\Phi_p^e(v)(A) = \bigvee \{ v(L_1 \land \ldots \land L_k) \mid \exists \ R \in \text{ground}(P) : R = (A \leftarrow L_1, \ldots, L_k) \},
$$

$$
\Phi_p^i(A) = \bigvee \{ \alpha \mid \exists \ R \in \text{ground}(P) : R = (A \leftarrow \alpha) \},
$$

$$
\Phi_p(v)(A) = \Phi_p^e(v)(A) \lor \Phi_p^i(A),
$$

where $A \in B_p$. We use the convention $\bigvee \emptyset = f$. 

3.2. THE EXTENDED PROGRAM CLASSES

3.2.2 Extensions of the Class of $\Phi^*$-accessible Programs

The first attempt to generalize $\Phi^*$-accessible programs (compare with Definition 3.1.4) occurs by means of

Definition 3.2.4 (Extended $\Phi^*$-accessible Programs) A normal logic program $P$ is in the class of extended $\Phi^*$-accessible programs if there exists a level mapping $l : B_P \to \gamma$ for $P$ and a valuation $v_0 : B_P \to B$ which is a model of $P$ and such that $v_0|N$ is a fixpoint of $\Phi_P^*$ and for all $R \in \text{ground}(P)$ with $R = (A \leftarrow L_1, \ldots, L_k)$ we either have

$$v_0(L_1 \land \ldots \land L_k) \neq f \text{ and } l(A) > l(L_i), \quad i = 1, \ldots, k$$

or

$$\exists 1 \leq i \leq k : v_0(L_i) = f \text{ and } l(A) > l(L_i).$$

Consider an extended $\Phi^*$-accessible program $P$ with respect to a model $v_0$ and a level mapping $l : B_P \to \gamma$. Let $\Gamma = \{2^{-\alpha} \mid \alpha \leq \gamma\}$ be ordered by $2^{-\alpha} < 2^{-\beta}$ if $\beta < \alpha$ and denote $2^{-\gamma}$ by 0.

Now we have already seen that $(B, \leq_l)$ and $(B_P, \leq_l)$ are complete lattices. The least element $v_\bot$ of the complete lattice $B_P$ is defined by $v_\bot(A) = f$ for all $A \in B_P$. The compact elements of $B_P$ are exactly the valuations $v$ with the property $\{A \in B_P \mid v(A) \neq f\}$ is finite (see Theorem 2.3.1). We define a rank function $r_l$ on the set of compact elements $D_C$ depending on the level mapping $l$. We define $r_l(v_\bot) = 0$ and $r_l(v) = \max\{l(A) \mid A \in B_P, v(A) \neq f\}$ for all $v \in D_C, v \neq v_\bot$. Let $d_l$ be the generalized ultrametric (gum) depending on the rank function $r_l$ (see Definition 2.3.4). Then we have $d_l(v, v) = 0$ and $d_l(v_1, v_2) = 2^{-\alpha}$ iff $v_1$ and $v_2$ differ on an atom $A \in B_P$ of level $\alpha$ but agree on all $A \in B_P$ of lower level, that is

$$d_l(v_1, v_2) = 2^{-\alpha} \iff \alpha = \min\{\{l(A) \mid A \in B_P, v_1(A) \neq v_2(A)\} \cup \{\gamma\}\}$$

for all $v_1, v_2 \in B_P$. One can also show that the space $(B_P, d_l)$ is spherically complete (see [40, Theorem 3.5 and Proposition 3.7]) but we do not need this fact here. We define

$$v'(A) = \begin{cases} v(A) & \text{if } A \in N \\ f & \text{otherwise} \end{cases} \quad \text{and} \quad v''(A) = \begin{cases} f & \text{if } A \in N \\ v(A) & \text{otherwise} \end{cases}$$

Therefore we have $v' \lor v'' = v$ and $v' \land v'' = f$. Similar to [21] we define $d_l(v, w) = d(v', w')$ and $d_l(v, w) = d(v'', w'')$ for all $v, w \in B_P$. The function $f$ in [21] is now defined by $f : B_P \to \Gamma$ with

$$f(v) = 2^{-\alpha}, \quad \alpha = \min\{\{l(A) \mid v'(A) \neq f, v_0(A) = f\} \cup \{\gamma\}\}.$$

Let $\rho : B_P \times B_P \to \Gamma$ be given by $\rho(v, w) = \max\{d_l(v, v_0), d_l(w, v_0), d_l(v, w), f(v), f(w)\}$ for all $v, w \in B_P$. It is easy to see that $\rho$ is a d-gum. We show

Lemma 3.2.1 The space $(B_P, \rho)$ is spherically complete.

Proof. Let $I \subseteq \gamma$ be an index set, $C = (B_2^{-\alpha}(v_\alpha))_{\alpha \in I}$ be a chain of non-empty balls in $B_P$. We have to show that $\bigcap C \neq \emptyset$. We know that $B_2^{-\alpha}(v_\alpha) = \{w \in B_P \mid \rho(v, w) \leq 2^{-\alpha}\}$ and assume without loss of the generality that the members of the chain are all pairwise different. Let

\begin{equation}
B_2^{-\alpha}(v_\alpha) \supseteq B_2^{-\beta}(v_\beta), \alpha, \beta \in I, \alpha \neq \beta.
\end{equation}
Using [21, Lemma 5] we get $\alpha < \beta$ or equivalently $2^{-\alpha} > 2^{-\beta}$. As the balls of the chain are not empty we get

$$\forall \alpha \in I \exists \omega \in B_{2^{-\alpha}}(v_{\alpha}): \rho(\omega, v_{\alpha}) \leq 2^{-\alpha}. \tag{3.3}$$

The definition of $\rho$ yields

$$\rho\vert_{\mathcal{L}_\alpha} = v_{\alpha}\vert_{\mathcal{L}_\alpha} \tag{3.4}$$

where $\mathcal{L}_\alpha = \{ A \in \mathcal{B} \mathcal{P} \mid l(A) < \alpha \}$. We define a valuation $v_K : \mathcal{B} \mathcal{P} \rightarrow \mathcal{B}$ such that for all $A \in \mathcal{B} \mathcal{P}$ we have

$$v_K(A) = \begin{cases} v_{\alpha}(A) & \exists \alpha \in I: v_{\beta}(A) = v_{\alpha}(A) \forall \beta \in I, \beta \geq \alpha, \\ \forall \alpha \in I: v_{\beta}(A) = v_{\alpha}(A) & \text{otherwise.} \end{cases} \tag{3.5}$$

We will show that $v_K \in \mathcal{C}$ but first let $\alpha, \beta \in I, \alpha < \beta$. Now we know that (3.2) is true and it follows $w_{\beta} \in B_{2^{-\alpha}}(v_{\alpha})$. Using (3.4) we get $v_{\beta}\vert_{\mathcal{L}_{\alpha}} = w_{\beta}\vert_{\mathcal{L}_{\alpha}} = v_{\alpha}\vert_{\mathcal{L}_{\alpha}}$. Our intermediate result is

$$\forall \alpha, \beta \in I, \beta > \alpha: v_{\alpha}\vert_{\mathcal{L}_\alpha} = v_{\beta}\vert_{\mathcal{L}_\alpha}. \tag{3.6}$$

From (3.5) and (3.6) it follows

$$\forall \alpha \in I: v_K\vert_{\mathcal{L}_\alpha} = v_{\alpha}\vert_{\mathcal{L}_\alpha}. \tag{3.7}$$

We have to show $\rho(v_K, v_{\alpha}) \leq 2^{-\alpha}$ for all $\alpha \in I$. Equation (3.7) is equivalent to $d(v_K, v_{\alpha}) \leq 2^{-\alpha}$ for all $\alpha \in I$. Therefore we get $d_2(v_K, v_{\alpha}) \leq 2^{-\alpha}$ and it remains to show

$$d_1(v_K, v_{\alpha}) \leq 2^{-\alpha}, d_1(v_{\alpha}, v_0) \leq 2^{-\alpha}, f(v_K) \leq 2^{-\alpha} \text{ and } f(v_{\alpha}) \leq 2^{-\alpha}. \tag{3.7}$$

As $\rho(w_{\alpha}, v_{\alpha}) \leq 2^{-\alpha}$ we get $f(v_{\alpha}) \leq 2^{-\alpha}$ and $d_1(v_{\alpha}, v_0) \leq 2^{-\alpha}$. Equation (3.7) yields $d_1(v_K, v_0), f(v_K) \leq 2^{-\alpha}$. Altogether we have $\rho(v_K, v_{\alpha}) \leq 2^{-\alpha}$ which means $v_K \in B_{2^{-\alpha}}(v_{\alpha})$ for all $\alpha \in I$. \hfill \blacksquare

To prove that for every program in the class of extended $\Phi^l$–accessible programs the operator $\Phi_\mathcal{P}$ has a unique fixpoint we will show the following theorem. It is a reformulation of [21, Proposition 8] and we prove it for the more general case where the extensional part of $\mathcal{P}$ is $3$-valued.

**Theorem 3.2.2** Let $\mathcal{P}$ be in the class of extended $\Phi^l$–accessible programs with respect to the level mapping $l : \mathcal{B} \mathcal{P} \rightarrow \gamma$ and the model $v_0 \mathcal{P}$ of $\mathcal{P}$. Then for all $v, w \in \mathcal{B} \mathcal{P}, v \neq w$ we have $\rho(\Phi_\mathcal{P}(v), \Phi_\mathcal{P}(w)) \leq \rho(v, w)$. In particular we have

(i) $d_1(\Phi_\mathcal{P}(v), v_0) < d_1(v, v_0),$

(ii) $f(\Phi_\mathcal{P}(v)), f(\Phi_\mathcal{P}(w)) < \rho(v, w),$

(iii) $d_2(\Phi_\mathcal{P}(v), \Phi_\mathcal{P}(w)) < \rho(v, w).$
3.2. THE EXTENDED PROGRAM CLASSES

Proof. From (i), (ii) and (iii) the rest of the statement of the theorem follows.

(i) Let $d_1(v, v_0) = 2^{-\alpha}$. We have to show $d_1(\Phi_{P^{-}}(v), v_0) \leq 2^{-[\alpha+1]}$. Let $A \in N \cap L_{\alpha+1}$. If for all $R \in \text{ground}(P)$ we have $\text{head}(R) \neq A$ we get

$\Phi_{P^{-}}(v)(A) = f = \Phi_{P^{-}}(v_0)(A) = v_0(A),$

as $v_0|_N$ is a fixpoint of $\Phi_{P^{-}}$. Now let $R \in \text{ground}(P)$ with $R = (A \leftarrow L_1, \ldots, L_n)$. Assume we have $v_0(L_1 \land \ldots \land L_n) \neq f$. Applying Definition 3.2.4 we get $l(L_i) < l(A) \leq \alpha$ for all $1 \leq i \leq n$. Using the premise of (i) and the fact that for all $L_i$ the corresponding predicate symbols belong to $\text{Neg}_{P}$, it emerges $v(L_1 \land \ldots \land L_n) = v_0(L_1 \land \ldots \land L_n)$. Now assume $v_0(L_1 \land \ldots \land L_n) = f$.

Then there must exists $1 \leq i \leq n$ with $v_0(L_i) = f$. Applying Definition 3.2.4 we get $l(L_i) < l(A) \leq \alpha$ and from the premise of (i) together with the fact that the predicate which belongs to $L_i$ in $\text{Neg}_{P}$ it follows $v(L_i) = v_0(L_i) = f$. Therefore we have $v(L_1 \land \ldots \land L_n) = f$. Altogether we get in each case $v_0(L_1 \land \ldots \land L_n) = v(L_1 \land \ldots \land L_n)$ and as the extensional part of $\Phi_{P}$ is independent of the valuation applied it follows $\Phi_{P}(v)(A) = \Phi_{P}(v_0)(A) = v_0(A)$.

(ii) Let $g(v, w) = 2^{-\alpha}$. We show $f(\Phi_{P}(v)) \leq 2^{-[\alpha+1]}$. The other case is similar and we omit it. Let $A \in \text{B}_{P}$, $l(A) \leq \alpha$. Because of the definition of $f$ the proof reduces to show that $\Phi_{P}(v_0)^{0'}(A) = f \lor v_0(A) \neq f$. If $A \in N$ the definition of $0'$: $\text{B}_{P} \rightarrow \text{B}_{P}$ yields immediately the claim. Now let $A \notin N$. We assume $\Phi_{P}(v)(A) \neq f \land v_0(A) = f$. Then we either have

$\exists(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P): v(L_1 \land \ldots \land L_n) \neq f \land v_0(L_1 \land \ldots \land L_n) = f$ or

$\exists(A \leftarrow \alpha) \in \text{ground}(P), \alpha \in \{t, u\}: \Phi_{P}(v)(A) \geq \alpha \neq f \land f = v_0(A) \geq \alpha,$

as $v_0$ is a model of $P$ which means $f = v_0(A) \geq v_0(L_1 \land \ldots \land L_n)$ or $v_0(A) \geq \alpha$. The second case cannot exist and therefore there must exist $1 \leq i \leq n$ with $v_0(L_i) = f$. Applying Definition 3.2.4 yields $l(L_i) < l(A) \leq \alpha$. We distinguish two cases:

1. $L_i = A \lor L_i = \neg A_i$ and $A_i \in N$. As $d_1(v, v_0) \leq g(v, w) = 2^{-\alpha}$ we get $v(L_i) = v_0(L_i) = f$ and therefore $v(L_1 \land \ldots \land L_n) = f$ which is a contradiction.

2. We have $L_i = A_i \notin N$. The definition of $g$ yields $f(v) \leq g(v, w) = 2^{-\alpha}$. From the definition of $f$ it follows $v^{v'}(A_i) = v(A_i) = f \lor v_0(A_i) \neq f$ but both cases are contradictions to our assumption. Altogether our assumption is false and the claim of (ii) follows.

(iii) Let $g(v, w) = 2^{-\alpha}$. We have to show $d(\Phi_{P}(v)^{0}, \Phi_{P}(w)^{0}) \leq 2^{-[\alpha+1]}$. Let $A \notin N$. If for all $R \in \text{ground}(P)$ we have $\text{head}(R) \neq A$ it follows

$(3.8) \quad \Phi_{P}(v)(A) = f = \Phi_{P}(w)(A)$

and we are finished. Now let $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$. We distinguish two cases:

1. Let $v_0(L_1 \land \ldots \land L_n) \neq f$. Then Definition 3.2.4 yields $l(L_i) < l(A) \leq \alpha$ for all $1 \leq i \leq n$. Using $d_1(v, v_0), d_1(w, v_0), d_2(v, w) \leq g(v, w) \leq 2^{-\alpha}$ we get $v(L_1 \land \ldots \land L_n) = w(L_1 \land \ldots \land L_n)$ and we are finished.

2. There exists $1 \leq i \leq n$ with $v_0(L_i) = f$. Then Definition 3.2.4 yields $l(L_i) < l(A) \leq \alpha$. We have once again to distinguish two cases. First let $L_i = A_i \lor L_i \equiv \neg A_i$ and $A_i \in N$. Because of $d_1(v, v_0), d_1(w, v_0) \leq g(v, w) \leq 2^{-\alpha}$ we get $v(L_i) = v_0(L_i) = w(L_i) = f$. It follows $v(L_1 \land \ldots \land L_n) = w(L_1 \land \ldots \land L_n) = f$ which completes the first case. Now let $L_i = A_i$ and $A_i \notin N$. Because of $f(v), f(w) \leq g(v, w) \leq 2^{-\alpha}$ and the definition of $f$ we get $v(A_i) = w(A_i) = f$. This yields $v(L_1 \land \ldots \land L_n) = w(L_1 \land \ldots \land L_n) = f$ which completes the second case.

Altogether we always get $v(L_1 \land \ldots \land L_n) = w(L_1 \land \ldots \land L_n)$ and as the extensional part of $\Phi_{P}$ is independent of the valuation applied it emerges $\Phi_{P}(v)(A) = \Phi_{P}(w)(A)$.
Applying Theorem 2.2.4 we get the result

**Theorem 3.2.3** Let $P$ be in the class of extended $\Phi^*$-accessible programs. Then $\Phi_P$ has a unique fixpoint.

Indeed we can show much more

**Lemma 3.2.4** The class of extended $\Phi^*$-accessible programs is equal to the class of $\Phi^*$-accessible programs. Let $P$ be in the class of extended $\Phi^*$-accessible programs with respect to the level mapping $l: B_P \rightarrow \gamma$ and the valuation $v_0: B_P \rightarrow B$. Then we can construct a 2-valued valuation $v'_0: B_P \rightarrow \{t, f\}$ which coincides with $v_0$ on $N$ (particularly $v_0|_N$ is 2-valued) such that $P$ is in the class of $\Phi^*$-accessible programs with respect to the level mapping $l$ and the valuation $v'_0$. The unique fixpoint $w$ of $\Phi_P$ is 2-valued and $w|_N = v_0|_N$.

**Proof.** Let $P$ be a program in the class of extended $\Phi^*$-accessible programs. First we show that the unique fixpoint of $\Phi_P$ is 2-valued. We use the construction of the transfinite sequence in the proof of Theorem 2.2.5. Let $w_0$ be the valuation with $w_0(A) = f$ for all $A \in B_P$ and $w_{\alpha+1} = \Phi_P(w_\alpha)$ for all $\alpha < \gamma$. As $(B_B, \phi, \Gamma)$ is spherically complete and $(B_{2^{-\beta}}(w_\beta))_{\beta < \alpha}$ is a decreasing chain of non-empty balls in $B^*_B$ where $\alpha$ is a limit ordinal we get a valuation $w'_\alpha$ with

$$\phi(w'_\alpha, w_\beta) \leq 2^{-\beta} \quad \forall \beta < \alpha.$$ \hspace{1cm} (3.9)

As $w'_\alpha$ can be 3-valued we define a 2-valued valuation $w_\alpha: B_P \rightarrow \{t, f\}$ with

$$w_\alpha(A) = \begin{cases} w'_\alpha(A) & A \in L_\alpha, \\ f & A \in B_P \setminus L_\alpha. \end{cases}$$

Using the definition of $\phi$ equation (3.9) yields $d_1(w'_\alpha, v_0), f(w'_\alpha) \leq 2^{-\alpha}$ and $d_1(w_\beta, v_0), d_2(w'_\alpha, w_\beta), f(w_\beta) \leq 2^{-\beta}$ for all $\beta < \alpha$. From the definition of $w_\alpha$ it follows $\phi(w_\alpha, w_\beta) \leq 2^{-\beta}$ for all $\beta < \alpha$ and therefore $w_\alpha$ is in the intersection of the balls $(B_{2^{-\beta}}(w_\beta))_{\beta < \alpha}$. The proof of Theorem 2.2.5 now yields that $w_\alpha$ is a fixpoint of $\Phi_P$. The construction of the sequence $(w_\alpha)$ together with the definition of $\Phi_P$ shows that each $w_\alpha$ is 2-valued. Particularly the unique fixpoint $w_\alpha$ is 2-valued.

Because of $\phi(w_\alpha, \Phi_P(w_\gamma)) = 0$ the definition of $\phi$ yields $d_1(w_\gamma, v_0) = 0$ which means that $w_\gamma$ and $v_0$ coincide on $N$. Particularly $v_0|_N$ is 2-valued.

As $v_0$ can be 3-valued on $B_P \setminus N$ we construct a new valuation $v'_0: B_P \rightarrow \{t, f\}$ by

$$v'_0(A) = \begin{cases} v_0(A) & A \in N, \\ t & A \in B_P \setminus N \text{ and } v_0(A) \geq u, \\ f & \text{otherwise}. \end{cases}$$

We show that $v'_0$ is a model of $P$. Let $A \in B_P$ and $R \in \text{ground}(P)$ with $R = (A \leftarrow L_1, \ldots, L_k)$. If $A \in N$ we have $v'_0(A) = v_0(A) \geq t$, $v_0(L_1 \wedge \ldots \wedge L_k) = v'_0(L_1 \wedge \ldots \wedge L_k)$ as the atoms corresponding to the literals $L_i$ belong to $N$ and $v_0$ is a model of $P$. Now let $A \in B_P \setminus N$. If $v_0(A) \geq t$ we have trivially $v'_0(A) = t \geq v'_0(L_1 \wedge \ldots \wedge L_k) = f$. If $v_0(A) = f$ we get $v'_0(A) = v_0(A) \geq t$, $v_0(L_1 \wedge \ldots \wedge L_k) = f$. We conclude that there exists $1 \leq i \leq k$ with $v_0(L_i) = f$. Both $L_i$ negative and $L_i$ positive yield $v'_0(L_i) = v_0(L_i) = f$ and we have $v'_0(L_1 \wedge \ldots \wedge L_k) = f$. So we are finished.
3.2. THE EXTENDED PROGRAM CLASSES

The condition on the level mapping in Definition 3.2.4 is satisfied for \( v'_0 \) as from \( v_0(L_i) = f \) and the definition of \( v'_0 \) it follows \( v'_0(L_i) = f \). Altogether \( \mathbf{P} \) is a \( \Phi^* \)-accessible program with respect to the level mapping \( l \) and the 2-valued valuation \( v'_0 \).

Now we are able to construct the unique fixpoint of an extended \( \Phi^* \)-accessible program \( \mathbf{P} \) in another way. Let

\[
\mathbf{P}' = \{ R' \mid \exists R \in \text{ground}(\mathbf{P}), \text{head}(R) \not\in N : \neg \text{pos}(R) \subseteq v_0^{-1}(f), \text{pos}(R) \cap N \subseteq v_0^{-1}(t), \\
\text{head}(R') = \text{head}(R), \text{pos}(R') = \text{pos}(R) \cap \text{Bp} \setminus N, \neg \text{pos}(R') = \emptyset \}.
\]

Of course \( \mathbf{P}' \) is a positive logic program and Tarski's theorem (see Theorem 2.1.1) together with Proposition 2.1.2 yields an interpretation \( I = \mathcal{T}_{\mathbf{P}'}(\emptyset) \subseteq \text{Bp} \setminus N \) which is a fixpoint of \( \mathbf{T}_{\mathbf{P}'} \). We construct a 2-valued valuation \( v \) by

\[
v(A) = \begin{cases} 
v_0(A) & A \in N, \\ t & A \in I, \\ f & A \in \text{Bp} \setminus (N \cup I) \end{cases}
\]

and show that \( v \) is the unique fixpoint of \( \Phi_{\mathbf{P}} \). Of course \( v|_N \) is a fixpoint of \( \Phi_{\mathbf{P}-} \). So let \( A \in \text{Bp} \setminus N \). If for all \( R \in \text{ground}(\mathbf{P}) \) we have \( \text{head}(R) \neq A \) we can conclude \( A \not\in I \) as well as \( \Phi_{\mathbf{P}}(v)(A) = f \) and we are finished.

Now let \( A \in I \). Then there exists \( R' \in \mathbf{P}' \) with \( \text{head}(R') = A \) and \( \text{pos}(R') \subseteq I \). It follows the existence of \( R \in \text{ground}(\mathbf{P}) \) with \( R = (A \leftarrow L_1, \ldots, L_k) \) and \( \neg \text{pos}(R) \subseteq (v_0|_N)^{-1}(f) \subseteq N \) as well as \( \text{pos}(R) \subseteq I \cup (v_0|_N)^{-1}(t) \). The definition of \( v \) yields \( v(L_1 \wedge \ldots \wedge L_k) = t = \Phi_{\mathbf{P}}(v)(A) = v(A) \).

Let \( A \not\in I \) and consider whatever \( R \in \text{ground}(\mathbf{P}) \) with \( R = (A \leftarrow p_1, \ldots, p_k, \neg q_1, \ldots, \neg q_m) \). We have \( v(A) = f \) and we must show \( v(p_1 \wedge \ldots \wedge p_k \wedge \neg q_1 \wedge \ldots \wedge \neg q_m) = f \). If \( \neg \text{pos}(R) \subseteq (v_0|_N)^{-1}(f) \) or \( \text{pos}(R) \cap N \subseteq v_0^{-1}(t) \) we are finished. So let \( \neg \text{pos}(R) \subseteq (v_0|_N)^{-1}(f) \) and \( \text{pos}(R) \cap N \subseteq v_0^{-1}(t) \). Then there exists \( R' \in \mathbf{P}' \) with \( R' = (A \leftarrow p'_1, \ldots, p'_n) \) and \( \text{pos}(R') = \text{pos}(R) \cap \text{Bp} \setminus N \). As \( A \not\in I \) there must exist \( 1 \leq i \leq k \) with \( p_i \not\in I \cup N \). We get \( v(p_i) = f \) and are finished.

**Example 3.2.5** As we have seen the model \( v_0 \) of \( \mathbf{P} \) in Definition 3.2.4 is always 2-valued on \( N \subseteq \text{Bp} \). This need not to be the case on \( \text{Bp} \setminus N \). We consider the simple normal logic program \( A \leftarrow B \) where \( A \) and \( B \) are ground atoms. Let \( v_0(A) = v_0(B) = u = 1 > 0 = l(B) \). Of course \( \mathbf{P} \) is extended \( \Phi^* \)-accessible with respect to the model \( v_0 \) and the level mapping \( l \). In this case \( N \) is empty and \( v_0 \) is not 2-valued.

Though we have more freedom in the definition of extended \( \Phi^* \)-accessible programs than in the definition of \( \Phi^* \)-accessible programs these classes coincide. As our aim is to get a larger class of programs we obviously must extend the freedom in the new class in which we will consider.

**Definition 3.2.5 (3-valued \( \Phi^* \)-accessible Programs)** A 3-valued NLPP \( \mathbf{P} \) is in the class of 3-valued \( \Phi^* \)-accessible programs if there exists a level mapping \( l: \text{Bp} \rightarrow \gamma \) for \( \mathbf{P} \) and a valuation \( v_0: \text{Bp} \rightarrow \mathbb{B} \) which is a model of \( \mathbf{P} \) and such that \( v_0|_N \) is a fixpoint of \( \Phi_{\mathbf{P}-} \) and for all \( R \in \text{ground}(\mathbf{P}) \) with \( R = (A \leftarrow L_1, \ldots, L_k) \) we either have

\[
v_0(L_1 \wedge \ldots \wedge L_k) \neq f \text{ and } l(A) > l(L_i), i = 1, \ldots, k
\]

or

\[
\exists 1 \leq i \leq k: v_0(L_i) = f \text{ and } l(A) > l(L_i).
\]
CHAPTER 3. EXT. TO THE CLASSES OF $\Phi^s$- AND $\Phi$-ACCESSIBLE PROGRAMS

We already mentioned that Theorem 3.2.2 is also true in the case where $P$ is a 3-valued NLP which means that it is true when $P$ is in the class of 3-valued $\Phi^s$-accessible programs. Applying Theorem 2.2.4 we get the result:

**Corollary 3.2.6** Let $P$ be in the class of 3-valued $\Phi^s$-accessible programs. Then $\Phi_P$ has a unique fixpoint. □

We give some examples of 3-valued $\Phi^s$-accessible programs which do not have a 2-valued fixpoint.

**Example 3.2.7 (3-valued $\Phi^s$-accessible Programs)** The simplest program $P$ which one can find is $A \leftarrow u$ where $A$ is a ground atom. It has of course the unique fixpoint $v(A) = u$ and is in the class of 3-valued $\Phi^s$-accessible programs. Here $N$ is empty.

Now let $P$ be the program

\[
\begin{align*}
A & \leftarrow \neg B \\
B & \leftarrow u.
\end{align*}
\]

$P$ is once again in the class of 3-valued $\Phi^s$-accessible programs with respect to the valuation $v_0$ and the level mapping $l$ where $v_0(A) = v_0(B) = u$ and $l(A) = 1 > 0 = l(B)$. Here $N = \{B\}$ and $v_0$ is not 2-valued on $N$. Furthermore $v_0$ is the unique fixpoint of $P$.

These examples show that the class of 3-valued $\Phi^s$-accessible programs is a proper superset of the class of extended $\Phi^s$-accessible programs. So let $\Phi^s_2$ be the class of $\Phi^s$-accessible programs, $\Phi^s_{2,\nu}$ be the class of extended $\Phi^s$-accessible programs and $\Phi^s_3$ be the class of 3-valued $\Phi^s$-accessible programs. Then we get

$$\Phi^s_2 = \Phi^s_{2,\nu} \subseteq \Phi^s_3.$$

### 3.2.3 Extensions of the Class of $\Phi$-Accessible Programs

Next we want to extend the definition of $\Phi$-accessible programs given in Definition 3.1.4 to the 3-valued case.

**Definition 3.2.6 (3-valued $\Phi$-Accessible Programs)** A 3-valued NLP $P$ is in the class of $\Phi$-accessible programs if there exists a level mapping $l : B_P \to \gamma$ for $P$ and a model $v_0 : B_P \to B$ of $P$ such that each $A \in B_P$ satisfies either (i), (ii) or (iii).

(i) \( \exists (A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P) : v_0(L_1 \land \ldots \land L_n) = t \text{ and } l(A) > l(L_i) \text{ for all } 1 \leq i \leq n \text{ or } \Phi^s_P(A) = t. \)

(ii) (a) \( \exists (A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P) : v_0(L_1 \land \ldots \land L_n) = u \text{ and } l(A) > l(L_i) \text{ for all } 1 \leq i \leq n \text{ or } \Phi^s_P(A) = u. \)

(b) \( \forall (A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P) \exists i \leq i \leq n : v_0(L_i) \neq t \text{ and } l(A) > l(L_i). \)

(iii) \( v_0(A) = f \text{ and } \forall (A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P) \exists i \leq i \leq n : v_0(L_i) = f \text{ and } l(A) > l(L_i). \)

**Remark 3.2.8** The valuation $v_0$ in Definition 3.2.6 is a fixpoint of the operator $\Phi_P$.  |
3.2. THE EXTENDED PROGRAM CLASSES

For the rest of the text let \( \varphi : \mathcal{B} \times \mathcal{B} \rightarrow \Gamma \) be the dislocated generalized ultrametric corresponding to a 3-valued \( \Phi \)-accessible program \( P \) with respect to the fixpoint \( v_0 \) of \( P \) and defined by

\[
\varphi(v, w) = \max\{d(v, v_0), d(w, v_0)\}.
\]

As every nonempty ball in the d-gum \((\mathcal{B}, \varphi)\) contains \( I \) we conclude that \((\mathcal{B}, \varphi)\) is spherically complete. We prove that \( \Phi \) is strictly contracting with respect to \((\mathcal{B}, \varphi)\).

**Lemma 3.2.9** Let \( P \) be in the class of 3-valued \( \Phi \)-accessible programs with respect to the level mapping \( l : B_\mathcal{P} \rightarrow \gamma \) and the model \( v_0 \) of \( P \). Then \( \Phi \) is strictly contracting with respect to \((\mathcal{B}, \varphi)\).

**Proof.** Let \( v, w : B_\mathcal{P} \rightarrow \mathcal{B} \) be valuations with \( \varphi(v, w) = 2^{-\alpha}, \alpha < \gamma \). Let \( A \in \mathcal{B} \) with \( l(A) \leq \alpha \). We distinguish three cases:

(i) Let \( v_0(A) = t \). Then the first condition in Definition 3.2.6 can only be true. If there exists \( A \xleftarrow{} L_1, \ldots, L_n \in \text{ground}(P) \) with \( v_0(L_1 \land \ldots \land L_n) = t \) we get \( l(A) > l(L_i) \) for all \( 1 \leq i \leq n \). Using our premise concerning \( \varphi \) we conclude \( v(L_1 \land \ldots \land L_n) = v_0(L_1 \land \ldots \land L_n) = t \). Otherwise we have \( \Phi(P)(A) = t \). In both cases we get the result \( \Phi(P)(v)(A) = \Phi(P)(w)(A) = v_0(A) = t \).

(ii) Let \( v_0(A) = u \). The second condition in Definition 3.2.6 can only be true. So we have \( \Phi(P)(A) = u \) or there exists a clause \( A \xleftarrow{} L'_1, \ldots, L'_n \in \text{ground}(P) \) with \( v_0(L'_1 \land \ldots \land L'_n) = u \) and \( l(A) > l(L'_i) \) for all \( 1 \leq i \leq n \). Using the premise concerning \( \varphi \) we conclude \( v(L'_1 \land \ldots \land L'_n) = v_0(L'_1 \land \ldots \land L'_n) = u \). For all other clauses \( A \xleftarrow{} L'_1, \ldots, L'_n \in \text{ground}(P) \) there exists \( 1 \leq i \leq m \) with \( v_0(L'_i) \neq t \) and \( l(A) > l(L'_i) \). Our premise concerning \( \varphi \) results in \( v(L'_1 \land \ldots \land L'_n), v_0(L'_1 \land \ldots \land L'_n) \leq u \). Altogether we get \( \Phi(P)(v)(A) = \Phi(P)(w)(A) = v_0(A) = u \) as \( v_0 \) is a model of \( P \).

(iii) Let \( v_0(A) = f \). Then the third case in the condition of Definition 3.2.6 can only be true. Therefore for all \( A \xleftarrow{} L_1, \ldots, L_n \in \text{ground}(P) \) there exists \( 1 \leq i \leq n \) with \( v_0(L_i) = f \) and \( l(A) > l(L_i) \). Using our premise concerning \( \varphi \) we get \( v(L_i) = v_0(L_i) = w(L_i) = f \) which yields \( v(L_1 \land \ldots \land L_n) = v(L_1 \land \ldots \land L_n) = f \) and finally \( \Phi(P)(v)(A) = \Phi(P)(w)(A) = v_0(A) = f \) as \( v_0 \) is a model of \( P \).

Applying once again Theorem 2.2.4 we get

**Lemma 3.2.10** Let \( P \) be in the class of 3-valued \( \Phi \)-accessible programs. Then \( \Phi \) has a unique fixpoint.

We will give some examples of 3-valued \( \Phi \)-accessible programs which have not a 2-valued unique fixpoint. We will see that there are 3-valued \( \Phi \)-accessible programs which are not in the class of 3-valued \( \Phi^1 \)-accessible programs. So we get proper subsets in both cases (see below). We want to distinguish between 3-valued \( \Phi \)-accessible programs which are NLPs and such which are only 3-valued NLPs. Therefore we define the class of extended \( \Phi \)-accessible programs to be all NLPs in the class of 3-valued \( \Phi \)-accessible programs. As in the case of extended \( \Phi^1 \)-accessible programs the unique fixpoint of an extended \( \Phi \)-accessible program \( P \) is 2-valued.

**Lemma 3.2.11** Let \( P \) be in the class of extended \( \Phi \)-accessible programs. Then the unique fixpoint of the operator \( \Phi \) is 2-valued.
Proof. Let $v_0$ be the unique fixpoint of $P$ and $M = \{ l(A) \mid v_0(A) = u, A \in B_P \}$. Assume that $v_0$ is not 2-valued. Then $M$ is a non-empty subset of the class of ordinal numbers. Therefore there exists $A \in B_P$ such that $l(A)$ is the least element of $M$. As $v_0(A) = u$ and $P$ is a NLP there must exists a clause $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_0(L_1 \land \ldots \land L_n) = u$ and $l(A) > l(L_i)$ for all $1 \leq i \leq n$. We can conclude that there exists $1 \leq i \leq n$ with $v_0(L_i) = u$ and $l(L_i) < l(A)$ which is a contradiction to the premise that $l(A)$ is the least element in $M$. So $v_0$ must be 2-valued.

Corollary 3.2.12 The class of $\Phi$-accessible programs and the class of extended $\Phi$-accessible programs coincide.

Proof. As every extended $\Phi$-accessible program $P$ is a NLP and as it has a unique fixpoint which is 2-valued the second condition in Definition 3.2.6 can never be true and it satisfies all conditions of being a $\Phi$-accessible program. The opposite direction is trivial.

We assume that it is clear that every $\Phi^*_3$-accessible program is $\Phi$-accessible but we show the following

Lemma 3.2.13 Every 3-valued $\Phi^*_3$-accessible program $P$ is also 3-valued $\Phi$-accessible.

Proof. Let $P$ be a 3-valued $\Phi^*_3$-accessible program. We define a transfinite sequence of valuations $(v_\alpha)$ starting from $v_0$. For every ordinal number $\alpha$ let

$$M_\alpha = \{ A \in B_P \mid \Phi_P^*(A) \neq t, v_\alpha(A) = t \text{ and replacing } v_0 \text{ through } v_\alpha \text{ condition (ii) in Definition } 3.2.6 \text{ is true except } v_\alpha(A) = u \},$$

$$N_\alpha = \{ A \in B_P \mid \Phi_P^*(A) = f \text{ and replacing } v_0 \text{ through } v_\alpha \text{ condition (iii) in Definition } 3.2.6 \text{ is true except } v_\alpha(A) = f \},$$

$$v_\alpha(A) = \begin{cases} v_0(A) & A \in B_P \setminus \bigcup_{\beta < \alpha} (M_\beta \cup N_\beta), \\ u & A \in \left( \bigcup_{\beta < \alpha} M_\beta \setminus \bigcup_{\beta < \alpha} N_\beta \right), \\ f & A \in \bigcup_{\beta < \alpha} N_\beta. \end{cases}$$

Let $\alpha$ and $\beta$ be ordinal numbers. We prove

1. $M_\alpha \cap M_\beta = \emptyset, N_\alpha \cap N_\beta = \emptyset, \alpha \neq \beta$ and $v_\alpha \leq_1 v_\beta$, $\alpha > \beta$ as well as $v_\alpha|_N = v_\beta|_N$ for all $\alpha, \beta$.

First let $\beta < \alpha$ and $A \in M_\beta$. We immediately conclude $v_\alpha(A) \leq_1 u$ and get $A \notin M_\alpha$. So we have $M_\alpha \cap M_\beta = \emptyset$. A similar conclusion shows $N_\alpha \cap N_\beta = \emptyset$. We prove $v_\alpha \leq_1 v_\beta$. Let $A \in B_P$. The case $v_\alpha(A) = f$ is trivial. So let $v_\alpha(A) = t$. We get $A \notin \bigcup_{\beta < \alpha} (M_\beta \cup N_\delta)$ and conclude $v_\beta(A) = t$. Let $v_\alpha(A) = u$ and assume $v_\beta(A) = f$. If $v_\alpha(A) = f$ we would for every ordinal $\delta$ never have $A \in M_\delta$ and therefore we would have $v_\alpha(A) = f$ which is a contradiction. Otherwise there exists $A \in \bigcup_{\delta < \beta} N_\delta$ and conclude $v_\alpha(A) = f$ which is once again a contradiction. So $v_\beta(A) \geq_1 u$ and we are finished.

Next we prove $v_\alpha|_N = v_\alpha|_N$ for all ordinals $\alpha$. Let $A \in N$ and assume that for all $\beta < \alpha$ we have already shown $v_\beta|_N = v_\beta|_N$. Let $A \in \bigcup_{\beta < \alpha} N_\beta$. Then there exists $\beta < \alpha$ with
3.2. THE EXTENDED PROGRAM CLASSES

$A \in N_\beta$. The definition of $N_\beta$ and our assumption yield $\Phi_P(v_\beta)(A) = f = v_\beta(A)$ which is a contradiction to the definition of $N_\beta$. Now let $A \in \bigcup_{\beta < \alpha} M_\beta$. Then there exists $\beta < \alpha$ with $A \in M_\beta$. The definition of $M_\beta$ and our assumption yield $\Phi_P(v_\beta)(A) = u = v_\beta(A)$ which is a contradiction to the definition of $M_\beta$. We conclude $v_\alpha(A) = v_0(A)$ and are finished.

Because of set theoretic considerations there must exist an ordinal $\alpha$ such that $v_{\alpha+1} = v_\alpha$. Let $\alpha$ be the least such ordinal. We prove that $v_\alpha$ is a fixpoint of $\Phi_P$ and that for all $A \in B_P$ exactly one condition in Definition 3.2.6 is satisfied. We distinguish three cases:

(1) Let $v_\alpha(A) = t$. If $\Phi_P(v_\alpha)(A) = t$ the first condition in Definition 3.2.6 is satisfied and we are already finished.

Let $\Phi_P(v_\alpha)(A) = u$. As $M_\alpha = \emptyset$ the second condition in Definition 3.2.6 is not satisfied for $v_\alpha$ and therefore there must exist $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_\alpha(L_1 \land \ldots \land L_n) = t$. Because of $v_\alpha \leq v_0$ we get $v_0(L_1 \land \ldots \land L_n) = t$ and Definition 3.2.5 yields $l(A) > l(L_i)$ for $1 \leq i \leq n$, that is, condition (i) in Definition 3.2.6 is satisfied for $v_\alpha$.

Now let $\Phi_P(v_\alpha)(A) = f$. As $M_\alpha = N_\alpha = \emptyset$ condition (ii) and (iii) in Definition 3.2.6 are not satisfied for $v_\alpha$. From (iii) we conclude that there exists $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_\alpha(L_1 \land \ldots \land L_n) \geq u$ and together with (ii) we get $v_\alpha(L_1, \ldots, L_n) = t$. Using $v_\alpha \leq v_0$ we have $v_0(L_1 \land \ldots \land L_n) = t$ and Definition 3.2.5 yields $l(A) > l(L_i)$ for $1 \leq i \leq n$.

In each case we obtain $\Phi_P(v_\alpha)(A) = v_\alpha(A) = t$.

(2) Let $v_\alpha(A) = u$. If $v_0(A) = t$ there exists $\beta < \alpha$ with $A \in M_\beta$ and $\Phi_P(v_\beta)(A) \leq t$. From the definition of $M_\beta$ we know that for all $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ there exists $1 \leq i \leq n$ with $v_\beta(L_i) \neq t$ and $l(A) > l(L_i)$. Using (3.10) we have $v_\alpha(L_i) \neq t$. If $\Phi_P(v_\alpha)(A) = u$ we are finished. So let $\Phi_P(v_\alpha)(A) = f$. Condition (ii) (a) in Definition 3.2.6 must be true for $v_\beta$. Therefore there exists $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_\beta(L_1 \land \ldots \land L_n) = u$ and $l(A) > l(L_i)$ for all $1 \leq i \leq n$. From (3.10) we obtain $v_\alpha(L_1 \land \ldots \land L_n) = u$ and (ii) in Definition 3.2.6 is satisfied for $v_\alpha$.

Let $v_0(A) = u$. Assume that there exists $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_\alpha(L_1 \land \ldots \land L_n) = t$. We get $v_0(L_1 \land \ldots \land L_n) = t$ and because $v_0$ is a model of $P$ we conclude $v_0(A) = t$ which is a contradiction to our premise. So (ii) (b) in Definition 3.2.6 is satisfied for $v_\alpha$. If $\Phi_P(v_\alpha)(A) = u$ the whole condition (ii) is satisfied for $v_\alpha$ and we are finished. Let $\Phi_P(v_\alpha)(A) = f$. As $N_\alpha = \emptyset$ there must exist $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ with $v_\alpha(L_1 \land \ldots \land L_n) \geq u$ and because (ii) (b) is satisfied for $v_\alpha$ we conclude $v_\alpha(L_1 \land \ldots \land L_n) = u$, that is, (ii) (a) is satisfied and we are finished.

In each case we obtain $\Phi_P(v_\alpha)(A) = v_\alpha(A) = u$.

(3) Let $v_\alpha(A) = f$. If $v_0(A) = f$ for all $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ there exists $1 \leq i \leq n$ with $v_0(L_i) = f$ and $l(A) > l(L_i)$. Using (3.10) we obtain $v_\alpha(L_i) = f$, that is, condition (iii) in Definition 3.2.6 is satisfied for $v_\alpha$.

Let $v_0(A) \neq f$. Then there exists $\beta < \alpha$ with $A \in N_\beta$. We conclude $\Phi_P(v_\alpha)(A) = f$ and for all $(A \leftarrow L_1, \ldots, L_n) \in \text{ground}(P)$ there exists $1 \leq i \leq n$ with $v_\beta(L_i) = f$ and $l(A) > l(L_i)$. Because of (3.10) we get $v_\alpha(L_i) \leq v_\beta(L_i) = f$ and condition (iii) in Definition 3.2.6 is satisfied for $v_\alpha$.

In each case we obtain $\Phi_P(v_\alpha)(A) = v_\alpha(A) = f$.

Example 3.2.14 (3-valued $\Phi$-accessible Programs) Of course the above examples of 3-valued $\Phi$-accessible programs are also examples of 3-valued $\Phi$-accessible programs which are not $\Phi$-accessible as they do not have a 2-valued fixpoint.
Consider the following NLP P
\[
A \leftarrow \neg A \\
A \leftarrow B \\
B \leftarrow t.
\]
Let \(1 = l(A) > l(B) = 0\) be the definition of a level mapping for P and \(v_0(A) = v_0(B) = t\) be the definition of a valuation of P. Clearly \(v_0\) is the unique fixpoint of \(\Phi_P\) and P is a 3-valued \(\Phi\)-accessible program with respect to the level mapping \(l\) and the fixpoint \(v_0\). P is not a 3-valued \(\Phi'\)-accessible program because of the first clause but as P is a NLP it is also (extended) \(\Phi\)-accessible and as proved above it must have a fixpoint which is 2-valued.

Now consider the following 3-valued NLP P
\[
A \leftarrow \neg A, B \\
A \leftarrow \neg B, C \\
B \leftarrow u \\
C \leftarrow u.
\]
We use the level mapping \(l\) given by \(1 = l(A) > l(B), l(C) = 0\) and the valuation \(v_0\) defined by \(v_0(A) = v_0(B) = v_0(C) = u\). Obviously \(v_0\) is a fixpoint of P and P is 3-valued \(\Phi\)-accessible with respect to the level mapping \(l\) and the model \(v_0\). Of course P is neither extended \(\Phi\)-accessible nor \(\Phi\)-accessible as P is not a NLP. P cannot be 3-valued \(\Phi'\)-accessible with respect to any model \(v_0\), too. In order to prove this claim assume \(v_0\) is such a model of P. We must have \(v_0(A) \geq_1 u\). Because of the first clause we conclude \(v_0(A) \geq_1 u\). So we either have \(v_0(\neg A \land B) \neq f\) and get a contradiction to the first condition in the definition of 3-valued \(\Phi'\)-accessible programs or we have \(v_0(\neg A) = f\) and get a contradiction to the second condition in the definition of 3-valued \(\Phi'\)-accessible programs.

Let P be the following NLP
\[
A \leftarrow B \\
B \leftarrow A \\
B \leftarrow C \\
B \leftarrow t \\
C \leftarrow t.
\]
Let \(l\) be the level mapping given by \(2 = l(A) > l(B) = 1 > l(C) = 0\) and \(v_0\) be the valuation defined by \(v_0(A) = v_0(B) = v_0(C) = t\). Then \(v_0\) is a supported model of P and P is \(\Phi\)-accessible with respect to the level mapping \(l\) and the supported model \(v_0\). P is not a 3-valued \(\Phi'\)-accessible program because then we could conclude the impossible condition \(l(A) > l(B) > l(A)\).

### 3.2.4 Summary

Summarizing the above results we got some different classes of programs which all have a unique fixpoint. We saw that there exist some subset relationships between these classes of programs and also saw that most are proper subsets. We want to illustrate the subset relationships in a diagram. In order to do this let \(\Phi_2\) be the class of \(\Phi\)-accessible programs,
3.2. THE EXTENDED PROGRAM CLASSES

$\Phi_{2,e}$ be the class of extended $\Phi$-accessible programs and $\Phi_3$ be the class of 3-valued $\Phi$-accessible programs. Remember the above shortcuts for the different classes of $\Phi^i$-accessible programs. Then we get

$$\Phi_2 = \Phi_{2,e} \subset \Phi_3$$

$$\Phi^i_2 = \Phi^i_{2,e} \subset \Phi^i_3$$

The above examples show that there exist $\Phi$-accessible programs which are not 3-valued $\Phi^i$-accessible. Therefore we do not get a subset relation between $\Phi_2$ and $\Phi^i_3$ in either direction. So the above diagram shows all possible subset relations which exist between the different classes.
CHAPTER 3. EXT. TO THE CLASSES OF $\Phi^*$ – AND $\Phi$–ACCESSIBLE PROGRAMS
Chapter 4

Topological Investigations of the Operators of the Well-Founded and AFP Model

4.1 The Well-Founded Model

First of all we want to recall the basic notations and definitions of the well-founded (partial) model given in the publication [15]. Let $P$ be a normal logic program. In this chapter we denote the two-valued immediate consequence operator of Definition 3.1.1 by $T_P$. We define the (3-valued) immediate consequence operator and unfounded sets.

**Definition 4.1.1 (3-valued $T_P$)** The (3-valued) immediate consequence operator $T_P : I_P \times I_P \rightarrow I_P$ is defined by

$$T_P(I) = \{ A \in I_P \mid \exists R \in \text{ground}(P) : \text{head}(R) = A, \text{pos}(R) \subseteq I^+, \text{neg}(R) \subseteq I^- \}.$$  

One can easily check that $T_P(I) = T_P((I, B_P \setminus I))$ where $I \in B_P$ is a (2-valued) interpretation of $P$.

**Definition 4.1.2 (Unfounded Sets)** Let $I$ be a partial interpretation and $A \subseteq B_P$. We call $A$ unfounded (set of $P$) with respect to $I$ if for all $p \in A$ the following condition is true

$$\forall R \in \text{ground}(P), \text{head}(R) = p : \text{pos}(R) \cap I^- \neq \emptyset \lor \text{neg}(R) \cap I^+ \neq \emptyset \lor \text{pos}(R) \cap A \neq \emptyset.$$  

One can easily show that if $A_1$ and $A_2$ are unfounded sets of $P$ with respect to the partial interpretation $I$ then $A_1 \cup A_2$ is also an unfounded set of $P$ with respect to $I$. Therefore it is meaningful to define the greatest unfounded set $U_P(I)$ of $P$ with respect to $I$ to be the union of all unfounded sets of $P$ with respect to $I$. We have that $U_P(I)$ is once again unfounded with respect to $I$.

The partial relations $\leq_k$ and $\leq_t$ in (3.1) can also be defined for partial interpretations.

**Definition 4.1.3** Let $I, J \in I_P \times I_P$ be partial interpretations. We define partial relations on $I_P \times I_P$.

$$I \leq_k J \iff I^+ \subseteq J^+, I^- \subseteq J^- \text{ (knowledge)}$$

$$I \leq_t J \iff I^+ \subseteq J^+, J^- \subseteq I^- \text{ (truth)}$$
One can show that \((I_P \times I_P, \leq_k)\) and \((I_P \times I_P, \leq_l)\) are complete lattices (and \((I_P \times I_P, \leq_k, \leq_l)\) is even a bilattice, see [10]).

**Definition 4.1.4 (Well-Founded Operator \(W_P\))** Let \(P\) be an arbitrary NLP. The well-founded operator \(W_P : I_P \times I_P \rightarrow I_P \times I_P\) is defined by \(W_P(I) = (T_P(I), U_P(I))\) for all \(I \in I_P \times I_P\).

One can show that the operator \(W_P\) is monotonic with respect to the \(\leq_k\)-relation. As \((I_P \times I_P, \leq_k)\) is a complete lattice one can get the least fixpoint \((lfp)\) of \(W_P\) by transfinite iteration of \(W_P\) starting from \(\emptyset\).

**Definition 4.1.5 (Transfinite Iteration of \(W_P\))** We define a transfinite sequence \(I_\alpha\) of partial interpretations using the operator \(W_P\) where \(\alpha\) is an ordinal number.

\[
\begin{align*}
I_0 &= \emptyset, \\
I_{\alpha+1} &= W_P(I_\alpha), \\
I_\alpha &= \bigcup_{\beta < \alpha} I_\beta, \text{ Lim}(\alpha).
\end{align*}
\]

As the above sequence is monotonically increasing Tarski’s theorem (Theorem 2.1.1) together with Proposition 2.1.2 yields that there must exist an ordinal number \(\alpha\) such that \(I_\alpha\) is a fixpoint of \(W_P\). As \(I_0 = \emptyset\) is consistent one can show that the whole sequence \((I_\beta)\) is composed of consistent interpretations, see [15, Lemma 3.4].

**Definition 4.1.6 (Well-Founded (partial) model)** Let \(\alpha\) be the closure ordinal of the above sequence \((I_\beta)\). If \(I_\alpha\) is not a total interpretation we call \(I_\alpha\) the well-founded partial model of \(P\) and if \(I_\alpha\) is a total interpretation we call \(I_\alpha\) the well-founded model of \(P\).

### 4.2 The Alternating Fixpoint Model

Next we want to give some slightly different definitions and notations for the alternating fixpoint model which reflect more exactly the aims of the following sections than the original definitions do. For more information on the original definitions and proofs of the claims see [14].

First of all we want to redefine the operator \(S_P\) given in the publication [14].

**Definition 4.2.1 (The Operator \(S_P\))** We define \(P(I)\) and \(S_P : I_P \rightarrow I_P\) by

\[
P(I) = \{R' \mid \exists R \in \text{ground}(P): \neg(R) \subseteq I, \text{head}(R') = \text{head}(R), \text{pos}(R') = \text{pos}(R), \neg(R') = \emptyset\},
\]

\[
S_P(I) = T_P^{\infty}(\emptyset) = T_{\text{P}(I)}^{\uparrow \omega}.
\]

Of course \(P(I)\) is always a positive logic program. The alternating fixpoint operator \(A_P\) is now easily defined by

**Definition 4.2.2 (The Alternating Fixpoint Operator \(A_P\))** Let \(A_P : I_P \rightarrow I_P\) be given by

\[
A_P(I) = B_P \setminus S_P(B_P \setminus S_P(I)) = B_P \setminus T_P^{\infty}(B_P \setminus T_P^{\infty}(\emptyset))(\emptyset).
\]
4.2. Top. Investigations of the Well-Founded Operator $W_P$

The operators $S_P$ and $A_P$ are both monotonic relative to the $\subseteq$-relation (which is easy to see). Once again using transfinite iteration of the operator $A_P$ starting with $\emptyset$ and using Tarski’s theorem, see Theorem 2.1.1, as well as Proposition 2.1.2 one gets the least fixpoint of the alternating fixpoint operator $A_P$.

**Definition 4.2.3 (Transfinite Iteration of the Operator $A_P$)** We define a transfinite sequence $(I_\alpha)$ of (2-valued) interpretations by

\[ I_0 = \emptyset, \]
\[ I_{\alpha+1} = A_P(I_\alpha), \]
\[ I_\alpha = \bigcup_{\beta < \alpha} I_\beta, \text{ Lim}(\alpha). \]

Now we can define the alternating fixpoint model of a normal logic program $P$.

**Definition 4.2.4 (Alternating Fixpoint Model)** Let $I_\alpha$ be the least fixpoint of the operator $A_P$ where $\alpha$ is the closure ordinal of the iteration. The alternating fixpoint partial model of $P$ is defined to be the partial interpretation $A_\infty = (S_P(I_\alpha), I_\alpha)$. If $A_\infty$ is a total interpretation we say $A_\infty$ is the alternating fixpoint total model.

4.3 Topological Investigations of the Well-Founded Operator $W_P$

We investigate the well-founded operator $W_P$ and the operator $U_P : I_P \times I_P \rightarrow I_P$ where $U_P(I)$ is the greatest unfounded set of $P$ with respect to $I$.

4.3.1 A New Representation of the Operator $U_P$

First we want to establish a new representation of the operator $U_P$. We need the following definition. Let $I \in I_P \times I_P$ be a partial interpretation. We define the set $P_U(I)$ by

\[ P_U(I) = \{ R' \mid \exists R \in \text{ground}(P) : \text{pos}(R) \cap I^- = \text{neg}(R) \cap I^+ = \emptyset, \]
\[ \text{head}(R') = \text{head}(R), \text{pos}(R') = \text{pos}(R), \text{neg}(R') = \emptyset \}. \]

Of course $P_U(I)$ is a positive logic program and Tarski’s theorem (Theorem 2.1.1) together with Proposition 2.1.2 states that $\text{lfp}(T^*_{P_U(I)}) = \text{lfp}(T^\omega_{P_U(I)}(\emptyset)) = T^\omega_{P_U(I)}(\emptyset)$. We prove

**Theorem 4.3.1 For all $I \in I_P \times I_P$ we have**

\[ U_P(I) = B_P \setminus \text{lfp}(T^*_{P_U(I)}) = B_P \setminus T^\omega_{P_U(I)}(\emptyset). \]

**Proof.** We choose $I \in I_P \times I_P$ and let $M_P = \text{lfp}(T^*_{P_U(I)})$ and $n(A) = \min\{ m < \omega \mid A \in T^m_{P_U(I)}(\emptyset) \}$ for all $A \in M_P$. We show first $M_P \subseteq U_P(I)^c$ using induction on $n(A)$. Let $A \in M_P$.

Let $n(A) = 0$. Then there exists $R^* = (A \leftarrow *) \in P_U(I)$ and the construction of $P_U(I)$ yields that there exists $R \in \text{ground}(P)$ with

\[ R = (A \leftarrow \neg q_1(r_1), \ldots, \neg q_k(r_k)) \text{ and } q_i(r_i) \notin I^+, \; 1 \leq i \leq k, \; k \in \mathbb{N}_0. \]
As \( R \) doesn’t satisfy the condition given in Definition 4.1.2 of unfounded sets we have \( A \notin U_p(I)^c \).

Now suppose we have proved the claim for all \( B \in M_{p_v} \) with \( n(B) < n(A) \). It follows that there exists \( R' \in P_U(I) \) with \( R' = (A \leftarrow p_1(s_1), \ldots, p_k(s_k), p_i(s_i) \in M_{p_v}, \text{ and } n(p_i(s_i)) < n(A) \) for \( 1 \leq i \leq k \). Applying the definition of \( P_U(I) \) there exists \( R \in \text{ground}(P) \) with

\[
R = (A \leftarrow p_1(s_1), \ldots, p_k(s_k), -q_i(r_1), \ldots, -q_j(r_l)) \text{ and } q_j(r_l) \notin I^+, p_i(s_i) \notin U_p(I) \cup I^-
\]

for \( 1 \leq j \leq l, 1 \leq i \leq k \) using the induction hypothesis. Therefore the condition in the definition of unfounded sets is not satisfied and we have \( A \notin U_p(I)^c \). Altogether we have \( M_{p_v} \subseteq U_p(I)^c \).

We prove the opposite direction. Let \( A \notin U_p(I) \). Using the definition of \( U_p(I) \) there exists \( R \in \text{ground}(P) \) with \( R = (A \leftarrow p_1(s_1), \ldots, p_k(s_k), -q_i(r_1), \ldots, -q_m(r_m), k, m \in \mathbb{N}_0 \) and \( q_j(r_l) \notin I^+ \), \( p_i(s_i) \notin U_p(I) \cup I^- \), \( 1 \leq i \leq k, 1 \leq j \leq m \). Applying the definition of \( P_U(I) \) we get a \( R' \in P_U(I) \) with \( R' = (A \leftarrow p_1(s_1), \ldots, p_k(s_k)) \). One obtains the following result

\[
\forall A \notin U_p(I) \exists R' \in P_U(I): \text{head}(R') = A \land \text{pos}(R') \subseteq U_p(I)^c.
\]

For all \( A \notin U_p(I) \) we define the sets \( K_A, K_A' \) and \( k: U_p(I)^c \rightarrow \omega + 1 \)

\[
K_A = \{ f : \alpha \rightarrow I_p | \alpha \leq \omega, f(0) = \{ A \}, \forall 0 < \beta < \alpha : I_B := f(\beta - 1) \neq \emptyset, \forall B \in I_B \exists R_B \in P_U(I): \text{head}(R_B) = B, f(\beta) = \bigcup_{B \in I_B} \text{pos}(R_B) \subseteq U_p(I)^c \}
\]

\[
K_A' = \{ f \in K_A | \forall f' \in K_A, f' : \beta \rightarrow I_p, \beta \geq \alpha : (f'|_{\alpha} = f \Rightarrow \alpha = \beta) \}
\]

\[
k(A) = \min \{ \alpha \leq \omega | \exists f \in K_A', f : \alpha \rightarrow I_p \}
\]

So \( K_A' \) contains maximum chains, particularly such chains which end with \( \emptyset \). Next we define \( N := \left( \bigcup_{f \in K_A'} \left( \bigcup \text{Im} f \right) \right) \setminus M_{p_v} \subseteq U_p(I)^c \). We distinguish two different cases:

\( N \neq \emptyset \): As \( A \in \bigcup_{f \in K_A'} \left( \bigcup \text{Im} f \right) \subseteq M_{p_v} \), we are already finished.

\( N \neq \emptyset \): We make the assumption that \( k(A) = \omega \). For all \( f \in K_A \) we have \( f : \omega \rightarrow I_p \) with \( \bigcup \text{Im} f \subseteq U_p(I)^c \). We show that \( N \cup U_p(I) \) is unfounded with respect to \( I \) (see Definition 4.1.2). Let \( B \in N \). Then there exists \( f \in K_A \), \( f : \omega \rightarrow I_p \), and there exists \( k < \omega \) with \( B \in f(k), B \notin M_{p_v} \cup U_p(I) \). We choose an arbitrary \( R \in \text{ground}(P) \) with \( R = (B \leftarrow p_1(s_1), \ldots, p_n(s_n), -q_i(r_1), \ldots, -q_m(r_m)) \) and \( n, m \geq 0 \). If one of the body literals is false with respect to \( I \) or if there exists \( 1 \leq i \leq n \) with \( p_i(s_i) \in U_p(I) \) the condition in the Definition 4.1.2 of unfounded sets is true. Otherwise there exists \( R' \in P_U(I) \) with \( R' = (B \leftarrow p_1(s_1), \ldots, p_n(s_n)) \) and \( \text{pos}(R') \subseteq U_p(I)^c \). It is clear that \( n > 0 \) must be true because otherwise we would have \( B \in M_{p_v} \) in contradiction to \( B \in N \). Likewise there must exist \( 1 \leq i \leq n \) with \( p_i(s_i) \notin M_{p_v} \), otherwise we could apply the definition of \( M_{p_v} \) and get \( B \in M_{p_v} \). We define \( F = f(k) \setminus \{ B \} \). The construction of \( f \in K_A' \) yields

\[
\forall C \in F \exists R_C \in P_U(I): \text{head}(R_C) = C, \text{pos}(R_C) \subseteq U_p(I)^c.
\]

Let \( f' : k + 1 \rightarrow I_p \) with \( f'|_k = f \) and \( f'(k) = \text{pos}(R') \cup \bigcup_{C \in F} \text{pos}(R_C) \). Then we have \( f' \in K_A \) and there exists \( f'' \in K_A' \) with \( f''|_{k + 1} = f' \). We get

\[
p_i(s_i) \in \text{pos}(R') \subseteq f'(k) \subseteq \bigcup_{C \in F} \text{Im} f'' \subseteq \bigcup_{f \in K_A'} \left( \bigcup \text{Im} f \right).
\]
4.3. Top. Investigations of the Well-Founded Operator $W_{\mathbf{P}}$

Altogether we have $p_i(s_i) \in N \cup U_{\mathbf{P}}(I)$ and therefore the condition in the Definition 4.1.2 of unfounded set with respect to I is true for $N \cup U_{\mathbf{P}}(I)$. So $N \cup U_{\mathbf{P}}(I)$ is unfounded with respect to I and we conclude $N \subseteq U_{\mathbf{P}}(I)$ in contradiction to the definition of N. That means, our assumption is false and we get $k(A) < \omega$ and there exists $f \in K_A'$ with $f: k(A) \rightarrow I_{\mathbf{P}}$.

Let $k = k(A) \geq 2$. The construction of $K_A'$ yields that $f(k-1) = \emptyset$ because otherwise there would exist a larger maximum chain $f'$ with $f'_{\mid k} = f$.

We claim $A \in M_{\mathbf{P}_{\omega}}$. We prove our claim by induction over $k \geq 2$.

Let $k = 2$. There exists $f: \{2\} \rightarrow I_{\mathbf{P}}$, $f(1) = \emptyset$, $f(0) = \{A\}$. It follows that there exists $R_A \in \mathbf{P}_U(I)$ with $R_A = (A \leftarrow \emptyset)$ and we have $A \in M_{\mathbf{P}_{\omega}}$.

Now let $k > 2$. We assume that our claim is already proved for all $B \in U_{\mathbf{P}}(I)^{\omega}$ with $k(B) < k$. Let $f \in K_A'$ with $f: k \rightarrow I_{\mathbf{P}}$, $f(0) = \{A\}$ and $f(k-1) = \emptyset$. The construction of $K_A$ yields that there exists $R_A \in \mathbf{P}_U(I)$ with $R_A = (A \leftarrow p_1(s_1), \ldots, p_n(s_n))$ and $\text{pos}(R_A) = f(1) \subseteq U_{\mathbf{P}}(I)^{\omega}$. We set $B_i = p_i(s_i)$ for $1 \leq i \leq n$ and get

\[
(4.1) \quad \forall \, 1 \leq i \leq n \exists f_i \in K_{B_i}' \quad f_i: k(B_i) \rightarrow I_{\mathbf{P}} \quad f_i(0) = \{B_i\}, \quad f_i(m) \subseteq f(m+1), \quad m < k - 1.
\]

Thereby one chooses in step m for $C \subseteq f_i(m)$ the clause $R_C \in \mathbf{P}_U(I)$ which is used in step $m+1$ in the construction of $f \in K_A'$ where $C \subseteq f(m+1)$. Particularly we have $k(B_i) < k$ because of (4.1) and by the induction hypothesis we get $B_i \in M_{\mathbf{P}_{\omega}}$ for all $1 \leq i \leq n$. The construction of $M_{\mathbf{P}_{\omega}}$ yields $A \in M_{\mathbf{P}_{\omega}}$ and that is what we claimed.

Altogether we have $U_{\mathbf{P}}(I)^{\omega} \subseteq M_{\mathbf{P}_{\omega}}$ and combined with the first part of the proof we get $U_{\mathbf{P}}(I) = B_{\mathbf{P}} \setminus M_{\mathbf{P}_{\omega}}$.

Now we are in a position to do some topological investigations concerning the well-founded operator $U_{\mathbf{P}}: (I_{\mathbf{P}} \times I_{\mathbf{P}}, Q_2) \rightarrow (I_{\mathbf{P}}, Q)$. We will apply the positive and negative atomic topologies $Q^+$ and $Q^-$ on $I_{\mathbf{P}}$.

4.3.2 Topological Investigations of the Operator $U_{\mathbf{P}}$

Lemma 4.3.2 (‘Semi’-Continuity of $U_{\mathbf{P}}$) Let $(I_n)_{n \in \mathbb{N}} \subseteq I_{\mathbf{P}} \times I_{\mathbf{P}}$ be a sequence of partial interpretations and $I \in I_{\mathbf{P}} \times I_{\mathbf{P}}$. Then we have

\[
I_n^+ \rightarrow I^+, \quad I_n^- \rightarrow I^- \quad \text{in} \quad Q^- \quad \Rightarrow \quad U_{\mathbf{P}}(I_n) \rightarrow U_{\mathbf{P}}(I) \quad \text{in} \quad Q^-.
\]

Proof. Let $\mathbf{P}_U(I)$ be defined as in subsection 4.3.1. Let $A \in U_{\mathbf{P}}(I)^{\omega}$. Using Theorem 4.3.1 this is equivalent to $A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))$. Then we prove by induction over $n \in \mathbb{N}$

\[
\forall \, n < \omega: \quad \left( A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))^{\omega} \right) \quad \Rightarrow \quad \exists N \in \mathbb{N}: \quad A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))^{\omega} \quad \forall \, m \geq N \right).
\]

Let $n = 1$. We have $A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))^{\omega}$ which means $R' = (A \leftarrow \emptyset) \in \mathbf{P}_U(I)$ and there exists $R \in \text{ground}(\mathbf{P})$ with $R = (A \leftarrow \neg q_1(r_1), \ldots, q_k(r_k))$, $k \in \mathbb{N}_0$ and $\text{neg}(R) \cap I^+ = \emptyset$. Using the premise of our claim there exists $K(A) \in \mathbb{N}$ with $\text{neg}(R) \cap I^+ = \emptyset$ for all $m \geq K(A)$. Therefore we have $R' \in \mathbf{P}_U(I_m)$ for all $m \geq K(A)$ which means $A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I_m))^{\omega}$ for all $m \geq K(A)$.

Now let $n > 1$. We assume that we have proved our intermediate claim for all $i < n$. Let $A \in T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))^{\omega}$, $n$ minimal. There exists $R' \in \mathbf{P}_U(I)$ with $R' = (A \leftarrow p_1(s_1), \ldots, p_k(s_k))$ and $\text{pos}(R') \subseteq T_{\mathbf{P}_{\omega}}(\mathbf{P}_U(I))^{\omega}$. Using the definition of $\mathbf{P}_U(I)$ there exists $R \in \text{ground}(\mathbf{P})$ with $R = (A \leftarrow p_1(s_1), \ldots, p_k(s_k))$. Using the definition of $\mathbf{P}_U(I)$ there exists $R \in \text{ground}(\mathbf{P})$ with $R = (A \leftarrow p_1(s_1), \ldots, p_k(s_k), \neg q_1(r_1), \ldots, \neg q_m(r_m))$ and $\text{neg}(R) \cap I^+ = \text{pos}(R) \cap I^- = \emptyset$. 

Applying the premise of our claim there exists \( K(A) \in \mathbb{N} \) with \( \neg \text{eg}(R) \cap I_m^+ = \text{eg}(R) \cap I_m^- = \emptyset \) for all \( m \geq K(A) \) which results in \( R' \in \mathcal{P}_U(I_m) \) for all \( m \geq K(A) \). Using the induction hypothesis there exists \( N \in \mathbb{N} \) with \( \text{eg}(R') \subseteq \mathcal{T}^{n-1}_U(I_m)(\emptyset) \) for all \( m \geq N \). Combining these conclusions we get \( A \in \mathcal{T}^n_U(I_m)(\emptyset) \) for all \( m \geq \max\{K(A), N\} =: N(A) \) which proves our intermediate claim.

Therefore we have \( A \in \mathcal{U}_P(I_m)^c \) for all \( m \geq N(A) \) which completes our proof. ■

**Remark 4.3.3** Under the premise of Lemma 4.3.2 and the restriction \( I = (\emptyset, \emptyset) \) we get the stronger result \( \mathcal{U}_P(I_n) \rightarrow \mathcal{U}_P(I) \) in \( Q \) because \( \mathcal{U}_P(I) \subseteq \mathcal{U}_P(I_n) \) for all \( n \) which follows from the monotonicity of the operator \( \mathcal{U}_P \).

**Counterexample 4.3.4** We show that \( \mathcal{U}_P(I_n) \rightarrow \mathcal{U}_P(I) \) in \( Q^+ \) is not true even under the stronger premise \( I_n \rightarrow I \) in \( Q_2 \). Let \( P \) be the following normal logic program without local variables

\[
\begin{align*}
p(x) & \leftarrow p(s(x)) \\
p(x) & \leftarrow \neg q(x).
\end{align*}
\]

Let \( I = (\{q(s^n(0)) \mid n \in \mathbb{N}_0\}, \emptyset) \). Then we have \( \mathcal{P}_U(I) = p(s^n(0)) \leftarrow p(s^{n+1}(0)) \mid n \in \mathbb{N}_0 \) and it follows \( M_{\mathcal{P}_U} = \text{lf}(\mathcal{T}_U(I)) = \mathcal{T}_U^\infty(I)(\emptyset) = \emptyset \). Let \( A = p(0) \) and therefore \( A \notin M_{\mathcal{P}_U} \), so \( A \notin \mathcal{U}_P(I) \). We define a sequence of partial interpretations \( I_n = \{q(s^k(0)) \mid 0 \leq k < n\}, \emptyset \) for all \( n \in \mathbb{N} \). Clearly we have \( I_n \rightarrow I \) in \( Q_2 \) and the premise of Lemma 4.3.2 is satisfied. We show that \( \mathcal{U}_P(I_n) \rightarrow \mathcal{U}_P(I) \) in \( Q^+ \) is false. We have \( \mathcal{P}_U(I_n) = \mathcal{P}_U(I) \cup \{p(s^{n+k}(0)) \leftarrow \mid k \in \mathbb{N}_0\} \) and we get

\[
A = p(0) \in \mathcal{T}_U^{n+1}(I_n)(\emptyset) \subseteq \mathcal{T}_U^\infty(I_n)(\emptyset) \text{ for all } n.
\]

Theorem 4.3.1 now yields \( A \notin \mathcal{U}_P(I_n) \) for all \( n \in \mathbb{N} \) which means that \( \mathcal{U}_P(I_n) \rightarrow \mathcal{U}_P(I) \) in \( Q^+ \) and therefore in \( Q \) is false (even under the stronger premise \( I_n \rightarrow I \) in \( Q \)).

### 4.3.3 Topological Investigations of the Well-Founded Operator \( \mathcal{W}_P \)

Let \( \mathcal{W}_P \) be the well-founded operator of Definition 4.1.4, that is, \( \mathcal{W}_P(I) = (\mathcal{T}_P(I), \mathcal{U}_P(I)) \) for all \( I \in \mathcal{I}_P \times \mathcal{I}_P \).

In this subsection we restrict the class of programs of consideration to the class of normal logic programs with no local variables, that means, for every clause \( R \in \mathcal{P} \) the free variables of the body of \( R \) are also variables of the head of \( R \).

Conditions were given in [38, Theorem 7, Corollary 6] under which the two-valued immediate consequence operator \( \mathcal{T}_P : \mathcal{I}_P \rightarrow \mathcal{I}_P \) is topologically continuous for such programs concerning the atomic topology \( Q \) defined on \( \mathcal{I}_P \). One can prove the same for the operator \( \mathcal{T}_P \) which we omit at this point.

Because of Corollary 2.4.3 and Counterexample 4.3.4 we cannot prove the continuity of \( \mathcal{W}_P \) in \( Q_2 \). The only thing we can state is the following Corollary to Lemma 4.3.2. There we make use of the topology \( Q_2' \) of Definition 2.4.2.

**Corollary 4.3.5** Let \( (I_n)_{n \in \mathbb{N}} \) be a sequence in \( \mathcal{I}_P \times \mathcal{I}_P \) and \( I \in \mathcal{I}_P \times \mathcal{I}_P \). Then we get

\[
I_n \rightarrow I \text{ in } Q_2 \implies (\mathcal{T}_P(I_n), \mathcal{U}_P(I_n)) \rightarrow (\mathcal{T}_P(I), \mathcal{U}_P(I)) \text{ in } Q_2'.
\]

□
4.4 Topological Investigations of the Operators of the Alternating Fixpoint Model

First we want to investigate the operator \( \text{Sp}_P : I_P \rightarrow I_P \) (see Definition 4.2.1) and then the alternating fixpoint operator \( \text{Ap}_P : I_P \rightarrow I_P \) (see Definition 4.2.2). Let \( P \) be once again a normal logic program with no local variables.

4.4.1 Topological Investigations of the Operator \( \text{Sp}_P \)

We can prove for \( \text{Sp}_P \) a similar result as we have proved for the operator \( \text{Up}_P \).

**Lemma 4.4.1 ('Semi'-Continuity of \( \text{Sp}_P \))** Let \( (I_n)_{n \in \mathbb{N}} \subseteq I_P \) be a sequence of (2-valued) interpretations of \( P \) and \( I \in I_P \). Then we have

\[
I_n \rightarrow I \text{ in } Q^+ \implies \text{Sp}(I_n) \rightarrow \text{Sp}(I) \text{ in } Q^+.
\]

**Proof.** Let \( I_n \rightarrow I \) in \( Q^+ \) and \( A \in \text{Sp}(I) = T^\infty_{P[I]}(\emptyset) \). We prove by induction over \( n \in \mathbb{N} \)

\[
\forall n < \omega : \left( A \in T^n_{P[I]}(\emptyset), \text{ n minimal } \implies \exists N \in \mathbb{N} : A \in T^n_{P[I_m]}(\emptyset) \forall m \geq N \right).
\]

Let \( n = 1 \). We have \( A \in T^1_{P[I]}(\emptyset) \) and there exists \( R \in \text{ground}(P) \) with \( R = (A \leftarrow \neg q_1(r_1), \ldots, \neg q_k(r_k)), k \in \mathbb{N}_0 \) and \( \text{neg}(R) \subseteq I \). Using the convergence property of \( Q^+ \) there exists \( N \in \mathbb{N} \) with \( \text{neg}(R) \subseteq I_m \) for all \( m \geq N \). Therefore we have \( A \in T^1_{P[I_m]}(\emptyset) \) for all \( m \geq N \).

Now let \( n > 1 \). We assume that we have proved our intermediate claim for all \( i < n \). Let \( A \in T^n_{P[I]}(\emptyset), \text{ n minimal} \). There exists \( R' \in P(I) \) with \( R' = (A \leftarrow p_1(s_1), \ldots, p_k(s_k)) \) and \( \text{pos}(R') \subseteq T^{n-1}_{P[I]}(\emptyset) \). Using the definition of \( P(I) \) there exists \( R \in \text{ground}(P) \) with \( R = (A \leftarrow p_1(s_1), \ldots, p_k(s_k), \neg q_1(r_1), \ldots, \neg q_m(r_m)) \) with \( \text{neg}(R) \subseteq I \). Applying the convergence property of \( Q^+ \) there exists \( N \in \mathbb{N} \) with \( \text{neg}(R) \subseteq I_m \) for all \( m \geq N \) which results in \( R' \in P(I_m) \) for all \( m \geq N \). Using the induction hypothesis for every \( B \in \text{pos}(R') \) there exists \( N' \in \mathbb{N} \) with \( \text{pos}(R') \subseteq T^{n-1}_{P[I_m]}(\emptyset) \) for all \( m \geq N' \). Combining these conclusions we get \( A \in T^n_{P[I_m]}(\emptyset) \) for all \( m \geq \max\{N, N'\} \) \& \( = N(A) \) which proves our intermediate claim.

Therefore we have \( A \in \text{Sp}(I_m) \) for all \( m \geq N(A) \) which completes our proof.

**Similarly, under the premise of Lemma 4.4.1 we cannot prove that \( \text{Sp}(I_n) \rightarrow \text{Sp}(I) \) in \( Q \) is true.**

**Counterexample 4.4.2** Let \( P \) be the following normal logic program without local variables

\[
\begin{align*}
p(x) & \leftarrow p(s(x)) \quad p(x) \leftarrow \neg q(x).
\end{align*}
\]

Let \( A = p(0) \) and \( I = \emptyset \). Then we have \( P(I) = \{p(s^n(0)) \leftarrow p(s^{n+1}(0)) | n \in \mathbb{N}_0\} \) and \( T^\infty_{P[I]}(\emptyset) = \emptyset \). So it follows \( \text{Sp}(0) \notin \text{Sp}(I) = T^\infty_{P[I]}(\emptyset) \). We define a sequence of (2-valued) interpretations by \( I_n = \{q(s^n(0))\}, n \in \mathbb{N}, \) and get \( I_n \rightarrow I \) in \( Q \). Furthermore we have \( P(I_n) = P(I) \cup \{p(s^n(0)) \leftarrow \} \) and therefore

\[
A = p(0) \in T^\infty_{P[I_m]}(\emptyset) \subseteq T^\infty_{P[I_n]}(\emptyset) = \text{Sp}(I_n) \forall n \in \mathbb{N}.
\]

Altogether we have \( I_n \rightarrow I \) in \( Q \), \( A \notin \text{Sp}(I) \) but \( A \in \text{Sp}(I_n) \) for all \( n \in \mathbb{N} \). So \( \text{Sp}(I_n) \rightarrow \text{Sp}(I) \) in \( Q \) is false.
4.4.2 Topological Investigations of the AFP Operator $A_P$

We investigate the continuity of the operator $A_P : \mathcal{I}_P \rightarrow \mathcal{I}_P$ in the topologies $Q^+$ and $Q$. We will see that the operator $A_P$ is neither continuous in $Q^+$ nor in $Q$ in common even under the premise $I_n \rightarrow I$ in $Q$ for a sequence of (2-valued) interpretations.

**Counterexample 4.4.3** Let $P$ be the following normal logic program with no local variables

$$q(x) \leftarrow \neg q(s(x))$$
$$q(x) \leftarrow q(s(x)), \neg p(x).$$

Let $I = \mathcal{B}_P$ and $I_n = \{p(s^k(0)) \mid k \in \mathbb{N}_0, k \neq n\} \cup \{q(s^k(0)) \mid k \in \mathbb{N}_0, k \neq n\}$. Then we have $I_n \rightarrow I$ in $Q$ and so in $Q^+$, too. Let

$$K := T^\infty_{P[I]}(\emptyset) = \{q(s^k(0)) \mid k \in \mathbb{N}_0\},$$
$$K_n := T^\infty_{P[I_n]}(\emptyset) = \{q(s^k(0)) \mid k \in \mathbb{N}_0, k \neq n\}$$
and we obtain

$$\mathcal{B}_P \setminus K = \{p(s^k(0)) \mid k \in \mathbb{N}_0\},$$
$$\mathcal{B}_P \setminus K_n = \{q(s^n(0))\} \cup \{p(s^k(0)) \mid k \in \mathbb{N}_0\}.$$ 

Now we have $q(0) \notin T^\infty_{P[\mathcal{B}_P \setminus K]}(\emptyset) = \emptyset$ but $q(0) \in T^\infty_{P[\mathcal{B}_P \setminus K_n]}(\emptyset) = \{q(s^k(0)) \mid k < n\}$ for all $n \in \mathbb{N}$. We conclude $q(0) \in A_P(I)$ but $q(0) \notin A_P(I_n)$ for all $n \in \mathbb{N}$. Therefore $A_P(I_n) \rightarrow A_P(I)$ is neither true in $Q^+$ nor in $Q$ and we have particularly $A_P(I) = \mathcal{B}_P$.

That the claim $(I_n \rightarrow I$ in $Q$ $\rightarrow A_P(I_n)^c \rightarrow A_P(I)^c$ in $Q^+$) is not true is shown by the following counterexample.

**Counterexample 4.4.4** Let $P$ be the following normal logic program with no local variables

$$p(x) \leftarrow p(s(x))$$
$$p(x) \leftarrow \neg q(x)$$
$$q(s(x)) \leftarrow \neg p(x)$$
$$q(s(x)) \leftarrow q(x).$$

Let $I = \emptyset$ and $I_n = \{q(s^n(0))\}$. Then we have $I_n \rightarrow I$ in $Q$ and $K := T^\infty_{P[I]}(\emptyset) = \emptyset$ as well as $K_n := T^\infty_{P[I_n]}(\emptyset) = \{p(s^k(0)) \mid k \leq n\}$. Therefore we have

$$T^\infty_{P[\mathcal{B}_P]}(\emptyset) = \{p(s^k(0)) \mid k \in \mathbb{N}_0\} \cup \{q(s^{k+1}(0)) \mid n \in \mathbb{N}_0\} = \mathcal{B}_P \setminus \{q(0)\},$$
$$T^\infty_{P[\mathcal{B}_P \setminus K_n]}(\emptyset) = \{p(s^k(0)) \mid k \in \mathbb{N}_0\} \cup \{q(s^{k+1}(0)) \mid k > n\}.$$ 

We obtain $q(s(0)) \notin A_P(I) = \{q(0)\}, q(s(0)) \in A_P(I_n) = \{q(s^k(0)) \mid k \leq n + 1\}$ for all $n \in \mathbb{N}$. Altogether we have shown that neither $A_P(I_n) \rightarrow A_P(I)$ in $Q$ is true nor that $A_P(I_n)^c \rightarrow A_P(I)^c$ in $Q^+$ is true.

4.5 Quasi-Metrics and the Well-Founded and AFP Model

In this section let $P$ be an arbitrary NLP. We transfer some definitions and results from [39] to the well-founded operator $W_P$ and the AFP operators $S_P$ and $A_P$.

First we define discrete quasi ultrametrics on $\mathcal{I}_P \times \mathcal{I}_P$ and $\mathcal{I}_P$ by
4.5. QUASI-METRICS AND THE WELL-FOUNDED AND AFP MODEL

**Definition 4.5.1** Let \( I, J \in I_P \times I_P \) be partial interpretations and \( I', J' \in I_P \) be (2-valued) interpretations. The discrete quasi ultrametrics \( d_k \) and \( d_l \) on \( I_P \times I_P \) and \( d_+ \) on \( I_P \) are defined to be

\[
d_k(I, J) = \begin{cases} 
0 & \text{if } I \leq_k J, \\
1 & \text{otherwise}, 
\end{cases}
\]

\[
d_l(I, J) = \begin{cases} 
0 & \text{if } I \leq_l J, \\
1 & \text{otherwise}, 
\end{cases}
\]

\[
d_+(I', J') = \begin{cases} 
0 & \text{if } I' \subseteq J', \\
1 & \text{otherwise}. 
\end{cases}
\]

The following lemma is a reformulation of [39, Proposition 1].

**Lemma 4.5.1 (Forward Cauchy Characterization)** Let \( (I_n)_{n \in \mathbb{N}} \) be a sequence in the quasi ultrametric space \( M = (I_P \times I_P, d_k) \). We get

\[(I_n)_{n \in \mathbb{N}} \text{ forward Cauchy sequence in } M \iff \exists m \in \mathbb{N}: I_n^+ \subseteq I_{n+1}^+ \text{ and } I_n^- \subseteq I_{n+1}^- \quad \forall n \geq m \]

\[(I_n)_{n \in \mathbb{N}} \text{ forward Cauchy sequence in } M \iff \exists m \in \mathbb{N}: I_n^+ \subseteq I_{n+1}^+ \text{ and } I_n^- \subseteq I_{n+1}^- \quad \forall n \geq m \]

**Proof.** We prove only the first statement. The second statement can be proved similarly. Let \( (I_n) \) be forward Cauchy with respect to \( d_k \). Then there exists \( m \in \mathbb{N} \) with \( d_k(I_n, I_{n+1}) = 0 \) for all \( n \geq m \). We have \( d_k(I_n, I_{n+1}) = 0 \) if \( I_n \leq_k I_{n+1} \) and \( I_n^+ \subseteq I_{n+1}^+ \) and \( I_n^- \subseteq I_{n+1}^- \). The claim follows from that.

As well we can transfer [39, Proposition 2] to our case

**Lemma 4.5.2 (Limit, Complete)** Let \( (I_n)_{n \in \mathbb{N}} \) be a forward Cauchy sequence in \( I_P \times I_P \) with respect to \( d_k \) and \( m \in \mathbb{N} \) as in Lemma 4.5.1. Then the limit \( \lim_{n \to \infty} I_n \) exists and is equal to \( (\bigcup_{n \geq m} I_n^+, \bigcap_{n \geq m} I_n^-) \) (or \( (\bigcup_{n \geq m} I_n^+, \bigcap_{n \geq m} I_n^-) \)). Particularly \( (I_P \times I_P, d_k) \) and \( (I_P \times I_P, d_l) \) are complete quasi ultrametric spaces.

**Proof.** Let \( (I_n) \) be a forward Cauchy sequence with respect to \( d_k \) and let \( I = (\bigcup_{n \geq m} I_n^+, \bigcap_{n \geq m} I_n^-) \). Let \( J \in I_P \times I_P \). If \( I \leq_k J \), then we have \( I_n \leq_k I \leq_k J \) for all \( n \geq m \), that is, \( d_k(I, J) = \lim_{n \to \infty} d_k(I_n, J) = 0 \). Otherwise \( I \leq_k J \) is not true and there exists \( m' \geq m \) such that \( I_{m'} \leq_k J \) is false. Therefore we have that \( I_n \leq_k J \) is false for all \( n \geq m' \), that is, \( d_k(I_n, J) = 1 = d_k(I, J) \) for all \( n \geq m' \). In both cases \( d_k(I, J) = \lim_{n \to \infty} d_k(I_n, J) \) is satisfied.

Next we transfer [39, Proposition 4, 5 and 6] to our case

**Lemma 4.5.3 (Forward Cauchy and the Operator W_P)** Let \( (I_n)_{n \in \mathbb{N}} \) be a forward Cauchy sequence in \( (I_P \times I_P, d_k) \). Then \( (W_P(I_n))_{n \in \mathbb{N}} \) is a forward Cauchy sequence in \( (I_P \times I_P, d_k) \).
Proof. Let \((I_n)\) be such a forward Cauchy sequence. Then there exists \(m \in \mathbb{N}\) with \(I_n \leq_k I_{n+1}\) for all \(n \geq m\). As \(W_P\) is monotonic with respect to \(d_k\) we obtain \(W_P(I_n) \leq_k W_P(I_{n+1})\) for all \(n \geq m\), that is, \((W_P(I_n))\) is a forward Cauchy sequence with respect to \(d_k\).  

Lemma 4.5.4 (Continuity) The operator \(T_P : (I_P \times I_P, d_k) \rightarrow (I_P, d_+^+)\) is Continuous. The operator \(U_P : (I_P \times I_P, d_k) \rightarrow (I_P, d_+)\) and the well-founded operator \(W_P : (I_P \times I_P, d_k) \rightarrow (I_P \times I_P, d_k)\) are not Continuous.

Proof. Let \((I_n)\) be a forward Cauchy sequence in \((I_P \times I_P, d_k)\). Then by means of Lemma 4.5.2 we have \(\text{Lim} I_n = I = (\bigcup_{n \geq m} I_n^+, \bigcup_{n \geq m} I_n^-)\) for \(m \in \mathbb{N}\) as defined in Lemma 4.5.1. We conclude \(I_n \leq_k I_{n+1}\) and therefore \(T_P(I_n) \subseteq T_P(I_{n+1})\) for all \(n \geq m\). So we get \(\text{Lim} T_P(I_n) = \bigcup_{n \geq m} T_P(I_n)\). Because of the monotonicity of \(T_P\) with respect to \(\leq_k\) we obtain \(\bigcup_{n \geq m} T_P(I_n) \subseteq T_P(I)\). Now let \(A \in T_P(I)\). Then there exists \(R \in \text{ground}(P)\) with \(\text{head}(R) = A\), \(\text{pos}(R) \subseteq I^+\) and \(\text{neg}(R) \subseteq I^-\). So there exists \(n \geq m\) with \(\text{pos}(R) \subseteq I_n^+\) and \(\text{neg}(R) \subseteq I_n^-\) which means that \(A \in T_P(I_n)\). Altogether we proved \(\text{Lim} T_P(I_n) = \bigcup_{n \geq m} T_P(I_n) = T_P(I)\), that is, \(T_P\) is Continuous.

Because of Counterexample 4.3.4 the operators \(U_P\) and \(W_P\) are not Continuous. Consider the sequence \((I_n)\) and \(I\) constructed above. We have \(\text{Lim} I_n = I\) with respect to \(d_k\) but \(A \in U_P(I)\) and \(A \notin U_P(I_n)\) for all \(n \in \mathbb{N}\). Therefore \(\text{Lim} U_P(I_n) = U_P(I)\) is not true. This fact is also the reason why \(W_P\) is not Continuous. In detail we have \(\text{Lim} W_P(I_n) = (\bigcup_{n \geq m} T_P(I_n), \bigcup_{n \geq m} U_P(I_n))\) and must show \(\bigcup_{n \geq m} U_P(I_n) = U_P(I)\) which is not true as we have already seen. So \(W_P\) is not Continuous.

Lemma 4.5.5 (Non-Expansiveness of the Operator \(W_P\)) The well-founded operator \(W_P : I_P \times I_P \rightarrow I_P \times I_P\) is non-expansive with respect to \(d_k\).

Proof. Let \(I, J \in I_P \times I_P\) with \(d_k(I, J) = 0\), that is, \(I \leq_k J\). The monotonicity of \(W_P\) yields \(W_P(I) \leq_k W_P(J)\), that is, \(d_k(W_P(I), W_P(J)) = 0\) and we are finished.

Because of the lack of the Continuity of the operator \(W_P\) we cannot use Rutten’s theorem to prove the existence of a fixpoint of \(W_P\).

One gets similar results for the operators \(S_P, A_P : I_P \rightarrow I_P\) of the AFP model.

Lemma 4.5.6 Let \((I_n)_{n \in \mathbb{N}}\) be a forward Cauchy sequence of (2-valued) interpretations in \((I_P, d_+)\). Then \((S_P(I_n))_{n \in \mathbb{N}}\) and \((A_P(I_n))_{n \in \mathbb{N}}\) are forward Cauchy sequences in \((I_P, d_+)\). The operators \(S_P\) and \(A_P\) are Continuous and non-expansive with respect to \(d_+\). Both have a fixpoint.

Proof. All statements are a direct consequence of the monotonicity of the operators \(S_P\) and \(A_P\) (even the Continuity of the operators). We apply Rutten’s theorem (Theorem 2.2.6) to get the existence of a fixpoint of both operators. As \(d_+(\emptyset, I) = 0\) for all \(I \in I_P\) the premises of Rutten’s theorem are satisfied.
Bibliography


Erklärung

Hiermit erkläre ich, Roland Heinze, daß ich die vorliegende Diplomarbeit in Informatik selbstständig durchgeführt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Bonn, den 19.12.2001,