Some Issues Concerning Fixed Points in Computational Logic: Quasi-Metrics, Multivalued Mappings and the Knaster-Tarski Theorem*

Pascal Hitzler† and Anthony Karel Seda

August 7, 2000

Department of Mathematics, National University of Ireland, Cork
E-mail: phitzler@ucc.ie, aks@ucc.ie

Abstract

Many questions concerning the semantics of disjunctive databases and of logic programming systems depend on the fixed points of various multivalued mappings and operators determined by the database or program. We discuss known versions, for multivalued mappings, of the Knaster-Tarski theorem and of the Banach contraction mapping theorem, and formulate a version of the classical fixed-point theorem (sometimes attributed to Kleene) which is new. All these results have applications to the semantics of disjunctive logic programs, and we will describe a class of programs to which the new theorem can be applied. We also show that a unification of the latter two theorems is possible, using quasi-metrics, which parallels the well-known unification of Rutten and Smyth in the case of conventional programming language semantics.

MSC2000 Subject Classifications: 54H25, 68N17
Keywords: Semantics of disjunctive logic programs, fixed points, multivalued mappings, Knaster-Tarski and Kleene theorems.

1 Introduction

In [GL91], Gelfond and Lifschitz defined the stable model semantics or answer set semantics of a disjunctive logic program or database Π, displaying the stable model as a fixed point of a certain multivalued mapping GL, see Section 4. Earlier, in [Prz88], Przymusinski defined the perfect model semantics for non-disjunctive programs and databases. Although not initially formulated in terms of fixed points of mappings T, the

*To appear in the Proceedings of the 14th Summer Conference on General Topology and its Applications: Special Session on Topology in Computer Science, New York, August, 1999. Topology Proceedings Volume 24, 1999 Summer Issue.

†The first named author acknowledges financial support under grant SC/98/621 from Enterprise Ireland.

stable and the perfect model can be viewed in these terms, see [BMP99, KKM93, SH97]. In [HS99a], the usual immediate consequence operator associated with a normal logic program P was extended to disjunctive programs Π, obtaining a multivalued operator T_Π. This extension is rather satisfactory in that it was further shown in [HS99a] that an interpretation is a supported model of Π, in a very natural sense, if and only if it is a fixed point of T_Π.

Thus, multivalued mappings T and their fixed points inevitably arise in connection with the semantics of disjunctive logic programs. It follows of course that fixed-point theorems must also be important in this same context. In [KM97], Khamsi and Misane established a version of the Knaster-Tarski theorem[1] for multivalued mappings, see Theorem 2.4 herein, and applied it to obtain the stable model of a class of signed disjunctive programs, see Theorem 4.4. This result employs monotonicity of T as formulated in Section 2 and a condition $(*)$ on T, also discussed in Section 2, which is automatically fulfilled in the case that T is single-valued. The condition $(*)$ allows one to carry out a standard transfinite induction argument to prove Theorem 2.4. However, as shown in Examples 4.8 and 4.9, it sometimes really is necessary to work transfinitely even in the case of non-disjunctive programs. The same sort of problem arises, incidentally, when dealing with operators in three-valued logic, see [Fit85, HS99b]. Thus, the name "Knaster-Tarski Theorem" for Theorem 2.4 is appropriate in that the iterations involved need not "cut off" at the first limit ordinal, ω. On the other hand, in [KKM93], Khamsi et al. established a version of the Banach contraction mapping theorem for multivalued mappings, see Corollary 3.6 below, and applied it to obtain the stable model of a countably stratified disjunctive program. In this case, needless to say, the iterates involved do cut off at ω.

Of course, in the context of partially ordered sets and single-valued mappings, the question of when the iterates just mentioned do cut off at ω is the question of continuity embraced by what is often referred to as *the* fixed-point theorem for complete partial orders or Kleene's theorem[1] and is the mainstay of conventional programming language semantics. Nevertheless, applications of the Banach contraction mapping theorem are made in this context, too, see [Rut96] for a discussion and references. Therefore, it has been a question of some considerable interest in the recent past to unify the order-theoretic and metric approaches to the semantics of imperative programming languages, see [Rut96, Smy87]. This work culminates, perhaps, in the quasi-metric fixed-point theorem of Rutten [Rut96], earlier formulated in terms of quasi-uniformities by Smyth [Smy87], which contains the Kleene and Banach theorems as special cases.

This paper has two main objectives, and the first of these is as follows. For computability purposes, it is desirable to eliminate where possible the need to work transfinitely in obtaining models of disjunctive programs such as the stable model. This ultimately depends on finding an appropriate form of continuity, which in itself raises

[1]The form we have in mind of the conventional Knaster-Tarski theorem is as follows: suppose that T is defined and monotonic on a complete partial order X. Then T has a least fixed point, which is also the least pre-fixed point of T, given by considering the supremum of the set of iterates $T^\alpha(\perp)$, where α ranges over the ordinals and $\perp$ denotes the bottom element of X. If, further, T is continuous, then the least fixed point of T is the supremum of the set of iterates $T^n(\perp)$, where n denotes a finite ordinal, see [SLG94]. In fact, the latter statement is an abstract form of the First Recursion theorem, and is usually attributed to Kleene.

two further questions which are interconnected. The first of these is the problem of establishing a satisfactory form of Kleene's theorem employing as weak a form of continuity as is practicable; the second is the identification of syntactic conditions on classes of programs which meet the continuity requirement and thereby make it possible to apply one's theorem. In answer to the first of these questions, we put forward Theorem 3.9 as a version of Kleene's theorem employing a natural and weak notion of continuity. In answer to the second question, and in showing that Theorem 3.9 is well-formulated, we show, in Section 4, how Theorem 3.9 may be applied to a natural and large class of programs. This class is a subclass of the programs to which Theorem 2.4 was applied in [KM97], but the extra information we gain is the knowledge that the process of iteration involved is not transfinite for these programs.

Our second main objective here is to carry out a unification in the context of multivalued mappings, see Theorem 3.4, of the Banach contraction mapping theorem established in [KKM93] and of our version of Kleene's theorem, Theorem 3.9. Quasi-metrics will be employed here also, as in the theorem of Rutten mentioned above, to obtain the unification, but the main point is, again, the identification in Definition 3.3 of an appropriate notion of continuity. The condition we propose is a rather weak continuity condition on orbits which are forward Cauchy sequences, but is strong enough to imply both the Banach theorem of [KKM93] and Theorem 3.9.

Thus, from the point of view of applications to computational logic, the broad picture for multivalued mappings closely parallels the situation for single-valued mappings: one has a version of the Knaster-Tarski theorem, Theorem 2.4, which essentially involves the transfinite. However, one also has a version of Kleene's theorem, Theorem 3.9, employing continuity (so that iterates close off at ω), and a version of the Banach contraction mapping theorem as in [KKM93], see Corollary 3.6. These two latter theorems, in turn, can be unified by a theorem employing quasi-metrics, see Theorem 3.4. We note, finally, that the full extent of the applicability of Theorem 3.9 to disjunctive programs is under investigation, and the results will be presented elsewhere.

Acknowledgement The authors wish to thank one of the editors, Reinhold Heckmann, for drawing their attention to some points in the first version of this paper which were in need of clarification. They also wish to thank an anonymous referee for alerting them to several significant points of contact between their discussion of stable models for disjunctive programs and extensions in default logic, and between the use of multivalued mappings and powerdomains. In particular, the referee brought to their attention the work of Klavins, Rounds and Zhang and the many papers of Rounds and Zhang, see for example [KRZ98, RZ98, ZR97a, ZR97b, ZR98], which approach some of the issues discussed here from a domain-theoretic point of view. These points of contact are discussed at the end of the paper in Section 5.

2 Multivalued Mappings and Monotonicity

In this section, we will establish some notation and make some preliminary observations concerning an extension of the notion of monotonicity from single-valued mappings to multivalued mappings which is suitable for the applications to computational logic that we have just discussed.

Let $T : X \to 2^X$ denote a multivalued mapping defined on a set X; we assume throughout that T is *non-empty* in that $T(x)$ is a non-empty set for all $x \in X$. A *fixed point* of T is a point $x \in X$ such that $x \in T(x)$. As usual, given a net $(x_i)_{i\in\alpha}$ in X and an element β of the directed set α, we call the subnet $(x_i)_{\beta\leq i}$ of $(x_i)_{i\in\alpha}$ a *tail* of $(x_i)_{i\in\alpha}$.

2.1 Definition Let $T : X \to 2^X$ be a multivalued mapping defined on X. An *orbit* of T is a net $(x_i)_{i\in\alpha}$, or just (x_i), in X, where α denotes an ordinal, such that $x_{i+1} \in T(x_i)$ for all $i \in \alpha$. An orbit $(x_i)_{i\in\alpha}$ of T is called an ω-*orbit* if α is the first limit ordinal, ω. An orbit $(x_i)_{i\in\alpha}$ of T will be said to be *eventually constant* if there is a tail $(x_i)_{\beta\leq i}$ of $(x_i)_{i\in\alpha}$ which is *constant* in that $x_i = x_j$ for all $i, j \in \alpha$ satisfying $\beta \leq i, j$.

If $T : X \to 2^X$ is a multivalued mapping and x is a fixed point of T, then we obtain an orbit of T which is eventually constant by setting $x = x_0 = x_1 = x_2 \ldots$. Conversely, suppose that $(x_i)_{i\in\alpha}$ is an orbit of T with the property that $x_{i+1} = x_i$ for all $i \in \alpha$ satisfying $\beta \leq i$, for some ordinal $\beta \in \alpha$. Then $x_\beta = x_{\beta+1} \in T(x_\beta)$ and we have a fixed point x_β of T. Thus, having a fixed point and having an orbit which is eventually constant are equivalent conditions on T.

2.2 Definition A multivalued mapping T defined on a partially ordered set X will be said to be *monotonic* if, for all $x, y \in X$ satisfying $x \leq y$ and for all $a \in T(x)$, there exists $b \in T(y)$ such that $a \leq b$.

For the rest of this section, $(X, \leq)$ will denote a complete partial order (cpo).

2.3 Definition An orbit $(x_i)_{i\in\alpha}$ of T is said to be *increasing* if we have $x_i \leq x_j$ for all $i, j \in \alpha$ satisfying $i \leq j$, and is said to be *eventually increasing* if some tail of the orbit is increasing. Finally, an increasing orbit $(x_i)_{i\in\alpha}$ of T is said to be *tight* if, for all limit ordinals $\beta \in \alpha$, we have $x_\beta = \sup\{x_i; i < \beta\}$.

Suppose that $(x_i)_{i\in\alpha}$ is an increasing orbit of T and that $\beta \in \alpha$ is a limit ordinal. Then $x_{\beta+1}$ is an element of $T(x_\beta)$ such that $x_i \leq x_{\beta+1}$ for all $i < \beta$, and of course $\sup\{x_i; i < \beta\} \leq x_\beta \leq x_{\beta+1}$. In particular, any increasing orbit $(x_i)_{i\in\alpha}$ which is tight (if such exists) must satisfy the following condition:
$(*)$ for any limit ordinal β, there exists x $(= x_{\beta+1}) \in T(\sup\{x_i; i < \beta\})$ such that $\sup\{x_i; i < \beta\} \leq x$.

This condition is a slight variant of a condition which was identified by Khamsi and Misane in [KM97] as a sufficient condition for the existence of fixed points of monotonic multivalued mappings. In fact, the following result was established in [KM97], except that it was formulated for decreasing orbits and infima and we have chosen to work with the dual notions instead, to maintain consistency, at least until we reach Section 4.

2.4 Theorem Let X be a complete partial order and let $T : X \to 2^X$ be a multivalued mapping which is non-empty, monotonic and satisfies $(*)$. Then T has a fixed point.

We omit details of the proof of this result except to observe that, starting with the bottom element $x_0 = \bot$ of X, the condition $(*)$ permits the construction, transfinitely, of a tight orbit (x_i) of T. Since this can be carried out for ordinals whose underlying cardinal is greater than that of X, we are forced to conclude that (x_i) is eventually constant and therefore that T has a fixed point.

Noting that $\sup\{x_i; i < \beta\} = \sup\{x_{i+1}; i < \beta\}$, one can view $(*)$ schematically as the statement "$\sup\{T(x_i); i < \beta\} \leq T(\sup\{x_i; i < \beta\})$" and it can therefore be thought of as a rather natural, weak continuity condition on T which is automatically satisfied by any monotonic single-valued mapping T on a cpo. As already noted, the question of when the orbit constructed in the previous paragraph becomes constant in ω steps is a question of continuity and will be taken up in the next section.

Theorem 2.4 was established in [KM97] in order to show the existence of (consistent) answer sets for a class of disjunctive programs called signed programs, see Section 4, a class of programs which includes examples related to the well-known Yale Shooting Domain. At the end of Section 4, we will give examples which show that it sometimes is necessary to work transfinitely in practice, a point which justifies the name "Knaster-Tarski theorem" applied to Theorem 2.4.

Thus, to summarize, monotonicity of T together with $(*)$ appears to give, for multivalued mappings, an exact analogue of the fixed-point theory for monotonic single-valued mappings due to Knaster and Tarski. Moreover, there are applications to the semantics of disjunctive programs which parallel those made in the standard, non-disjunctive case. In the next section, we take up the issue of establishing a corresponding Kleene theorem for multivalued mappings, and in Section 4 we will illustrate its use by applying it to a class of examples.

3 Quasi-Metrics and Multivalued Mappings

Our main objective in this section is to use quasi-metrics to obtain the unified approach we promised earlier to the Banach and Knaster-Tarski fixed-point theorems for multivalued mappings. We begin by recalling some basic definitions which are made in relation to quasi-metric spaces, see [Rut96, Sed97], and by making some new ones in relation to multivalued mappings defined on quasi-metric spaces.

3.1 Definition A set X together with a function $d : X \times X \to \mathbb{R} \cup \{+\infty\}$ is called a *quasi-metric space* if for all $x, y, z \in X$ the following conditions hold.

(1) $d(x, y) = d(y, x) = 0$ if and only if $x = y$.

(2) $d(x, z) \leq d(x, y) + d(y, z)$.

A quasi-metric space satisfying $d(x, y) \leq \max\{d(x, z), d(z, y)\}$ for all $x, y, z \in X$ (the strong triangle inequality) is called a quasi-*ultra*metric space.

A sequence (x_n) in X is a (*forward*) *Cauchy sequence* if, for all $\varepsilon > 0$, there exists $n_0 \in \mathbb{N}$ such that, whenever $n_0 \leq m \leq n$, we have $d(x_m, x_n) < \varepsilon$. A Cauchy sequence (x_n) *converges to* $x \in X$ (written $x_n \to x$, or $\lim x_n = x$) if, for all $y \in X$, $d(x, y) = \lim d(x_n, y)$. Finally, X is called *Cauchy sequence complete* or simply *complete* if every Cauchy sequence in X converges.

Notice that it is a standard fact that limits in quasi-metric spaces are unique.

3.2 Definition Let (X, d) be a quasi-metric space. A multivalued mapping $T : X \to 2^X$ is called a *contraction* if there exists a real number k in the interval $[0, 1)$ such that, for all $x, y \in X$ and for all $a \in T(x)$, there exists $b \in T(y)$ satisfying $d(a, b) \leq kd(x, y)$. We say that T is *non-expanding* if, for all $x, y \in X$ and for all $a \in T(x)$, there exists $b \in T(y)$ satisfying $d(a, b) \leq d(x, y)$.

These definitions are clearly extensions of well-known definitions made for single-valued mappings, and indeed collapse to them in the case that T is single-valued. An obvious and natural definition of continuity of T is the following: for every Cauchy sequence (x_n) in X with limit x and for every choice of $y_n \in T(x_n)$, we have that (y_n) is a Cauchy sequence and $\lim y_n \in T(x)$. In fact, the weaker definition following, which is implied by the one just given, suffices for our purposes and will be used throughout.

3.3 Definition Let $T : X \to 2^X$ be a multivalued mapping defined on a quasi-metric space (X, d). We say that T is *continuous* if we have $\lim x_n \in T(\lim x_n)$ for every ω-orbit (x_n) of T which is a Cauchy sequence.

Again, this definition collapses to a natural one in the case that T is single-valued. In fact, if T is single-valued, it simply states the condition that $\lim T(x_n) = \lim x_{n+1} = \lim x_n = T(\lim x_n)$ for every ω-orbit which is a Cauchy sequence.

Finally, if (X, d) is a quasi-metric space, we define the associated partial order $\leq_d$ on X by $x \leq_d y$ iff $d(x, y) = 0$.

The main result of this section is the following theorem.

3.4 Theorem Let (X, d) be a complete quasi-metric space and let $T : X \to 2^X$ denote a non-empty and continuous multivalued mapping on X. Then T has a fixed point if either of the following two conditions holds:
(a) T is a contraction.
(b) T is non-expanding and there is $x_0 \in X$ and $x_1 \in T(x_0)$ such that $d(x_0, x_1) = 0$ i.e. $x_0 \leq_d x_1$.

Proof: (a) Let $x_0 \in X$. Since $T(x_0) \neq \emptyset$, we can choose $x_1 \in T(x_0)$. Since T is a contraction, there is $x_2 \in T(x_1)$ such that $d(x_1, x_2) \leq kd(x_0, x_1)$. Applying this argument repeatedly, we obtain a sequence (x_n) such that for all $n \geq 0$ we have $x_{n+1} \in T(x_n)$ and $d(x_{n+1}, x_{n+2}) \leq kd(x_n, x_{n+1})$. Thus, (x_n) is an ω-orbit. Using the triangle inequality, we obtain

$$d(x_n, x_{n+m}) \leq \sum_{i=0}^{m-1} d(x_{n+i}, x_{n+i+1}) \leq \sum_{i=0}^{m-1} k^{n+i} d(x_0, x_1) \leq \frac{k^n}{1 \perp k} d(x_0, x_1).$$

Thus, (x_n) is a (forward) Cauchy sequence in X and therefore is an ω-orbit of T which is Cauchy. Since X is complete, (x_n) has a limit x_ω. Now, by continuity of T, we obtain $x_\omega \in T(x_\omega)$ and x_ω is a fixed point of T, as required.
(b) Let x_0 and $x_1 \in T(x_0)$ satisfy $d(x_0, x_1) = 0$. Since T is non-expanding, there is $x_2 \in T(x_1)$ with $d(x_1, x_2) \leq d(x_0, x_1) = 0$. Inductively, we obtain a sequence (x_n) such that $x_{n+1} \in T(x_n)$ and $d(x_n, x_{n+k}) \leq \sum_{i=0}^{k-1} d(x_{n+i}, x_{n+i+1}) = 0$. Hence, (x_n) is an orbit of T which is forward Cauchy and therefore has a limit x_ω. By continuity of T again, we see that x_ω is a fixed point of T. ■

3.5 Remark The proof given here of Part (a) of Theorem 3.4 is, up to the last step, exactly the same as the first half of the proof of the multivalued contraction mapping theorem established by Khamsi et al. in [KKM93], except that we are working with a quasi-metric rather than with a metric and therefore care needs to be taken that no use is made of symmetry. However, we have included the details in order to make this paper self-contained. On the other hand, the proof we give next of Corollary 3.6, which roughly corresponds to the second half of the proof given in [KKM93], is shorter and technically somewhat simpler than the proof given in [KKM93].

We show next that Theorem 3.4 includes both the theorem of Khamsi et al. just mentioned, and also a natural extension of Kleene's theorem to multivalued mappings, see Theorem 3.9. As stated earlier, this unification is in direct analogy with the single-valued case and is the main goal of this section.

3.6 Corollary Suppose that (M, d) is a complete metric space, and that T is a non-empty multivalued contraction on M with the property that the set $T(x)$ is closed for every $x \in M$. Then T has a fixed point.

Proof: We show that the condition that $T(x)$ is closed for every x together with that of T being a contraction implies that T is continuous, and the result then follows from Part (a) of Theorem 3.4.

First note that (M, d) being a complete metric space means that (M, d) is complete as a quasi-metric space, and obviously T satisfies (a). Now suppose that (x_n) is an orbit of T which is a forward Cauchy sequence and hence a Cauchy sequence; we want to show that $x_\omega \in T(x_\omega)$, where x_ω is the limit of (x_n).

Since T is a contraction, for every n there exists $y_n \in T(x_\omega)$ such that $d(x_{n+1}, y_n) \leq kd(x_n, x_\omega)$. Therefore, $d(y_n, x_\omega) \leq d(y_n, x_{n+1}) + d(x_{n+1}, x_\omega) \leq kd(x_n, x_\omega) + d(x_{n+1}, x_\omega)$. Hence, we have $y_n \to x_\omega$. But each $y_n \in T(x_\omega)$, and $T(x)$ is closed for every x. Consequently, the limit x_ω of the sequence y_n also belongs to $T(x_\omega)$. So, $x_\omega \in T(x_\omega)$, and it follows that T is continuous as required. ■

We next turn our attention to demonstrating that Theorem 3.4 contains a version of Kleene's theorem for multivalued mappings. It will be necessary to make some preliminary observations, as follows, concerning partially ordered sets and the quasi-metrics they carry.

Let $(X, \leq)$ be a partial order. Define a function $d_\leq : X \times X \to \mathbb{R}$ by

$$d_\leq(x, y) = \begin{cases} 0 & \text{if } x \leq y, \\ 1 & \text{otherwise.} \end{cases}$$

Then $(X, d_\leq)$ is a quasi-ultrametric space and $d_\leq$ is called the *discrete quasi-metric* on X, see [Rut96], or the *quasi-metric associated with* $\leq$. Note that $\leq_{d_\leq}$ and $\leq$ always coincide.

We say that a partial order $(X, \leq)$ is ω*-complete* if each increasing sequence (x_n) in X has a least upper bound or supremum $\sup(x_n)$. Thus, $(X, \leq)$ is an ω-complete partial order in the usual sense (an ω-cpo) if it is ω-complete and has a bottom element $\bot$.

The following elementary result collects together the basic facts concerning the relationship between a partially ordered set $(X, \leq)$ and its associated quasi-metric space $(X, d_{\leq})$.

3.7 Proposition Let $(X, \leq)$ be a partial order and let (X, d) denote the associated quasi-metric space. Then the following hold.
(i) A non-empty multivalued mapping $T : X \to 2^X$ is monotonic if and only if it is non-expanding.
(ii) A sequence (x_n) in X is eventually increasing in $(X, \leq)$ if and only if it is a Cauchy sequence in (X, d).
(iii) The partially ordered set $(X, \leq)$ is ω-complete if and only if (X, d) is complete as a quasi-metric space. Furthermore, in the presence of either form of completeness, the limit of any Cauchy sequence is the least upper bound of any increasing tail of the sequence.

Notice that neither Part (iii) of this result nor the next definition assumes the presence of a bottom element.

3.8 Definition Let the partial order $(X, \leq)$ be ω-complete and let $T : X \to 2^X$ be a non-empty multivalued mapping on X. We say that T is *ω-continuous* if T is monotonic and, for any ω-orbit (x_n) of T which is eventually increasing, we have $\sup(x_n) \in T(\sup(x_n))$, where the supremum is taken over any increasing tail of (x_n).

We obtain finally the following version of Kleene's theorem for multivalued mappings as an easy corollary of our main result. Some of its applications will be discussed below in Section 4.

3.9 Theorem (Second Corollary of Theorem 3.4) Let $(X, \leq)$ be an ω-complete partial order (with bottom element) and let $T : X \to 2^X$ be a non-empty, ω-continuous multivalued mapping on X. Then T has a fixed point.

Proof: Since $(X, \leq)$ is ω-complete, the associated quasi-metric space (X, d) is complete by Proposition 3.7. Furthermore, T is monotonic, since it is ω-continuous, and is therefore non-expanding by Proposition 3.7 again. On taking $x_0 = \bot$ and $x_1 \in T(x_0)$ arbitrarily, we have x_0 and x_1 satisfying $d(x_0, x_1) = 0$. The result will therefore follow from Part (b) of Theorem 3.4 as soon as we have established that T is continuous in the sense of Definition 3.3.

Let (x_n) be any ω-orbit of T which is a Cauchy sequence. Then (x_n) is eventually increasing and, by ω-continuity of T, we have $\sup(x_n) \in T(\sup(x_n))$, where the supremum is taken over any increasing tail of (x_n). In other words, we have $\lim x_n \in T(\lim x_n)$ and hence we have the continuity of T that we require. ■

3.10 Remark The Knaster-Tarski theorem for single-valued mappings T asserts that the fixed point produced by the usual proof is the least fixed point of T. This assertion does not immediately carry over to the case of multivalued mappings T without additional assumptions. One such simple, though rather strong, condition is the following: for each $x \in X$, assume that $T(x)$ has a least element M_x and that $M_x \leq M_y$ whenever

$x \leq y$. To see that this suffices, suppose that x is any fixed point of T, and construct the orbit (x_n) of T by setting $x_0 = \bot$ and $x_{n+1} = M_{x_n}$ for each n. Then (x_n) converges to a fixed point $\overline{x}$. Noting that $\bot \leq x$ and that $M_x \leq x$, we see that $x_n \leq x$ for all n. Hence, $\overline{x} \leq x$.

4 An Application of Theorem 3.9

As already mentioned, Theorem 2.4 was applied in [KM97] in order to find answer sets for a certain class of extended disjunctive programs, see Lemma 4.3 and Theorem 4.4 below. In this section, we will define a subclass of these programs to which the multivalued Kleene theorem, Theorem 3.9, can be applied instead.

We will first give some preliminary definitions and results that will be needed in presenting our own results; they can all be found in [KM97].

4.1 Definition Let `Lit` denote the set of all ground literals in a first-order language $\mathcal{L}$. Thus, `Lit` contains all ground atoms A in $\mathcal{L}$ (the positive literals) together with all negated atoms $\neg A$ (the negative literals). A *rule* r is an expression of the form

$$\forall(L_1 \vee \cdots \vee L_n \leftarrow L_{n+1} \wedge \cdots \wedge L_m \wedge \texttt{not}\ L_{m+1} \wedge \cdots \wedge \texttt{not}\ L_k)$$

where $L_i \in \texttt{Lit}$ for each i. Rules are usually written as

$$L_1, \ldots, L_n \leftarrow L_{n+1}, \ldots, L_m, \texttt{not}\ L_{m+1}, \ldots, \texttt{not}\ L_k.$$

As is customary in this subject, we utilize here both the classical negation $\neg$ mentioned earlier in defining negative literals, and the negation `not`. The latter negation, `not`, is usually interpreted to mean *negation as failure*, which is the standard way of implementing "not" in practice. Employing these two forms of negation in conjunction results in a significant increase in expressiveness. We refer the reader to [GL91] for a discussion of this point and note here that current answer set programming systems [Lif99a] usually include both kinds of negation.

Given a rule r, we set $\texttt{Head}(r) = \{L_1, \ldots, L_n\}$, $\texttt{Pos}(r) = \{L_{n+1}, \ldots, L_m\}$ and $\texttt{Neg}(r) = \{L_{m+1}, \ldots, L_k\}$. A rule r is said to be *disjunctive* if $n \geq 2$, and *non-disjunctive* otherwise. An *extended disjunctive program* is a countable set of disjunctive rules. If all the rules are non-disjunctive, the program is said to be *non-disjunctive*.

In order to describe the answer set semantics, or stable model semantics, for extended disjunctive programs, we first consider programs without negation, `not`. Thus, let Π denote a disjunctive program in which $\texttt{Neg}(r)$ is empty for each rule $r \in \Pi$. A subset X of `Lit`, i.e. $X \in 2^{\texttt{Lit}}$, is said to be *closed by rules in* Π if, for every $r \in \Pi$ such that $\texttt{Pos}(r) \subseteq X$, we have that $\texttt{Head}(r) \cap X \neq \emptyset$. The set $X \in 2^{\texttt{Lit}}$ is called an *answer set* for Π if it is a minimal set which satisfies:

1. If X contains complementary literals, then $X = \texttt{Lit}$.

2. X is closed by rules in Π.

We denote the set of answer sets of Π by $\alpha(\Pi)$. If Π is non-disjunctive, then $\alpha(\Pi)$ is a singleton set. However, if Π is disjunctive, then $\alpha(\Pi)$ may contain more than one element.

Now suppose that Π is a disjunctive program that may contain `not`. For a set $X \in 2^{\texttt{Lit}}$, consider the program Π^X defined as follows.

1. If $r \in \Pi$ is such that $\texttt{Neg}(r) \cap X$ is not empty, then we remove r i.e. $r \notin \Pi^X$.

2. If $r \in \Pi$ is such that $\texttt{Neg}(r) \cap X$ is empty, then the rule r' belongs to Π^X, where r' is defined by $\texttt{Head}(r') = \texttt{Head}(r)$, $\texttt{Pos}(r') = \texttt{Pos}(r)$ and $\texttt{Neg}(r') = \emptyset$.

It is clear that the program Π^X does not contain `not` and therefore $\alpha\left(\Pi^X\right)$ is defined. We say that X is an *answer set* or *stable model* of Π if $X \in \alpha\left(\Pi^X\right)$. So, answer sets are fixed points of the operator GL introduced by Gelfond and Lifschitz in [GL91], where $GL(X) = \alpha\left(\Pi^X\right)$. The operator GL is in general not monotonic. However, for non-disjunctive programs it is *antimonotonic* in that we have $GL(X) \supseteq GL(Y)$ whenever $X \subseteq Y$. This fact is used in order to obtain a monotonic operator by applying the operator GL twice. For this purpose, we partition a given program, if possible, into two suitable subprograms as follows.

4.2 Definition An extended disjunctive logic program Π is said to be *signed* if there exists $S \in 2^{\texttt{Lit}}$, called a *signing*, such that every rule $r \in \Pi$ satisfies one or other of the following conditions.

1. If $\texttt{Neg}(r) \cap S$ is empty, then $\texttt{Head}(r) \subseteq S$ and $\texttt{Pos}(r) \subseteq S$. Let Π_S be the subprogram of Π consisting of those rules which satisfy this condition.

2. If $\texttt{Neg}(r) \cap S$ is not empty, then $\texttt{Head}(r) \cap S = \texttt{Pos}(r) \cap S = \emptyset$ and $\texttt{Neg}(r) \subseteq S$. Let $\Pi_{\bar{S}}$ be the subprogram of Π consisting of those rules which satisfy this condition, where $\bar{S}$ denotes the set $\texttt{Lit} \setminus S$.

Clearly, the programs Π_S and $\Pi_{\bar{S}}$ are disjoint and $\Pi = \Pi_S \cup \Pi_{\bar{S}}$. A signed program Π is said to be *semi-disjunctive* if there exists a signing S such that Π_S is non-disjunctive.

It turns out that, for signed semi-disjunctive programs, the operator $T : 2^{\bar{S}} \to 2^{2^{\bar{S}}}$ defined by

$$T(X) = \alpha\left(\Pi_{\bar{S}}^{\alpha\left(\Pi_S^X\right)}\right)$$

is monotonic with respect to the ordering $\supseteq$ which is the dual of the order of subset inclusion, $\subseteq$. In fact, for the remainder of this section we will be concerned with decreasing orbits, and ω-continuity with respect to decreasing orbits etc. So, let us note that $2^{\texttt{Lit}}$ is a complete lattice with respect to $\subseteq$, and therefore the ordering $\supseteq$ on $2^{\texttt{Lit}}$ turns this set into an ω-cpo (with bottom element). Since it is natural to think of the ordering $\subseteq$ on $2^{\texttt{Lit}}$, rather than its dual, the notions and results of this section will be formulated with respect to $\subseteq$. But, in fact, we will later on apply the dual version of Theorem 3.9, where the notions of monotonicity, ω-continuity and ω-cpo will be taken to mean the duals of the corresponding notions introduced in Section 3, see for example Lemma 4.3.

The following lemma, [KM97, Lemma 2], establishes the dual of the hypothesis $(*)$ on T which was used in Theorem 2.4.

4.3 Lemma With the notation already established, let Π be a signed semi-disjunctive program, let (X_β) be a decreasing orbit of T in $2^{\bar{S}}$ and let X denote $\bigcap_\beta X_\beta$. Then there exists $Z \subseteq \bar{S}$ such that $Z \in T(X)$ and $Z \subseteq X$.

From this lemma, it follows by Theorem 2.4 that the operator T has a fixed point. The proof of the next theorem was based on this observation.

4.4 Theorem Let Π be a signed semi-disjunctive program which is safe[2] with respect to the partition $(\Pi_S, \Pi_{\bar{S}})$, where S is a signing for which Π_S is non-disjunctive. Then Π has a consistent answer set, that is, an answer set which does not contain any complementary literals.

The proof of this result utilizes only the single fact from Lemma 4.3 that a fixed point of T can be found (by applying Theorem 2.4). So, if a fixed point of T can be found by other means, the proof of Theorem 4.4, as given in [KM97], is still valid.

Now, if Π is a program as in Theorem 4.4 and, in addition to this, T is ω-continuous (using the notion dual to the one defined in Section 3), then we obtain the fixed point of T using no more than ω iterations. We will see that a finiteness condition together with an acyclicity condition suffices to achieve this.

4.5 Definition A program Π is said to be of *finite type* if, for each $L \in \mathtt{Lit}$, the set of rules in Π with L in their head is finite[3]. A program Π is called *acyclic* if there is a mapping $l : \mathtt{Lit} \to \mathbb{N}$, called a *level mapping*, such that $l(L) = l(\neg L)$ for each literal L and, for every rule r in Π and for all L in $\mathtt{Head}(r)$ and all L' in $\mathtt{Pos}(r) \cup \mathtt{Neg}(r)$, we have $l(L) > l(L')$.

The condition on a program that it is of finite type was used in [Sed95] in order to establish the continuity, in the Cantor topology, of the immediate consequence operator of a normal logic program, that is, of a non-disjunctive program. Later on it was shown in [Sed97] that continuity in the Cantor topology is closely related to continuity in quasi-metric spaces. Thus, in the light of Section 3, it is not surprising that programs of finite type make an appearance again in our present setting.

Acyclic normal logic programs were studied in the context of termination analysis, see [Bez89, Cav91]. In [SH98], the larger class of locally hierarchical programs was studied from a topological point of view. Our Definition 4.5 gives us a natural extension of these concepts to the disjunctive case.

We now inductively define the following sets for a signed semi-disjunctive program

[2] This concept is defined in [KM97], but it will not be needed here.

[3] When working with non-ground programs, a sufficient condition to obtain this for the ground instantiation of the program is the absence of local variables. See also Example 4.9.

with signing S.

$$\begin{aligned} X_0 &= \mathtt{Lit}, \\ Y_i &= \alpha\left(\Pi_S^{X_i}\right), \\ X_{i+1} &\in \alpha\left(\Pi_{\bar{S}}^{Y_i}\right) \text{ with } X_{i+1} \subseteq X_i, \\ X &= \bigcap_{i\in\mathbb{N}} X_i, \\ Y &= \bigcup_{i\in\mathbb{N}} Y_i. \end{aligned}$$

Indeed, these sets are well-defined since Π_S, and therefore $\Pi_S^{X_i}$, is non-disjunctive for each i, and since the operator T, where $T(X_i) = \alpha\left(\Pi_{\bar{S}}^{\alpha\left(\Pi_S^{X_i}\right)}\right)$ as above, is monotonic. With this notation, we have the following lemma.

4.6 Lemma Let Π be a signed semi-disjunctive program with signing S such that $\Pi_{\bar{S}}$ is of finite type. Then the following hold with respect to the ordering $\subseteq$ on $2^{\mathtt{Lit}}$.

(i) The sequence X_i is decreasing.

(ii) The sequence $\Pi_S^{X_i}$ of programs is increasing with respect to set-inclusion, and $\bigcup \Pi_S^{X_i} = \Pi_S^X$.

(iii) The sequence Y_i is increasing.

(iv) The sequence $\Pi_{\bar{S}}^{Y_i}$ of programs is decreasing with respect to set-inclusion, and $\bigcap \Pi_{\bar{S}}^{Y_i} = \Pi_{\bar{S}}^{Y}$.

(v) $Y = \alpha\left(\Pi_S^X\right)$.

(vi) X is closed by rules in $\Pi_{\bar{S}}^Y$.

(vii) For each L in X, there is a rule r in $\Pi_{\bar{S}}^Y$ with $L \in \mathtt{Head}(r)$ such that the following two conditions are satisfied.

(vii.1) $\mathtt{Pos}(r) \subseteq X$.

(vii.2) For any literal $L' \in \mathtt{Head}(r)$ with $L' \neq L$, we have $L' \notin X$.

Proof: (i) This follows immediately from the definition of the X_i.
(ii) This follows from (i), (iii) follows from (ii), and (iv) follows from (iii).
(v) If $L \in Y$, then there is $i_0 \in \mathbb{N}$ such that $L \in Y_i = \alpha\left(\Pi_S^{X_i}\right)$ for all $i \geq i_0$. Since the sequence $\Pi_S^{X_i}$ of programs is increasing with respect to set-inclusion and $\Pi_S^{X_i} \subseteq \Pi_S^X$ for each i, we obtain $L \in \alpha\left(\Pi_S^X\right)$ and therefore $Y \subseteq \alpha\left(\Pi_S^X\right)$. Now let r be a clause in Π_S^X. If $\mathtt{Pos}(r) \subseteq Y$, then there is $i \in \mathbb{N}$ such that $\mathtt{Pos}(r) \subseteq Y_i$. But each Y_i is closed by rules in $\Pi_S^{X_i}$ and $\Pi_S^{X_i}$ is non-disjunctive for each i, hence we obtain that $\mathtt{Head}(r) \in Y_i$. So $\mathtt{Head}(r) \in Y$ and it follows that Y is closed by rules in Π_S^X. Since answer sets of Π_S^X

are sets which are minimally closed by rules in $\Pi_{\bar{S}}^X$ and since $Y \subseteq \alpha\left(\Pi_{\bar{S}}^X\right)$, we obtain that $Y = \alpha\left(\Pi_{\bar{S}}^X\right)$.
(vi) This was shown in [KM97].
(vii.1) Let $L \in X$ be a literal. We know that $L \in X_n$ for all n. But X_n is minimally closed by rules in $\Pi_{\bar{S}}^{Y_n}$, therefore we also know that, for each n, there must be a rule r in $\Pi_{\bar{S}}^{Y_n}$ with $L \in \mathtt{Head}(r)$ and $\mathtt{Pos}(r) \subseteq X_n$. Since $\Pi_{\bar{S}}$ is of finite type, we also know that there are only finitely many rules r in $\Pi_{\bar{S}}^{Y_n}$ with $L \in \mathtt{Head}(r)$. But $\Pi_{\bar{S}}^{Y_i+1} \subseteq \Pi_{\bar{S}}^{Y_i}$ for all i, so it follows that there must be a rule r in $\Pi_{\bar{S}}^{Y}$ with $L \in \mathtt{Head}(r)$ such that $\mathtt{Pos}(r) \subseteq X_i$ for all i. Hence $\mathtt{Pos}(r) \subseteq X$.
(vii.2) Let $r_1, \ldots, r_n$ be all the rules in $\Pi_{\bar{S}}^Y$ with $L \in \mathtt{Head}(r_i)$ and $\mathtt{Pos}(r_i) \subseteq X$, noting that $\Pi_{\bar{S}}^Y$ is of finite type so that there exist only finitely many such rules. There must now be a $j_0 \in \mathbb{N}$ such that, for all $j \geq j_0$, we have that each r_i is a rule in $\Pi_{\bar{S}}^{Y_j}$ with $\mathtt{Pos}(r_i) \subseteq X_j$. Now, for each $i = 1, \ldots, n$, suppose that there is a literal $L_i \neq L$ in $\mathtt{Head}(r_i)$ with $L_i \in X$. Then we have $L_i \in X_j$ for all $j \geq j_0$. It is now easy to see that $X_j \setminus \{L\}$ is closed by rules in $\Pi_{\bar{S}}^{Y_j}$, which contradicts the fact that X_j is minimally closed by rules in $\Pi_{\bar{S}}^{Y_j}$. ∎

If the program $\Pi_{\bar{S}}$ additionally satisfies the acyclicity condition, then X is already a fixed point of T, as we show next.

4.7 Theorem Let Π be a signed semi-disjunctive program with signing S such that $\Pi_{\bar{S}}$ is of finite type and is acyclic. Let (X_n) be a decreasing ω-orbit of T in $2^{\bar{S}}$ and let $X = \bigcap_i X_n$. Then $X \in T(X)$.

Proof: We know from Lemma 4.3 that there is $Z \subseteq X$ with $Z \in T(X)$. Assume $Z' = X \setminus Z \neq \emptyset$. Since $\Pi_{\bar{S}}^Y$ is acyclic, there must be an $L \in Z'$ of minimal level. But $L \in X$ so, by Lemma 4.6 (vii), there must be a rule r which satisfies Conditions (vii.1) and (vii.2). By (vii.1) and minimality of the level of L, we obtain $\mathtt{Pos}(r) \subseteq Z$ and since Z is closed by rules in $\Pi_{\bar{S}}^Y$, there must be a literal $L' \in \mathtt{Head}(r)$ with $L' \in Z$. But $Z \subseteq X$, and, by (vii.2) of Lemma 4.6, we obtain $L \in Z$, which contradicts the assumption $L \in Z'$. ∎

As already mentioned above, the proof of Theorem 4.4 now carries over directly from [KM97], so that each signed semi-disjunctive program which is safe with respect to the partition $(\Pi_S, \Pi_{\bar{S}})$, where S is a signing for which Π_S is non-disjunctive and $\Pi_{\bar{S}}$ is of finite type and acyclic, has a consistent answer set. From the proof of Theorem 4.4 together with Theorem 4.7, this answer set turns out to be $Y \cup X$, with notation as defined in the paragraph preceding Lemma 4.6. The novelty of this theorem lies in the fact that the answer set can be found by applying the operator T no more than ω times.

We conclude with two examples which show that the conditions of being of finite type and acyclic are indeed necessary. We will use the notation from Lemma 4.6.

4.8 Example Let Π be the ground instantiation of the following program, where x

denotes a variable and 0 a constant.

$$\begin{aligned} p(x) &\leftarrow \texttt{not}\ q(x) \\ q(s(x)) &\leftarrow \texttt{not}\ p(x) \\ r(0) &\leftarrow q(x), \texttt{not}\ p(x) \end{aligned}$$

The program Π is signed with signing $S = \{p(s^n(0)); n \in \mathbb{N}\}$ and is trivially semi-disjunctive. Note, however, that $\Pi_{\bar{S}}$ is not of finite type. We now make the following calculations:

$$\begin{aligned} X_0 &= \texttt{Lit}, \\ Y_0 &= \emptyset, \\ X_i &= \{r(0)\} \cup \{q(s^n(0)); n \geq i\} \text{ for } i \geq 1, \\ Y_i &= \{p(s^n(0)); n = 1, \ldots, i\} \text{ for } i \geq 1. \end{aligned}$$

As expected, the set $X_\omega = \bigcap_i X_i = \{r(0)\}$ is not a fixed point of T nor is $X_\omega \cup \bigcup_i Y_i = \{r(0)\} \cup \{p(s^n(0)); n \in \mathbb{N}\}$ an answer set of Π. However, taking $X_{\omega+1} = T(X_\omega) = \emptyset$, which is a fixed point of T, we obtain $\{p(s^n(0)); n \in \mathbb{N}\}$ as answer set of Π.

The following example shows that the acyclicity condition on $\Pi_{\bar{S}}$ cannot be dropped.

4.9 Example Let Π be the ground instantiation of the following program, where x is a variable and a constant symbol 0 is added to the language underlying Π.

$$\begin{aligned} t(x) &\leftarrow t(x) \\ p(x) &\leftarrow \texttt{not}\ q(x) \\ q(s(x)) &\leftarrow \texttt{not}\ p(x) \\ r(x) &\leftarrow q(x), \texttt{not}\ p(x) \\ r(x) &\leftarrow r(s(x)), \texttt{not}\ t(x) \end{aligned}$$

The program Π is signed with respect to the signing $S = \{p(s^n(0)), t(s^n(0)); n \in \mathbb{N}\}$ and is trivially semi-disjunctive. Note, however, that this program is never acyclic relative to any level mapping. We now make the following calculations:

$$\begin{aligned} X_0 &= \texttt{Lit}, \\ Y_0 &= \emptyset, \\ X_i &= \{q(s^n(0)); n \geq i\} \cup \{r(s^n(0)); n \in \mathbb{N}\} \text{ for } i \geq 1, \\ Y_i &= \{p(s^n(0)); n = 1, \ldots, i\} \text{ for } i \geq 1. \end{aligned}$$

As expected, the set $X_\omega = \bigcap_i X_i = \{r(s^n(0)); n \in \mathbb{N}\}$ is not an answer set of $\Pi_{\bar{S}}$ nor is $X_\omega \cup \bigcup_i Y_i = \{r(s^n(0)); n \in \mathbb{N}\} \cup \{p(s^n(0)); n \in \mathbb{N}\}$ an answer set of Π. However, if we keep on iterating and calculate

$$\begin{aligned} Y_{\omega+1} &= \alpha\left(\Pi_S^{X_\omega}\right) = \{p(s^n(0)); n \in \mathbb{N}\}, \text{ and} \\ X_{\omega+1} &= T(X_\omega) = \emptyset \end{aligned}$$

we obtain $X_{\omega+1}$ as fixed point of T and $\{p(s^n(0))\}$ as answer set of Π.

5 Conclusions and Further Work

As already noted in the Introduction, we are indebted to an anonymous referee for calling our attention to several points of contact between this work, default logic and domain theory, and we close by discussing some of these and some of the lines of research which appear to emerge from these points of contact.

First, logic programs of the form considered in this paper can be viewed, in a rather simple way, as *default theories* in the sense of [Rei80]. Default theories constitute an important formalism in the area of non-monotonic reasoning, and we refer the reader to [BF91, GL91, Boc95, Lif99b] and the references contained therein for a discussion of the relationship between default logic and logic programs. From this point of view, the standard models of a disjunctive program Π, such as the stable model, correspond to extensions in default logic: truth in a model corresponds to default theorem. Furthermore, Rounds and Zhang [RZ98, ZR97a, ZR97b, ZR98] study a version of default reasoning from a domain-theoretic point of view. In particular, they focus on the Smyth powerdomain noting that this can be used to model non-monotonicity. This results, for example, in the implementation of a non-monotonic reasoning system, see [KRZ98], which bears relationship to answer set programming systems which are currently under investigation, see [Lif99a, MT99]. It should therefore be of interest to relate the syntactic conditions discussed in this paper, which provide stable models, to those used in the literature on default theories to provide extensions (and vice versa), and also to relate these to the results of Rounds and Zhang, see in particular the Introduction of [ZR97b] where it is noted that such connections are not at all obvious but need to be investigated.

In Chapter 6 of [ZR98], a treatment is given of the semantics of disjunctive logic programs (as considered here) with the same overall objective as our own. Zhang and Rounds base their treatment on the Smyth powerdomain, again. One feature of such an approach is that, by using the right domain, the concept of multivalued function is avoided and continuity can always be taken to be Scott continuity. On the other hand, the approach followed here needs less formal, abstract theory and employs quite simple syntactical conditions on programs. Once again, the interaction between these various approaches (and between other standard approaches, see [LMR92]) needs to be investigated.

Finally, we note that in their approach, Rounds and Zhang have made extensive use of Kleene's strong three-valued logic, see also [Boc95], which was introduced to logic programming in [Fit85], and is also known as the *Fitting semantics* for logic programs. The authors have investigated this in detail from a topological point of view in [HS99b, HS00a, HS00b, HS00c], again with the same intention of employing domain-theoretic-like methods in logic programming semantics. This is, therefore, another point of contact, amongst many, which should be interesting and fruitful to pursue.

References

[Bez89] M. Bezem, *Characterizing Termination of Logic Programs with Level Mappings.* In: E.L. Lusk and R.A. Overbeek (Eds.), Proceedings of the North American Conference on Logic Programming. MIT Press, Cambridge MA, 1989, pp. 69–80.

[BF91] N. Bidoit and C. Froidevaux, *Negation by Default and Unstratifiable Logic Programs,* Theoretical Computer Science **78** (1991), 85–112.

[Boc95] A. Bochman, *Default Consequence Relations as a Logical Framework for Logic Programs.* In: V.W. Marek and A. Nerode (Eds.), Proceedings of the Third International Conference on Logic Programming and Non-monotonic Reasoning (LPNMR'95), Lexington, KY, USA, June 26-28, 1995. Lecture Notes in Computer Science, Vol. 928, Springer-Verlag, 1995, pp. 245–258.

[BMP99] S. Bouamama, D. Misane and S. Priess-Crampe, *An Application of Ultrametric Spaces in Logic Programming.* Preprint, July, 1999, pp. 1–6.

[Cav91] L. Cavedon, *Acyclic Logic Programs and the Completeness of SLDNF-Resolution,* Theoretical Computer Science **86** (1991), 81–92.

[Fit85] M. Fitting, *A Kripke-Kleene Semantics for General Logic Programs,* J. Logic Programming **2** (1985), 295–312.

[GL91] M. Gelfond and V. Lifschitz, *Classical Negation in Logic Programs and Disjunctive Databases,* New Generation Computing **9** (1991), 365–385.

[HS99a] P. Hitzler and A.K. Seda, *Multivalued Mappings, Fixed-Point Theorems and Disjunctive Databases.* In: A. Butterfield and S. Flynn (Eds.), Proceedings of the Third Irish Workshop on Formal Methods (IWFM'99). Electronic Workshops in Computing (eWiC), British Computer Society, 1999, pp. 1–18.

[HS99b] P. Hitzler and A.K. Seda, *Characterizations of Classes of Programs by Three-Valued Operators.* In: M. Gelfond, N. Leone and G. Pfeifer (Eds.), Proceedings of the Fifth International Conference on Logic Programming and Non-Monotonic Reasoning (LPNMR'99), El Paso, Texas, December, 1999. Lecture Notes in Artificial Intelligence Vol. 1730, Springer-Verlag, 1999, pp. 357–371.

[HS00a] P. Hitzler and A.K. Seda, *Classes of Logic Programs which Possess Unique Supported Models.* Proceedings of Information'2000, Aizu, Japan, October 2000. Special Issue of Information, pp. 1–8, to appear.

[HS00b] P. Hitzler and A.K. Seda, *A New Fixed-point Theorem for Logic Programming Semantics.* Proceedings of the joint IIIS & IEEE meeting of the Fourth World Multiconference on Systemics, Cybernetics and Informatics (SCI2000) and the Sixth International Conference on Information Systems Analysis and Synthesis (ISAS2000), Orlando, Florida, USA, July, 2000. International Institute of Informatics and Systemics: IIIS, Vol. VII, Computer Science and Engineering Part 1, 2000, pp. 418-423.

[HS00c] P. Hitzler and A.K. Seda, *Localizing Uniquely Determined Programs.* Preprint, Department of Mathematics, University College Cork, 2000. Submitted to Proceedings of MFCSIT2000, pp. 1–15.

[KKM93] M.A. Khamsi, V. Kreinovich and D. Misane, *A New Method of Proving the Existence of Answer Sets for Disjunctive Logic Programs: A Metric Fixed Point*

Theorem for Multivalued Maps. Proceedings of the Workshop on Logic Programming with Incomplete Information, Vancouver, British-Columbia, Canada, 1993, pp. 58–73.

[KM97] M.A. Khamsi and D. Misane, *Disjunctive Signed Logic Programs*, Fundamenta Informaticae **32** (3-4) (1997/98), 349–357.

[KRZ98] E. Klavins, W.C. Rounds and G-Q. Zhang, *Experimenting with Power Default Reasoning.* Proceedings of the Fifteenth National Conference on Artificial Intelligence and the Tenth Innovative Applications of Artificial Intelligence Conference, AAAI 98, IAAI 98, July 26-30, 1998, Madison, Wisconsin, USA. AAAI Press / The MIT Press, 1998, pp. 846-852.

[Lif99a] V. Lifschitz, *Answer Set Planning.* In: D. de Schreye (Ed.), Proceedings of the Sixteenth International Conference on Logic Programming (ICLP'99), Las Cruces, New Mexico, November 1999. MIT Press, Cambridge MA, 1999, pp. 23–37.

[Lif99b] V. Lifschitz, *Success of Default Logic.* In: Logical Foundations for Cognitive Agents: Contributions in Honour of Ray Reiter. Springer Verlag, 1999, pp. 208-212.

[LMR92] J. Lobo, J. Minker and A. Rajaseker, *Foundations of Disjunctive Logic Programming.* MIT Press series in Logic Programming, Cambridge MA, 1992.

[MT99] V.M. Marek and M. Truszczyński, *Stable Models and an Alternative Logic Programming Paradigm.* In: K.R. Apt, V.W. Marek, M. Truszczyński and D.S. Warren (Eds.), The Logic Programming Paradigm: A 25-Year Perspective. Springer-Verlag, Berlin, 1999, pp. 375–398.

[Prz88] T. Przymusinski, *On the Declarative Semantics of Deductive Databases and Logic Programs.* In: Jack Minker (Ed.), Foundations of Deductive Databases and Logic Programming. Morgan Kaufmann Publishers Inc., Los Altos, 1988, pp. 193–216.

[Rei80] R. Reiter, *A Logic for Default Reasoning,* Artificial Intelligence **13** (1) (1980), 81–132.

[RZ98] W.C. Rounds and G-Q. Zhang, *Clausal Logic and Logic Programming in Algebraic Domains.* Preprint, 1998, submitted to Information and Computation, pp. 1–21.[4]

[Rut96] J.J.M.M. Rutten, *Elements of Generalized Ultrametric Domain Theory*, Theoretical Computer Science **170** (1996), 349–381.

[Sed95] A.K. Seda, *Topology and the Semantics of Logic Programs,* Fundamenta Informaticae **24** (1995), 359–386.

[Sed97] A.K. Seda, *Quasi-metrics and the Semantics of Logic Programs,* Fundamenta Informaticae **29** (2) (1997), 97–117.

[4]Available from http://www.cs.uga.edu/∼gqz/.

[SH97] A.K. Seda and P. Hitzler, *Topology and Iterates in Computational Logic.* Proceedings of the Twelfth Summer Conference on General Topology and its Applications: Special Session on Topology in Computer Science, Ontario, August 1997. Topology Proceedings 1997 (22), 427–469.

[SH98] A.K. Seda and P. Hitzler, *Strictly Level-Decreasing Logic Programs.* In: A. Butterfield and S. Flynn (Eds.), Proceedings of the Second Irish Workshop on Formal Methods (IWFM'98). Electronic Workshops in Computing (eWiC), British Computer Society, 1998, pp. 1–18.

[Smy87] M.B. Smyth, *Quasi Uniformities: Reconciling Domains with Metric Spaces.* In: M. Main, A. Melton, M. Mislove and D. Schmidt (Eds.), Mathematical Foundations of Programming Language Semantics. Lecture Notes in Computer Science, Vol. 198, Springer-Verlag, Berlin, 1987, pp. 236–253.

[SLG94] V. Stoltenberg-Hansen, I. Lindström and E. Griffor, *Mathematical Theory of Domains.* Cambridge Tracts in Theoretical Computer Science No. 22, Cambridge University Press, Cambridge, 1994.

[ZR97a] G-Q. Zhang and W.C. Rounds, *Complexity of Power Default Reasoning.* In: Proceedings of the Twelfth Annual IEEE Symposium on Logic in Computer Science, LICS'97, Warsaw, Poland, July 1997. IEEE Computer Society Press pp. 328–339.

[ZR97b] G-Q. Zhang and W.C. Rounds, *Power defaults (preliminary report).* In: J. Dix, U. Furbach and A. Nerode (Eds.), Proceedings of the Fourth International Conference on Logic Programming and Non-Monotonic Reasoning (LPNMR'97), Dagstuhl, Germany, July 28-31, 1997. Lecture Notes in Computer Science, Vol. 1265, Springer-Verlag, 1997, pp. 152–169.

[ZR98] G-Q. Zhang and W.C. Rounds, *Semantics of Logic Programs and Representation of Smyth Powerdomain.* Preprint, 1998, pp. 1–24.[5]

[5] Available from http://www.cs.uga.edu/∼gqz/ under the title "An information-system representation of the Smyth powerdomain".