IPDL: a new modal logic of computation

Johan van Benthem Nick Bezhanishvili
Sebastian Enqvist

October 3, 2017

Abstract

We propose a new perspective on logics of computation by combining instantial neighborhood logic INL with bisimulation safe operations adapted from PDL and dynamic game logic. INL is a recently proposed modal logic, based on a richer extension of neighborhood semantics which permits both universal and existential quantification over individual neighborhoods. We show that a number of game constructors from game logic can be adapted to this setting to ensure invariance for instantial neighborhood bisimulations, which give the appropriate bisimulation concept for INL. We also prove that our extended logic IPDL is a conservative extension of dual-free game logic, and its semantics generalizes the monotone neighborhood semantics of game logic. Finally, we provide a sound and complete system of axioms for IPDL, and establish its finite model property and decidability.

1 Introduction

In this paper, we introduce a new modal logic of computation, in the style of propositional dynamic logic, based on *instantial neighborhood logic* INL [3]. The logic INL is based on a recent variant of monotone neighborhood semantics for modal logics, called instantial neighborhood semantics. In the standard neighborhood semantics, the box operator has the interpretation: $\Box p$ is true at a point if *there exists* a neighborhood in which *all* the elements satisfy the proposition p. So the box operator has a built-in fixed existential-universal quantifier pattern. In instantial neighborhood logic, we allow both universal and existential quantification over individual neighborhoods, so the basic modality has the form $\Box(p_1, ..., p_n; q)$. This formula is true at a point if *there exists* a neighborhood N in which *all* the elements satisfy the proposition q, and furthermore each of the propositions $p_1, ..., p_n$ are satisfied by *some* elements of N. INL is more expressive than monotone neighborhood logic, and comes with a natural associated notion of bisimulation together with a Hennessy-Milner theorem for finite models. It has a complete system of axioms, has the finite model property, is decidable and PSpace-complete.

Formally, our proposal is to consider an extension of the base language INL by bisimulation safe "program constructors", as in the standard propositional dynamic logic of sequential programs (PDL). The usual repertoire here consists of choice, test, sequential composition and a Kleene star for program iteration. Similar additions have already been studied extensively for the standard (monotone) neighborhood semantics, where the constructors are interpreted as methods of constructing complex *games* (this idea dates back to [13]). In the neighborhood setting, some additional operations are available, including the *dual* construction. This is a very powerful construction, and it is well known that dynamic game logic is not contained in any fixed level of the μ-calculus alternation hierarchy [4].

We think of our extended logic, which we call instantial PDL (IPDL for short), as a dynamic logic for a richer notion of computation than sequential programs. We consider a computational process as an agent acting in an uncertain environment that affects the outcome of each action. This is similar to the thinking behind the alternating-time temporal logic ATL of Alur et al. [1]. Dynamic game logic can be interpreted in a similar way, thinking of processes as "games against the environment". Instantial neighborhood semantics introduces a more fine-grained perspective to this setting, with a more expressive language and a finer bisimulation concept than standard neighborhood bisimilarity, namely the instantial neighborhood bisimulations of [3]. The game theoretic interpretation is made formally precise in Section **??**.

We provide sound and complete axioms for our instantial propositional dynamic logic IPDL, prove decidability via finite model property, and establish bisimulation invariance. The latter amounts to bisimulation safety for our program constructors. The completeness proof for the language IPDL, including all the program constructors that we consider, is based on the standard completeness proof for PDL (see [5] for an exposition), but involves some non-trivial new features. In particular, the axiom system requires two distinct induction rules, corresponding to a nested least fixpoint induction, and the model construction makes heavy use of a normal form for INL-formulas established in [3].

Overview of the paper

We first introduce syntax and semantics of instantial neighborhood logic, and extensions of it leading up to the full language IPDL, in an abstract setting, provide sound and complete systems of axioms, and prove decidability. We then motivate the setup by providing a formally precise game theoretic interpretation of the neighborhood semantics, with concrete interpretations of the program constructors that are familiar from game logic. Finally, we prove that our logic is a conservative extension of the dual free fragment of dynamic game logic.

2 Instantial neighborhood logic

2.1 Syntax and semantics

We start by reviewing the basic language for instantial neighborhood semantics. The only difference with our first paper on instantial neighborhood logic is that we are interpreting the language over *labelled* neighborhood structures, where the labels play the same role as "atomic programs" in PDL or "atomic games" in game logic.

The syntax of INL is given by the following grammar:

$$\varphi := p \in \mathsf{Prop} \mid \varphi \wedge \varphi \mid \neg\varphi \mid [a](\Psi;\varphi)$$

where a ranges over a fixed set $\mathcal{A}$ of *atomic labels*, and Ψ ranges over finite sets of formulas of INL. We have deviated a bit from the syntax of [3] here in allowing Ψ to be a finite *set* rather than a tuple of formulas. We shall sometimes write $[a](\psi_1, ..., \psi_n; \varphi)$ rather than $[a](\{\psi_1, ..., \psi_n\}; \varphi)$, in particular we write $[a](\psi;\varphi)$ rather than $[a](\{\psi\};\varphi)$, and $[a]\varphi$ rather than $[a](\emptyset;\varphi)$.

Formulas in INL will be interpreted over neighborhood structures.

Definition 1. A *neighborhood* frame is a structure (W, R) where W is a set and R associates with each $a \in \mathcal{A}$ a binary relation $R_a \subseteq W \times \mathcal{P}W$. A *neighborhood model* (W, R, V) is a neighborhood frame together with a valuation $V : \mathsf{Prop} \to \mathcal{P}W$.

We define the interpretations of all formulas in a neighborhood model $\mathfrak{M} = (W, R, V)$ as follows:

- $[\![p]\!] = V(p)$.

- $[\![\varphi \wedge \psi]\!] = [\![\varphi]\!] \cap [\![\psi]\!]$.

- $[\![\neg\varphi]\!] = W \setminus [\![\varphi]\!]$.

- $u \in [\![[a](\psi_1, ..., \psi_k; \varphi)]\!]$ iff there is some $Z \subseteq W$ such that:

$(u, Z) \in R_a$ and $Z \subseteq [\![\varphi]\!]$, $Z \cap [\![\psi_i]\!] \neq \emptyset$ for $i \in \{1, ..., k\}$

We write $\mathfrak{M}, v \Vdash \varphi$ for $v \in [\![\varphi]\!]$, and we write $\Vdash \varphi$ and say that φ is *valid* if, for every game model $\mathfrak{M}$ and $v \in W$, we have $\mathfrak{M}, v \Vdash \varphi$. We allow the notation $[\![-]\!]_{\mathfrak{M}}$ to make explicit reference to the model in the background.

Neighborhood models come with a natural notion of bisimulation, introduced in a more general setting in [3]. For this definition, the so called *Egli-Milner lifting* of a binary relation will play an important role:

Definition 1. The *Egli-Milner lifting* of a binary relation $R \subseteq X \times Y$, denoted $\overline{R}$, is a relation from $\mathcal{P}X$ to $\mathcal{P}Y$ defined by: $Z\overline{R}Z'$ iff:

1. For all $z \in Z$ there is some $z' \in Z'$ such that zRz'.

2. For all $z' \in Z'$ there is some $z \in Z$ such that zRz'.

We write $R; S$ for the composition of relations R and S. It is well known that the Egli-Milner lifting preserves relation composition:

$$\overline{R; S} = \overline{R}; \overline{S}$$

Definition 2. Let $\mathfrak{M} = (W, R, V)$ and $\mathfrak{M}' = (W', R', V')$ be any neighborhood models. The relation $B \subseteq W \times W'$ is said to be an *instantial neighborhood bisimulation* if for all uBu' and all atomic labels a we have:

Atomic For all p, $u \in V(p)$ iff $u' \in V'(p)$.

Forth For all Z such that uR_aZ, there is some Z' such that $u'R'_aZ'$ and $Z\overline{B}Z'$.

Back For all Z' such that $u'R'_aZ'$ there is some Z such that uR_aZ and $Z\overline{B}Z'$.

We say that pointed models $\mathfrak{M}, w$ and $\mathfrak{N}, v$ are *bisimilar*, written $\mathfrak{M}, w \underline{\longleftrightarrow} \mathfrak{N}, v$, if there is an instantial neighborhood bisimulation B between $\mathfrak{M}$ and $\mathfrak{N}$ such that wBv.

It is easy to check that all formulas of INL are invariant for instantial neighborhood bisimilarity:

Proposition 1. *If* $\mathfrak{M}, w \underline{\longleftrightarrow} \mathfrak{N}, v$ *then* $\mathfrak{M}, w \Vdash \varphi$ *iff* $\mathfrak{N}, v \Vdash \varphi$*, for each formula* φ *of INL.*

2.2 Axiomatization

We now turn to the task of axiomatizing the valid formulas of INL. Our system of axioms is a gentle modification of the axiom system for instantial neighborhood logic presented in [3].

INL axioms

Mon: $[a](\psi_1, ..., \psi_n; \varphi) \to [a](\psi_1 \vee \alpha_1, ..., \psi_n \vee \alpha_n; \varphi \vee \beta)$

Weak: $[a](\Psi; \varphi) \to [a](\Psi'; \varphi)$ for $\Psi' \subseteq \Psi$

Un: $[a](\psi_1, ..., \psi_n; \varphi) \to [a](\psi_1 \wedge \varphi, ..., \psi_n \wedge \varphi; \varphi)$

Lem: $[a](\Psi; \varphi) \to [a](\Psi \cup \{\gamma\}; \varphi) \vee [a](\Psi; \varphi \wedge \neg\gamma)$

Bot: $\neg[a](\bot; \varphi)$

Rules

MP:

$$\frac{\varphi \to \psi \qquad \varphi}{\psi}$$

RE:

$$\frac{\varphi \leftrightarrow \psi \qquad \theta}{\theta[\varphi/\psi]}$$

where $\theta[\varphi/\psi]$ is the result of substituting some occurrences of the formula ψ by φ in θ.

We denote this system of axioms by $\mathsf{Ax1}$ and write $\mathsf{Ax1} \vdash \varphi$ to say that the formula φ is provable in this axiom system. We also write $\varphi \vdash_{\mathsf{Ax1}} \psi$ for $\mathsf{Ax1} \vdash \varphi \to \psi$, and say that φ *provably entails* ψ.

Theorem 1. *The system* $\mathsf{Ax1}$ *is sound and complete for validity on neighborhood models.*

The proof of this result is essentially the same as in [3], and will not be repeated here.

Since the proof in [3] constructs a finite model for each consistent formula, we also get:

Theorem 2. *The logic INL is decidable and has the finite model property.*

3 Program operations

We now extend the language INL with four basic PDL-style operations: test, choice, parallel composition and sequential composition. The resulting language will be called *dynamic instantial neighborhood logic*, or (DINL). The syntax of DINL is defined by the following dual grammar.

$$\varphi := p \in \mathsf{Prop} \mid \varphi \wedge \varphi \mid \neg\varphi \mid [\pi](\Psi; \varphi)$$

$$\pi := a \in \mathcal{A} \mid \varphi? \mid \pi \cup \pi \mid \pi \cap \pi \mid \pi \circ \pi$$

We define the interpretation $[\![o]\!]$ of each operation $o \in \{\cup, \cap, \circ\}$ in a neighborhood model $\mathfrak{M}$ as a binary map from pairs of neighborhood relations to neighborhood relations, as follows:

- $R_1[\![\cup]\!]R_2 = R_1 \cup R_2$
- $R_1[\![\cap]\!]R_2 = \{(w, Z_1 \cup Z_2) \mid (w, Z_1) \in R_1 \;\&\; (w, Z_2) \in R_2\}$
- $(w, Z) \in R_1[\![\circ]\!]R_2$ iff there is some set Y and some family of sets F such that $(w, Y) \in R_1$, $(Y, F) \in \overline{R}_2$ and $Z = \bigcup F$.

The interpretation $[\![?]\!]$ of the test operator will be a map $[\![?]\!]$ assigning a neighborhood relation to each subset Z of W, defined by:

$$[\![?]\!]Z := \{(u, \{u\}) \mid u \in Z\}$$

Note that $[\![?]\!]$ is monotone in the sense that $Z \subseteq Z'$ implies $[\![?]\!]Z \subseteq [\![?]\!]Z'$. Each operator $o \in \{\cup, \cap, \circ\}$ is also monotone, in the sense that $R_1[\![o]\!]R_2 \subseteq R_1'[\![o]\!]R_2'$ whenever $R_1 \subseteq R_1'$ and $R_2 \subseteq R_2'$. For the sequential composition operator, this uses the well known fact that the Egli-Milner lifting is monotone, i.e. $\overline{R} \subseteq \overline{R'}$ whenever $R \subseteq R'$.

We can now define the semantic interpretations of all formulas, and the neighborhood relations corresponding to all complex labels π, by the following mutual recursion:

- $[\![p]\!] = V(p)$.

- $[\![\varphi \wedge \psi]\!] = [\![\varphi]\!] \cap [\![\psi]\!]$.

- $[\![\neg\varphi]\!] = W \setminus [\![\varphi]\!]$.

- $u \in [\![[\pi](\psi_1, ..., \psi_k; \varphi)]\!]$ iff there is some $Z \subseteq W$ such that:

$(u, Z) \in R_\pi$ and $Z \subseteq [\![\varphi]\!]$, $Z \cap [\![\psi_i]\!] \neq \emptyset$ for $i \in \{1, ..., k\}$.

- $R_{\pi_1 o \pi_2} = R_{\pi_1}[\![o]\!]R_{\pi_2}$ for $o \in \{\cup, \cap, \circ\}$.

- $R_{\varphi?} = [\![?]\!][\![\varphi]\!]$

To motivate the semantic interpretations of the dynamic operators, we show how they in a precise sense generalize familiar operations from game logic.

Definition 3. Let $\mathfrak{M} = (W, R, V)$ be a neighborhood model. Then $\mathfrak{M}$ is said to be *monotone* if for all atomic labels $a \in \mathcal{A}$, $w \in W$ and $Z, Z' \subseteq W$: if $(u, Z) \in R_a$ and $Z \subseteq Z'$ then $(u, Z') \in R_a$ also.

The definitions of the dynamic operations are tailored towards obtaining the following result:

Proposition 2. *All formulas of DINL are invariant for instantial neighborhood bisimulations.*

3.1 Axiomatization

Our axiom system for DINL will take the sound and complete axioms for INL as its foundation, and extend it with reduction axioms for the test, choice, parallel composition and sequential composition operators. The axioms and rules are listed below; note that the INL axioms and the axioms for frame constraints are now stated for arbitrary complex labels π rather than just atoms a.

INL axioms:

(Mon), (Weak), (Un), (Lem) and (Bot)

Reduction axioms:

Test: $[\gamma?](\Psi;\varphi) \leftrightarrow \gamma \wedge \bigwedge \Psi \wedge \varphi$

Ch: $[\pi_1 \cup \pi_2](\Psi;\varphi) \leftrightarrow [\pi_1](\Psi;\varphi) \vee [\pi_2](\Psi;\varphi)$

Pa: $[\pi_1 \cap \pi_2](\Psi;\varphi) \leftrightarrow \bigvee\{[\pi_1](\Theta_1;\varphi) \wedge [\pi_2](\Theta_2;\varphi) \mid \Psi = \Theta_1 \cup \Theta_2\}$

Cmp: $[\pi_1 \circ \pi_2](\psi_1,...,\psi_n;\varphi) \leftrightarrow [\pi_1]([\pi_2](\psi_1;\varphi),...,[\pi_2](\psi_n;\varphi);[\pi_2]\varphi)$

Rules:

(MP) and (RE)

We denote this system of axioms by $\mathsf{Ax2}$ and write $\mathsf{Ax2} \vdash \varphi$ to say that the formula φ is provable in this axiom system. We also write $\varphi \vdash_{\mathsf{Ax2}} \psi$ for $\mathsf{Ax2} \vdash \varphi \to \psi$. We shall sometimes drop the reference to $\mathsf{Ax2}$ to keep notation cleaner.

Proposition 3 (Soundness)**.** *If $\mathsf{Ax2} \vdash \varphi$, then φ is valid on all neighborhood models.*

By applying soundness of the reduction axioms, we can use a standard argument to obtain for every consistent formula φ of DINL a provably (and hence semantically) equivalent formula φ^t in INL, which is then satisfiable by Theorem 1. For example, the formula $[\gamma?](\psi_1,...,\psi_n;\varphi)^t$ is defined to be $\gamma^t \wedge \psi_1^t \wedge ... \wedge_n^t \wedge \varphi$. We get:

Theorem 3 (Completeness)**.** *A formula φ of DINL is valid on all neighborhood models iff $\mathsf{Ax2} \vdash \varphi$.*

Furthermore, the finite model property and decidability clearly carry over from INL:

Theorem 4. *The logic DINL is decidable and has the finite model property.*

4 Game interpretation

4.1 Games and neighborhoods

In this section we provide an interpretation of our neighborhood semantics in terms of games. We shall think of programs as divided into two components: *computations* that merely change the internal state of the system, and *input-output* stages at the end of each computation in which the system communicates with the external world. In concrete applications this could consist of receiving a signal along some channel of communication, producing a "side effect" of the computation in the form of an observable output, checking the value of some random variable etc. Furthermore we shall assume that the system has only finitely many internal states, but that there are prima facie infinitely many possible results of communicating with the outside world at an "input-output" stage. This assumption seems to be fairly mild, since many models of computation (Turing machines, finite automata, Mealy/Moore machines etc.) do in fact assume that the system itself has only finitely possible internal states but can in principle receive or produce an infinite number of possible inputs and/or outputs. In a game theoretic context one interpretation in particular comes to mind, namely to think of the program terms as denoting *game forms* that describe the underlying structure of a game in terms of possible moves for each player at each position of the game, and taking the possible "outputs" of a game to be elements of $\mathbb{R}$ representing the possible *payoffs* for some given player. For a finite game tree, this application will of course satisfy our assumption. [1]

We begin with revisiting some standard and basic definitions of game theory, mainly to fix notation:

Definition 2. Let Σ be any set. A (wellfounded) Σ-*tree* $\mathcal{T}$ non-empty and prefix closed subset of Σ^*, the collection of finite words over the alphabet Σ, such that every subset of $\mathcal{T}$ that is linearly ordered by the prefix relation is finite. The empty word ε is called the *root* of the tree. Given a word $\vec{m} \in \mathcal{T}$, if $m' \in \Sigma$ is such that $\vec{m} \cdot m' \in \mathcal{T}$ then $\vec{m} \cdot m'$ is called a *child* of $\vec{m}$ and $\vec{m}$ is called the *parent* of $\vec{m} \cdot m'$. A node in $\mathcal{T}$ is called a *leaf* if it has no children. A *branch* of a tree $\mathcal{T}$ is a maximal subset of $\mathcal{T}$ that is linearly ordered by the prefix relation.

Definition 3. Let O be any non-empty set. A *n-player game* with *outcomes in O* is a structure $\mathcal{G} = (\Pi, \Sigma, \mathcal{T}, t, o)$ where:

- Π is a set of n *players*,
- Σ is a set of *moves*,
- $\mathcal{T}$ is a wellfounded Σ-tree,
- t is a map sending each non-leaf node in $\mathcal{T}$ to a player in Π,
- o is a map sending each leaf node in $\mathcal{T}$ to an *outcome* in O.

A *strategy* for a player p in Π is a map $\sigma : t^{-1}[\mathcal{T}] \to \Sigma$ such that $\vec{m} \cdot \sigma(\vec{m}) \in \mathcal{T}$ for each $\vec{m} \in \mathcal{T}$ with $t(\vec{m}) = p$. A *match* M in a game $\mathcal{G}$ is a branch of $\mathcal{T}$, and a match M is said to be *guided* by a strategy σ for $p \in \Pi$ if for all $\vec{m} \in M$ with $t(\vec{m}) = p$, $\vec{m} \cdot \sigma(\vec{m}) \in M$ also. Since branches of $\mathcal{T}$ are in one-to-one

[1] Note also that the assumption that a system has only finitely many states is generally consistent with our neighborhood semantics, since all the logics that we consider here will be shown to have the finite model property, so we could equally well use *finite* neighborhood models for a sound and complete semantics.

correspondence with leaves, we can also identify a match with the corresponding leaf. The *outcome set* of a strategy σ, denoted $o[\sigma]$, is the set of all elements of O of the form $o(l)$ where l corresponds to some σ-guided match.

Definition 4. Let Γ be an infinite set and W a finite set. A *two-player game $\mathcal{G}$ over W with communication over the channel* Γ is a tuple $(\Sigma, \mathcal{T}, t, o)$, where Σ is a finite set and

$$(\{\mathsf{A}, \mathsf{B}, [\mathtt{I/O}]\}, \Sigma \cup \Gamma, \mathcal{T}, t, o')$$

is a three-player game with players $\{\mathsf{A}, \mathsf{B}, [\mathtt{I/O}]\}$, such that:

– if $t(\vec{m}) = [\mathtt{I/O}]$ then the children of $\vec{m}$ are precisely the nodes of the form $\vec{m} \cdot s$ for $s \in \Gamma$,

– if some child of $\vec{m}$ is a leaf in $\mathcal{T}$, then all children of $\vec{m}$ are leaves, $t(\vec{m}) = [\mathtt{I/O}]$, and $o(\vec{m} \cdot s) = o(\vec{m} \cdot s')$ for all $s, s' \in \Gamma$,

– if $t(\vec{m}) \neq [\mathtt{I/O}]$ then $m' \in \Sigma$ for every child $\vec{m} \cdot m'$ of $\vec{m}$,

– the root ε is not a leaf of $\mathcal{T}$.

The players of a two-player game $\mathcal{G}$ with communication over a channel are A and B. We think of the extra player $[\mathtt{I/O}]$ as a process that deterministically produces the new internal state of the system at the end of a computation, and also non-deterministically chooses a signal along the channel Γ. We say that the game $\mathcal{G}$ is *atomic* if the only nodes of the form $\vec{m} \cdot s$ for $s \in \Gamma$ are leaves, and *composite* otherwise - the intution being that nodes belonging to $[\mathtt{I/O}]$ are stages of a composite game at which one if its component games ends and the system communicates with the external world. We denote the set of all games over W with communication over the channel Γ by $\mathbb{G}(W, \Gamma)$.

Definition 5. A *dynamic game over* W with communication over the channel Γ is a map:

$$g : W \to \mathbb{G}(W, \Gamma)$$

The *basic power relation* $\mathtt{Pow}(g) \subseteq W \times \mathcal{P}W$ induced by g is defined by setting $(w, Z) \in \mathtt{Pow}(g)$ iff there is a strategy σ for Player A in $g(w)$ with $o[\sigma] = Z$.

Definition 6. A relation $R \subseteq W \times \mathcal{P}W$ will be called a *standard neighborhood relation* if for all $w \in W$:

1. $R[w] \neq \emptyset$,

2. $\emptyset \notin R[w]$.

Proposition 4. *Let Γ be an infinite set and W any finite set. A neighborhood relation $R \subseteq W \times \mathcal{P}W$ is standard iff there is a dynamic game $g : W \to \mathbb{G}(W, \Gamma)$ such that $R = \mathtt{Pow}(g)$, iff there is an atomic dynamic game g such that $R = \mathtt{Pow}(g)$.*

Proof. It is clear that $\mathtt{Pow}(g)$ is a standard neighborhood relation for any dynamic game $g : W \to \mathbb{G}(W, \Gamma)$, atomic or composite. Conversely, suppose that $R \subseteq W \times \mathcal{P}W$ is a standard neighborhood relation on W. We sketch the construction of a dynamic game g: construct the game $g(w)$ to be played in three rounds:

Round 1 Player A chooses a set Z such that wRZ,

Round 2 Player B chooses an element of the set Z,

Round 3 Player $[\mathtt{I}/\mathtt{O}]$ chooses a signal in Γ.

The outcome of a match is the element chosen by Player B in the second round. It is clear that this is an atomic game with communication along the channel Γ and that $\mathtt{Pow}(g) = R$. $\square$

Remark 1. Note that we have not assumed that all neighborhood relations are standard, since we want to represent the test operator by a neighborhood relation. We can think of this particular case as a non-standard game, which violates the usual condition that each player has at least one available strategy.

4.2 Game operations

Next, we consider three operations on dynamic games:

Choice

Given dynamic games $g_1, g_2 : W \to \mathbb{G}(W, \Gamma)$, we define the game $g_1 \cup g_2$ that offers Player A a choice between the two games g_1, g_2. Formally, given that $g_1(w) = (\Sigma_1, \mathcal{T}_1, t_1, o_1)$ and $g_2(w) = (\Sigma_2, \mathcal{T}_2, t_2, o_2)$, we define $(g_1 \cup g_2)(w) = (\Sigma', \mathcal{T}', t', o')$ by setting:

- $\Sigma' = \Sigma_1 \cup \Sigma_2 \cup \{L, R\}$
- $\mathcal{T}' = \{\varepsilon\} \cup \{L \cdot \vec{m} \mid \vec{m} \in \mathcal{T}_1\} \cup \{R \cdot \vec{m} \mid \vec{m} \in \mathcal{T}_2\}$
- $t'(\varepsilon) = \mathsf{A}$,
- $t'(L \cdot \vec{m}) = t_1(\vec{m})$ for $\vec{m} \in \mathcal{T}_1$ and $t'(R \cdot \vec{m}) = t_2(\vec{m})$ for $\vec{m} \in \mathcal{T}_2$,
- if $X \cdot \vec{m}$ is a leaf then $o'(X \cdot \vec{m}) = o_1(\vec{m})$ if $X = L$, $o'(X \cdot \vec{m}) = o_2(\vec{m})$ if $X = R$.

Dual choice

The dual choice operation offers a choice between two games for B rather than A: given dynamic games $g_1, g_2 : W \to \mathbb{G}(W, \Gamma)$ we define the game $g_1 \cap g_2$ as follows. If $g_1(w) = (\Sigma_1, \mathcal{T}_1, t_1, o_1)$ and $g_2(w) = (\Sigma_2, \mathcal{T}_2, t_2, o_2)$, then we define $(g_1 \cap g_2)(w) = (\Sigma', \mathcal{T}', t', o')$ by setting:

- $\Sigma' = \Sigma_1 \cup \Sigma_2 \cup \{L, R\}$

- $\mathcal{T}' = \{\varepsilon\} \cup \{L \cdot \vec{m} \mid \vec{m} \in \mathcal{T}_1\} \cup \{R \cdot \vec{m} \mid \vec{m} \in \mathcal{T}_2\}$
- $t'(\varepsilon) = \mathsf{B}$,
- $t'(L \cdot \vec{m}) = t_1(\vec{m})$ for $\vec{m} \in \mathcal{T}_1$ and $t'(R \cdot \vec{m}) = t_2(\vec{m})$ for $\vec{m} \in \mathcal{T}_2$,
- if $X \cdot \vec{m}$ is a leaf then $o'(X \cdot \vec{m}) = o_1(\vec{m})$ if $X = L$, $o'(X \cdot \vec{m}) = o_2(\vec{m})$ if $X = R$.

Sequential composition

The sequential composition $g_1 \circ g_2$ of two games $g_1, g_2 : W \to \mathbb{G}(W, \Gamma)$ is defined as a game where first a match of g_1 is played, and then a match of g_2 is played. If $g_1(w) = (\Sigma_1, \mathcal{T}_1, t_1, o_1)$ and $g_2(v) = (\Sigma_2^v, \mathcal{T}_2^v, t_2^v, o_2^v)$ for each $v \in W$ then we define $(g_1 \cup g_2)(w) = (\Sigma', \mathcal{T}', t', o')$ by setting:

- $\Sigma' = \Sigma_1 \cup \Sigma_2$
- $\mathcal{T}' = \mathcal{T}_1 \cup \{\vec{m} \cdot \vec{n} \mid \vec{m} \text{ a leaf of } \mathcal{T}_1 \ \& \ \vec{n} \in \mathcal{T}_2^{o_1(\vec{m})}\}$
- $t'(\vec{m}) = t_1(\vec{m})$ for $\vec{m} \in \mathcal{T}_1$ and $t'(\vec{m} \cdot \vec{n}) = t_2^{o_1(\vec{m})}(\vec{n})$ for $\vec{m}$ a leaf of $\mathcal{T}_1$ and $\vec{n} \in \mathcal{T}_2^{o_1(\vec{m})}$,
- $o'(\vec{m} \cdot \vec{n}) = o_2^{o_1 \vec{m}}(\vec{n})$ if $\vec{m} \cdot \vec{n}$ is a leaf.

Note that all three game operations preserve the conditions we imposed on games in $\mathbb{G}(W, \Gamma)$, so that they are indeed well defined binary operations on $\mathbb{G}(W, \Gamma)$. Furthermore, we can now show that our game interpretation of neighborhood structures gives the right result for the program constructors:

Proposition 5. *Let $g_1, g_2 : W \to \mathbb{G}(W, \Gamma)$ be dynamic games over W with communication along the channel Γ, where W is a finite set of states and Γ an infinite set. Then:*

- $\mathtt{Pow}(g_1 \cup g_2) = \mathtt{Pow}(g_1)[\![\cup]\!]\mathtt{Pow}(g_2)$
- $\mathtt{Pow}(g_1 \cap g_2) = \mathtt{Pow}(g_1)[\![\cap]\!]\mathtt{Pow}(g_2)$
- $\mathtt{Pow}(g_1 \circ g_2) = \mathtt{Pow}(g_1)[\![\circ]\!]\mathtt{Pow}(g_2)$

Proof. The easy proofs for choice and dual choice are left to the reader. We focus on the sequential composition operation:

Pick some $w \in W$, and fix the notation $g_1(w) = (\Sigma_1, \mathcal{T}_1, t_1, o_1)$ and $g_2(v) = (\Sigma_2^v, \mathcal{T}_2^v, t_2^v, o_2^v)$ for each $v \in W$. Suppose first that $(w, Z) \in \mathtt{Pow}(g_1 \circ g_2)$. Then there is some strategy σ for Player A in $(g_1 \circ g_2)(w)$ such that $o[\sigma] = Z$. We define a strategy σ_1 for Player A in $g_1(w)$ by letting σ_1 be the restriction of the map σ to the set of positions in $g_1(w)$ belonging to A. Let $o_1[\sigma_1] \subseteq W$ be the outcome set of this strategy in $g_1(w)$, so that $(w, o_1[\sigma_1]) \in \mathtt{Pow}(g_1)$. For each leaf $\vec{m} \in \mathcal{T}_1$ corresponding to a σ_1-guided match, define the strategy $\sigma_2^{\vec{m}}$ for Player A in $g_2(o_1(\vec{m}))$ by setting:

$$\sigma_2^{\vec{m}}(\vec{n}) = \sigma(\vec{m} \cdot \vec{n})$$

for each $\vec{n}$ belonging to Player A in $g_2(o_1(\vec{m}))$. Then for each leaf $\vec{m}$ in $\mathcal{T}_1$ corresponding to a σ_1-guided match, we have

$$(o_1(\vec{m}), o_2^{o_1(\vec{m})}[\sigma_2^{\vec{m}}]) \in \mathtt{Pow}(g_2)$$

But it is quite obvious that the union of the family F of all sets $o_2^{o_1(\vec{m})}[\sigma_2^{\vec{m}}]$, for $\vec{m}$ in $\mathcal{T}_1$ corresponding to a σ_1-guided match, is equal to $g[\sigma]$, i.e. to Z. Since $(w, o_1[\sigma_1]) \in \mathtt{Pow}(g_1)$ and $(o_1[\sigma_1], F) \in \overline{\mathtt{Pow}(g_2)}$, it follows that $(w, Z) \in \mathtt{Pow}(g_1)[\![\circ]\!]\mathtt{Pow}(g_2)$ as required.

Conversely, suppose that $(w, Z) \in \mathtt{Pow}(g_1)[\![\circ]\!]\mathtt{Pow}(g_2)$. Then there is a set $Y \subseteq W$ and a family of sets $F \subseteq \mathcal{P}W$ such that $(w, Y) \in \mathtt{Pow}(g_1)$, $(Y, F) \in \overline{\mathtt{Pow}(g_2)}$ and $\bigcup F = Z$. By definition of $\mathtt{Pow}(g_1)$, there exists a strategy σ_1 for A in $g_1(w)$ such that $o_1[\sigma_1] = Y$. Furthermore, by the definition of a game with communication along the channel Γ, it is clear that for each $v \in Y$ there are at least $|\Gamma|$ many σ_1-guided matches $\vec{m}$ such that $o_1(\vec{m}) = v$. So let f be a surjective map from the set of σ_1-guided matches onto F such that $(o_1(\vec{m}), f(\vec{m})) \in \mathtt{Pow}(g_2)$ for each σ_1-guided match $\vec{m}$, which clearly exists since:

- the domain of f is of greater cardinality than F, which must be finite,
- $(Y, F) \in \overline{\mathtt{Pow}(g_2)}$ and
- $o_1[\sigma_1] = Y$.

For each σ_1-guided match $\vec{m}$, since $(o_1(\vec{m}), f(\vec{m})) \in \mathtt{Pow}(g_2)$ there exists a strategy $\sigma_2^{\vec{m}}$ for Player A in $g_2(o_1(\vec{m}))$ such that $o_2^{o_1(\vec{m})}[\sigma_2^{\vec{m}}] = f(\vec{m})$. Now, define a strategy σ in $(g_1 \circ g_2)(w)$ by setting $\sigma(\vec{m}) = \sigma_1(\vec{m})$ for a position $\vec{m}$ belonging to Player A and which is also a position in $g_1(w)$, and set

$$\sigma(\vec{m} \cdot \vec{n}) = \sigma_2^{\vec{m}}(\vec{n})$$

for a position of the form $\vec{m} \cdot \vec{n}$ where $\vec{m}$ is a leaf in $g_1(w)$ corresponding to a σ_1-guided match. It is easy to see that $o[\sigma] = Z$, so $(w, Z) \in \mathtt{Pow}(g_1 \circ g_2)$ as required. $\square$

5 Program iteration and the language IPDL

We now introduce the final operation that we consider here, a Kleene star for finite iteration. This operation will be set up to generalize the game iteration operation from game logic. The corresponding language will be denoted by IPDL, read "instantial PDL", and is given by the following dual grammar:

$$\varphi := p \in \mathsf{Prop} \mid \varphi \wedge \varphi \mid \neg\varphi \mid [\pi](\Psi; \varphi)$$

$$\pi := a \in \mathcal{A} \mid \varphi? \mid \pi \cup \pi \mid \pi \cap \pi \mid \pi \circ \pi \mid \pi^*$$

For the semantic interpretation of the Kleene star, it will be useful to first define the relation skip by:

$$\mathsf{skip} := \{(w, \{w\}) \mid w \in W\}$$

We now define a relation $R^{[\xi]}$ for each ordinal ξ by induction as follows.

- $R^{[0]} = \emptyset$
- $R^{[\xi+1]} = \mathsf{skip}[\![\cup]\!](R[\![\circ]\!]R^{[\xi]})$
- $R^{\kappa} = \bigcup_{\xi<\kappa} R^{[\xi]}$ if κ is a limit ordinal.

We define $[\![*]\!]R$ to be equal to $R^{[\xi]}$, where ξ is the smallest ordinal satisfying $R^{[\xi]} = R^{[\xi+1]}$. It is easy to see that this is a standard least fixpoint construction, in particular we have:

Proposition 6. *Let W be a finite set and $R \subseteq W \times \mathcal{P}(W)$. Then:*

$$[\![*]\!]R = \bigcup_{n\in\omega} R^{[n]}$$

Semantics of IPDL-formulas in a neighborhood model $\mathfrak{M} = (W, R, V)$ are now defined as follows:

- $[\![p]\!] = V(p)$.

- $[\![\varphi \wedge \psi]\!] = [\![\varphi]\!] \cap [\![\psi]\!]$.

- $[\![\neg\varphi]\!] = W \setminus [\![\varphi]\!]$.

- $u \in [\![[\pi](\psi_1, ..., \psi_k; \varphi)]\!]$ iff there is some $Z \subseteq W$ such that:

$(u, Z) \in R_\pi$ and $Z \subseteq [\![\varphi]\!]$, $Z \cap [\![\psi_i]\!] \neq \emptyset$ for $i \in \{1, ..., k\}$.

- $R_{\pi_1 o \pi_2} = R_{\pi_1}[\![o]\!]R_{\pi_2}$ for $o \in \{\cup, \cap, \circ\}$.

- $R_{\varphi?} = [\![?]\!][\![\varphi]\!]$.

- $R_{\pi^*} = [\![*]\!]R_\pi$.

Proposition 7. *All formulas of IPDL are invariant for instantial neighborhood bisimulations.*

The proof of this is a bisimulation safety argument, and the step for the Kleene star involves using the bisimulation safety of union and sequential composition to prove the appropriate back-and-forth conditions for each approximant $R^{[\xi]}_\pi$ of the least fixpoint $R_{\pi^*} = [\![*]\!]R_\pi$. We omit the details.

5.1 Axiomatization

Our axiomatization for IPDL is given below.

INL axioms:

(Mon), (Weak), (Un), (Lem) and (Bot).

Reduction axioms from DINL:

(Test), (Ch), (Pa) and (Cmp).

Basic rules:

(MP) and (RE).

Kleene star

Finally we add axioms and rules for iteration. The Kleene star is a least fixpoint construction, and a standard approach to axiomatizing least fixpoints is to use one *fixpoint axiom* and one *induction rule* (see [10]). The fixpoint axiom **Fix** is stated as follows:

$$[\pi^*](\Psi; \varphi) \leftrightarrow (\bigwedge \Psi \wedge \varphi) \vee [\pi \circ \pi^*](\Psi; \varphi)$$

We will actually need *two* induction rules:

Ind1:

$$\frac{\varphi \to \gamma \qquad [\pi]\gamma \to \gamma}{[\pi^*]\varphi \to \gamma}$$

Ind2:

$$\frac{(\psi \wedge \varphi) \to \gamma \qquad [\pi](\gamma; [\pi^*]\varphi) \to \gamma}{[\pi^*](\psi; \varphi) \to \gamma}$$

Remark 2. The reason that we require two distinct induction rules can be seen as follows: the reduction axioms for IPDL should be interpreted as encoding a recursive translation of the language IPDL into the modal μ-calculus (interpeted on instantial neighborhood models). When we pass by formulas involving the Kleene-star in this translation, the translation will not surprisingly involve least fixpoint operators, and the induction rules then correspond to the Kozen-Park induction rules for least fixpoint operators. This step of the translation is trickier than the step for the Kleene star in a translation of PDL into the μ-calculus (see [6]), and requires use of nested least fixpoint variables.

Note also that the second induction axiom only involves a single instantial formula ψ. This is because we can "pre-process" an arbitrary formula $[\pi^*](\psi_1, ..., \psi_n; \varphi)$ by applying the axiom **Fix**, and then applying the composition axiom (Cmp) to the formula $[\pi \circ \pi^*](\psi_1, ..., \psi_n; \varphi)$ to obtain the formula:

$$[\pi]([\pi^*](\psi_1; \varphi), ..., [\pi^*](\psi_n; \varphi); [\pi^*]\varphi)$$

Here, each occurrence of the operator $[\pi^*]$ is followed by at most one instantial formula.

We denote this axiom system as $\mathsf{Ax3}$ and write $\varphi \vdash_{\mathsf{Ax3}} \psi$ to say that $\mathsf{Ax3} \vdash \varphi \to \psi$. We will also sometimes drop the explicit reference to the system $\mathsf{Ax3}$, simply writing $\vdash \varphi$ or $\varphi \vdash \psi$.

Theorem 5. *The axiom system* $\mathsf{Ax3}$ *is sound and complete for validity over neighborhood models.*

The soundness part of this theorem is a fairly straightforward check. For the completeness proof, we shall rely heavily on the following lemma, which was proved (in a slightly different formulation) in [3]: fix a finite and subformula closed set of formulas Σ. An *atom* over Σ is a maximal consistent subset of Σ, and we denote the set of atoms over Σ by $\mathsf{At}(\Sigma)$. Given any atom $w \in \mathsf{At}(\Sigma)$, let $\widehat{w}$ be its conjunction, and let $\widehat{Z} = \{\widehat{w} \mid w \in Z\}$ for a set of atoms Z.

Lemma 1. *Let* $[\pi](\Psi;\varphi)$ *be any formula such that each formula in* $\Psi \cup \{\varphi\}$ *is a boolean combination of formulas in* Σ. *Then* $[\pi](\Psi;\varphi)$ *is provably equivalent to a disjunction of formulas of the form* $[\pi](\widehat{Z}; \bigvee \widehat{Z})$ *for* $Z \subseteq \mathsf{At}(\Sigma)$ *being some set of atoms with* $w \vdash \varphi$ *for each* $w \in Z$ *and for all* $\psi \in \Psi$ *there is some* $v \in Z$ *with* $v \vdash \psi$.

We shall also need an adapted concept of Fischer-Ladner closure:

Definition 4. A set Σ of formulas is said to be *Fischer-Ladner closed* if the following clauses hold:

- If $\varphi \in \Sigma$, and the main connective of φ is not $\neg$, then the formula $\neg\varphi$ is in Σ.
- Any subformula of a formula in Σ is in Σ.
- If $[\gamma?](\Psi;\varphi)$ is in Σ then so is $\gamma \wedge \bigwedge \Psi \wedge \varphi$.
- If $[\pi_1 \circ \pi_2](\psi_1, ..., \psi_n; \varphi) \in \Sigma$, then $[\pi_1]([\pi_2](\psi_1;\varphi), ..., [\pi_1](\psi_n;\varphi); [\pi_2]\varphi)$ is in Σ too.
- If $[\pi_1 \cup \pi_2](\Psi;\varphi) \in \Sigma$ then $[\pi_1](\Psi;\varphi) \vee [\pi_2](\Psi;\varphi) \in \Sigma$ too.
- If $[\pi_1 \cap \pi_2](\Psi;\varphi) \in \Sigma$ then the formula:

$$\bigvee \{[\pi_1](\Theta_1;\varphi) \wedge [\pi_2](\Theta_2;\varphi) \mid \Psi = \Theta_1 \cup \Theta_2\}$$

  is in Σ too.
- If $[\pi^*](\Psi;\varphi) \in \Sigma$ then $(\bigwedge \Psi \wedge \varphi) \vee [\pi \circ \pi^*](\Psi;\varphi)$ is in Σ too.

Lemma 2. *Every formula φ is a member of some finite Fischer-Ladner closed set of formulas.*

Proof. Standard, see for example [5]. $\square$

Fix a finite and Fischer-Ladner closed set of formulas Σ. An *atom* over Σ is a maximal consistent subset of Σ, and we denote the set of atoms over Σ by $\mathsf{At}(\Sigma)$. Given any atom $w \in \mathsf{At}(\Sigma)$, let $\widehat{w}$ be its conjunction, and let $\widehat{Z} = \{\widehat{w} \mid w \in Z\}$ for a set of atoms Z.

Lemma 3. *Let $[\pi](\Psi; \varphi)$ be any formula such that each formula in $\Psi \cup \{\varphi\}$ is a boolean combination of formulas in Σ. Then $[\pi](\Psi; \varphi)$ is provably equivalent to a disjunction of formulas of the form $[\pi](\widehat{Z}; \bigvee \widehat{Z})$ for $Z \subseteq \mathsf{At}(\Sigma)$ being some set of atoms with $w \vdash \varphi$ for each $w \in Z$ and for all $\psi \in \Psi$ there is some $v \in Z$ with $v \vdash \psi$.*

Definition 5. Given any label π, we define the relation $S^{\Sigma}_{\pi} \subseteq \mathsf{At}(\Sigma) \times \mathcal{P}(\mathsf{At}(\Sigma))$ by setting $(w, Z) \in S^{\Sigma}_{\pi}$ iff $\widehat{w} \wedge [\pi](\widehat{Z}; \bigvee \widehat{Z})$ is consistent with respect to the system $\mathsf{Ax3}$.

The *canonical neighborhood model* over Σ denoted $\mathfrak{C}^{\Sigma}$ is defined as the triple $(W^{\Sigma}, R^{\Sigma}, V^{\Sigma})$ where W^{Σ} is the set of atoms over Σ, $R^{\Sigma}_a = S^{\Sigma}_a$ for each atomic label a, and $V^{\Sigma}(p) = \{w \in W^{\Sigma} \mid p \in w\}$.

The key lemma in the completeness proof, which is proved using the induction rules for the Kleene star, is the following:

Lemma 4. *For each label π, we have $S^{\Sigma}_{\pi^*} \subseteq [\![*]\!](S^{\Sigma}_{\pi})$.*

Lemma 4 is needed to prove Lemma 5 below, by induction on the complexity of program terms. Say that a label π is *safe* if, for every formula γ such that the term $\gamma?$ appears in π, we have $\gamma \in \Sigma$ and furthermore, $\gamma \in w$ iff $\mathfrak{C}^{\Sigma}, w \Vdash \gamma$ for each $w \in \mathsf{At}(\Sigma)$.

Lemma 5. *For every safe label π, we have $S^{\Sigma}_{\pi} \subseteq R^{\Sigma}_{\pi}$.*

Using Lemma 5 we can prove a truth lemma for the canonical model:

Lemma 6. *For every atom w and any $\psi \in \Sigma$, we have $(\mathfrak{C}^{\Sigma}, w) \Vdash \psi$ if and only if $\psi \in w$.*

Finally, we can now prove Theorem 5: suppose the formula φ is not provable, so that $\neg\varphi$ is consistent. By Lemma 2, $\neg\varphi$ belongs to some finite Fischer-Ladner closed set Σ and since $\neg\varphi$ is consistent it belongs to some atom w. Hence $\varphi \notin w$ and by Lemma 6 we have $\mathfrak{C}^{\Sigma}, w \nVdash \varphi$. So φ is not valid.

As a corollary to the completeness proof, which produces a finite model for a consistent formula, we get:

Theorem 6. *IPDL has the finite model property and is decidable.*

6 Comparison with game logic

We now show that IPDL can, in a precise sense, be viewed as a language extension of dual-free game logic. We shall denote this language simply by GL, for "game logic", although the full dynamic game logic also includes a dual constructor. Formally, formulas of GL and game terms are defined by the following dual grammar:

$$\varphi := p \in \mathsf{Prop} \mid \varphi \wedge \varphi \mid \neg\varphi \mid [\pi]\varphi$$

$$\pi := a \in \mathcal{A} \mid \varphi? \mid \pi \circ \pi \mid \pi \cup \pi \mid \pi \cap \pi \mid \pi^*$$

where Prop is a fixed set of propositional variables and $\mathcal{A}$ is a set of atomic games, both assumed to be countably infinite. Note that GL is a syntactic fragment of IPDL. Here, $\cup$ is interpreted as "angelic choice" (choice for Player I), $\cap$ is interpreted as "demonic choice" (choice for Player II), $\circ$ is sequential game composition and $*$ is finite game iteration (controlled by Player I).

Semantics of game logic formulas are again given by neighborhood frames, with the extra constraint that neighborhoods associated with a world are upwards closed under subsethood:

Definition 6. A neighborhood frame (W, R) is said to be a *monotonic power frame* if the following condition holds for each $a \in \mathcal{A}$:

(Monotonicity) For all $u \in W$, if $(u, Z) \in R_a$ and $Z \subseteq Z'$ then $(u, Z') \in R_a$.

A monotonic power model is a neighborhood model whose underlying frame is a monotonic power frame.

In order to provide the semantic interpretations of formulas in a model, we need to provide semantic interpretations of the game constructors. We shall use double vertical lines $\|-\|$ to refer to semantic interpretations of formulas in GL and game constructors in monotonic neighborhood models, in order to distinguish it from the semantics given for IPDL, where we use square brackets $[\![-]\!]$. We follow the definitions in [2]. Formally, we define operations on the lattice $\mathcal{N}W = \mathcal{P}(W \times \mathcal{P}(W))$ of *neighborhood relations* over W as follows:

- $R\|\cup\|R' = R \cup R'$

- $R\|\cap\|R' = R \cap R'$

- $(u, Z') \in R\|\circ\|R'$ iff there is some $Z \subseteq W$ with $(u, Z) \in R$ and $(v, Z') \in R'$ for all $v \in Z$.

- $\|?\|(Z) = \{(w, Z') \in W \times \mathcal{P}(W) \mid w \in Z \cap Z'\}$

Finally, we define $\|*\|R$ to be the least fixpoint in the lattice $\mathcal{N}W$ of the monotone map F defined by:

$$FS = \mathsf{skip}^{\uparrow}\|\cup\|(R\|\circ\|S)$$

where $\mathsf{skip}^{\uparrow} = \{(w, Z) \in W \times \mathcal{P}(W) \mid w \in Z\}$. We can now set up the semantics of GL. Fixing a monotonic power model $\mathfrak{M}$, we define the interpretation of every formula φ and the neighborhood relations R_π corresponding to each game term π in the obvious way, so that in particular we have -$R_{\pi_1 \cup \pi_2} = R_{\pi_1} \|\cup\| R_{\pi_2}$, $R_{\pi_1 \cap \pi_2} = R_{\pi_1} \|\cap\| R_{\pi_2}$ etc., and $u \in \|[\pi]\varphi\|$ iff $(u, \|\varphi\|) \in R_a$. For a monotonic power model $\mathfrak{M} = (W, R, V)$ and $u \in W$ we shall also write $\mathfrak{M}, u \vDash \varphi$ for $u \in \|\varphi\|$. Since semantic interpretations are always defined relative to a model, if necessary we shall use the notation $\|-\|_{\mathfrak{M}}$ rather than $\|-\|$ to make it clear which model $\mathfrak{M}$ is being referred to. We write $\vDash \varphi$ if $\mathfrak{M}, u \vDash \varphi$ for every pointed monotone power model $(\mathfrak{M}, u)$. We get the following result, showing in what sense IPDL indeed generalizes the semantics of GL:

Proposition 8. *For any GL-formula φ, and any monotonic power model $\mathfrak{M}$, we have $\|\varphi\|_{\mathfrak{M}} = [\![\varphi]\!]_{\mathfrak{M}}$.*

From this proposition, we get the following result:

Theorem 7. *IPDL is a conservative extension of GL. That is, for every GL-formula φ, we have*

$$\vDash \varphi \text{ iff } \Vdash \varphi$$

In other words: the formulas of IPDL that are valid on arbitrary neighborhood frames form a conservative extension of the GL-formulas that are valid over monotonic power frames.

7 Concluding remarks

We have explored a propositional dynamic logic defined over instantial neighborhood logic. A language extension that is clearly related to the framework of this paper is the addition to the base language of least and greatest fixpoint operators, which for standard modal logic results in the *modal μ-calculus*. It is well known that PDL can be viewed as a fragment of the modal μ-calculus. In fact, our logic IPDL can also be translated into the analogous extension of INL with fixpoints. The translation is not straightforward though, and in fact the best translation we have found so far even causes an exponential blowup in formula size. We have omitted this material here due to lack of space. The fixpoint extension of INL is a very well behaved language: as shown in [3], INL is a coalgebraic modal logic corresponding to a weak pullback preserving functor - the double covariant powerset functor - that additionally preserves finite sets. (This should be contrasted with the monotone neighborhood functor, which is the appropriate functor for monotone modal logic and is known *not* to preserve weak pullbacks - see [12]. The monotone neighborhood functor is not suitable for INL since INL-formulas are not invariant for the behavioural equivalence associated with this functor.) This means that the μ-calculus extension of INL will inherit a number properties that hold in much wider generality: the language has the finite model property and is decidable [15], a sound and complete system of axioms is available [8] and the uniform interpolation property holds

[11]. Note however that it does *not* mean that we obtain our completeness result (and hence decidability and finite model property) for free, since completeness for fragments of modal μ-calculi does not generally follow easily from completeness of the full languages. Witnessing examples are Reynold's highly non-trivial completeness proof for CTL* [14] (which is a fragment of the μ-calculus [7]), or Parikh's game logic, which still lacks a complete system of axioms.

There is a growing body of work on PDL-like coalgebraic logics, with generic results on axiomatizability, see for example [9]. This setting is clearly related to the present work, however our system IPDL is not covered by this framework as it stands: while the covariant powerset functor is a monad, the *double* covariant powerset functor is not, which would be a requirement for existing work on coalgebraic PDL-logics to readily apply[2]. Perhaps the framework can be modified to capture IPDL as an instance – we offer this as a challenge and an interesting direction for future research.

References

[1] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM (JACM)*, 49(5):672–713, 2002.

[2] J. van Benthem. *Logic in games.* MIT Press, Cambridge, MA, 2014.

[3] J.van Benthem, N. Bezhanishvili, S. Enqvist, and J. Yu. Instantial neighborhood logic. *The Review of Symbolic Logic*, 10(1):116–144, 2017.

[4] D. Berwanger. Game logic is strong enough for parity games. *Studia Logica*, 75(2):205–219, 2003.

[5] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic.* Number 53 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[6] F. Carreiro and Y. Venema. PDL inside the μ-calculus: A syntactic and an automata-theoretic characterization. *Advances in Modal Logic*, 10:74–93, 2014.

[7] M. Dam. CTL* and ECTL* as fragments of the modal μ-calculus. *Theoretical Computer Science*, 126(1):77–96, 1994.

[8] S. Enqvist, F. Seifan, and Y. Venema. Completeness for coalgebraic fixpoint logic. In *Proceedings of the 25th EACSL Annual Conference on Computer Science Logic (CSL 2016)*, volume 62 of *LIPIcs*, pages 7:1–7:19, 2016.

[9] H.H. Hansen and C. Kupke. Weak completeness of coalgebraic dynamic logics. *arXiv preprint arXiv:1509.03017*, 2015.

[2] We are thankful to Helle Hansen for pointing this out to us.

[10] D. Kozen. Results on the propositional μ-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[11] J. Marti, F. Seifan, and Y. Venema. Uniform interpolation for coalgebraic fixpoint logic. In *Proceedings of the Sixth Conference on Algebra and Coalgebra in Computer Science (CALCO 2015)*, pages 238–252, 2015.

[12] Johannes Marti and Yde Venema. Lax extensions of coalgebra functors. In *International Workshop on Coalgebraic Methods in Computer Science*, pages 150–169. Springer, 2012.

[13] R. Parikh. The logic of games and its applications. *Annals of Discrete Mathematics*, 24:111–139, 1985.

[14] M. Reynolds. An axiomatization of full computation tree logic. *Journal of Symbolic Logic*, pages 1011–1057, 2001.

[15] Y. Venema. Automata and fixed point logic: a coalgebraic perspective. *Information and Computation*, 204:637–678, 2006.