Chapter 8
μ-Levels of Interpolation

Giovanna D'Agostino

Abstract In this paper we discuss the problem of interpolation in the alternation levels of the μ-Calculus. In particular, we consider interpolation and uniform interpolation for the alternation free fragment, and, more generally, for the level Δ_n of the alternation hierarchy of the μ-calculus.

8.1 Introduction

Interpolation is a desirable property for a logic. In very general terms it states that if a property is a consequence of another one, then only the common language between the two properties is important in order to have this consequence. In this way interpolation connects syntax with semantics, and, especially in its uniform version, it allows modularization. Larisa Maksimova has been a pioneer in this area, writing many papers and books on interpolation for classical and non classical logics. Here we consider the μ-Calculus, a very expressive and well known formalism extending Modal Logic with fixed points of monotone operators; in particular, we consider the stratification of μ-formulas defined by means of the alternation of least and greatest fixed point operators. This alternation is a major cause of complexity for μ-formulas, and many logics used in applications sit in the lower levels of the hierarchy. In particular, we consider the fragment of the μ-Calculus where formulas do not contain *real* alternations between least and greatest fixed point operators: the Alternation Free Fragment AF. Formulas in this fragment behave better from a computational point of view than formulas allowing alternation, and hence this fragment has been quite considered in the literature on the μ-Calculus (see Gutierrez et al. 2014 and Facchini et al. 2013 for recent results on AF).

G. D'Agostino (✉)
Department of Mathematics and Computer Science, Via delle Scienze 208,
33100 Udine, Italy
e-mail: giovanna.dagostino@uniud.it

S. Odintsov (ed.), *Larisa Maksimova on Implication, Interpolation, and Definability*,
Outstanding Contributions to Logic 15, https://doi.org/10.1007/978-3-319-69917-2_8

Another theme of this paper is automata which have been used with great success in studying the μ-Calculus, helping to prove completeness and complexity results and the like. As for interpolation, D'Agostino and Hollenberg (2000) shows how to construct the uniform interpolant of a formula starting from an automaton equivalent to the formula. Uniform interpolation via automata has been studied also in (Lutz et al. 2012) in the context of Descriptive Logics. In this paper we use automata to obtain the ordinary interpolant of a valid implication between alternation free formulas. This is the maximum we could hope, since we also prove that the alternation free fragment does not enjoy the uniform version of interpolation.

The paper is structured as follows: in Sect. 8.2 we recall the definitions of μ-formulas, alternating modal automata and levels of the Alternation Hierarchy. In Sect. 8.3 we summarize the results we already know about interpolation and levels, and finally in Sects. 8.4 and 8.5 we give the interpolation result for AF.

8.2 μ-Calculus and Automata

We consider the μ-formulas as constructed from a set of propositional constants $Prop$, their negations $\{\neg P : P \in Prop\}$, and a set of variables Var, using the operators: $\varphi_1 \vee \varphi_2$, $\varphi_1 \wedge \varphi_2$, $\Box(\varphi_1)$, $\Diamond(\varphi)$, $\mu X\varphi$, and $\nu X\varphi$, where $\varphi_1, \varphi_2, \varphi \in \mu$ and X in Var.

The language $\mathscr{L}(\varphi)$ of a formula φ is the set of propositional constants appearing in it.

The semantics of μ-formulas is defined as usual over pointed labelled graphs (e.g. Kripke models): given a graph $G = (V, R, L)$ (with $L : Prop \cup Var \to Pow(V)$), a μ-formula φ is interpreted as a subset $[\![\varphi]\!]_G$ of V, defined as follows:

$$
\begin{array}{lll}
[\![P]\!]_G & := L(P) & \text{for } P \in \mathscr{P}; \\
[\![\neg P]\!]_G & := V \setminus L(P) & \text{for } P \in \mathscr{P}; \\
[\![X]\!]_G & := L(X) & \text{for } X \in \mathscr{V}ar; \\
[\![\varphi \wedge \psi]\!]_G & := [\![\varphi]\!]_G \cap [\![\psi]\!]_G; & \\
[\![\Diamond\varphi]\!]_G & := \{s \in V \mid [\![\varphi]\!]_G \cap \{t : sRt\} \neq \emptyset\}; & \\
[\![\Box\varphi]\!]_G & := \{s \in V \mid [\![\varphi]\!]_G \supseteq \{t : sRt\}\}; & \\
[\![\mu X.\varphi]\!]_G & := \bigcap\{S \subseteq V \mid [\![\varphi]\!]_{G[X:=S]} \subseteq S\}; & \\
[\![\nu X.\varphi]\!]_G & := \bigcup\{S \subseteq V \mid [\![\varphi]\!]_{G[X:=S]} \supseteq S\}; &
\end{array}
$$

where $G,[X := S]$ is equal to G except that $L(X) = S$. Note that $[\![\mu X.\varphi]\!]_G$ is the least fixpoint of the monotone operator $S \mapsto [\![\varphi]\!]_{G[X:=S]}$, and $[\![\nu X.\varphi]\!]_G$ is the greatest fixpoint.

In the following, we denote $s \in [\![\varphi]\!]_G$ by $G, s \models \varphi$. If the graph G is a tree T, then by $T \models \varphi$ we mean $T, r \models \varphi$ where r is the root of T. Since μ formulas are bisimulation invariant, we may restrict the semantics to the class of trees.

8.2.1 Automata

In this section we define a class of automata running on trees that correspond to μ-formulas.

Definition 1 An alternating modal automaton over a finite alphabet A is a tuple

$$\mathscr{A} = \langle Q, q_0, \delta, \Omega \rangle,$$

such that:

- Q is a finite set of states;
- $q_0 \in Q$ is the initial state;
- Ω is a function from Q to the natural numbers;
- δ is a function which associates to every $q \in Q$ and $\sigma \in A$ a positive modal formula over Q.

If we want to stress the alphabet of the automaton, we will write it inside the tuple:

$$\mathscr{A} = \langle Q, A, q_0, \delta, \Omega \rangle.$$

An alternating automaton is said to be *non deterministic* if for all q, a there exists $X \subseteq Pow(Q)$ such that

$$\delta(q, a) = \bigvee_{J \in X} \left(\bigwedge_{q \in J} \Diamond(q) \wedge \Box \left(\bigvee_{q \in J} q \right) \right).$$

The notion of acceptance of a A-labeled tree $\mathscr{T} = (T, \lambda)$ by an automaton

$$\mathscr{A} = \langle Q, q_0, \delta, \Omega \rangle$$

is defined in terms of a two players game, as follows. Positions are of the form (t, F), where $F \in Mod^+(Q)$ (the positive modal formulas with variables in Q). The initial position is (t_0, q_0), where t_0 is the root of T. In a position (t, F) with $F \in Mod^+(Q)$ the play proceeds as follows:

1. if $F = \Diamond(G)$ $(F = \Box(G))$ then it is Player I's turn (Player II, respectively); he choses a son t' of t in T, and goes to position (t', G);
2. if $F = \bigvee_i G_i$ $(F = \bigwedge_i G_i)$ then it is Player I's turn (Player II, respectively); he chooses one disjunct G_i (conjunct, respectively) and goes to position (t, G);
3. if $F = q$, the game starts again from position $(t, \delta(q, \lambda(t)))$.

If a Player cannot move, it looses. Otherwise an infinite play is generated, which goes through infinite positions in $T \times Q$:

$$(t_0, q_0) \ldots (t_1, q_1) \ldots$$

The play is won by Player I if $limsup_i\ \Omega(q_i)$ is even, or, in other words, if the maximum among all $\Omega(q_i)$, for q_i appearing infinitely often along the play, is even.

A strategy for Player I is a function σ having as domain the set of partial plays π ending in a position for Player I, and such that $\sigma(\pi)$ is a legal move for Player I. The strategy σ is winning if Player I wins any play in which he follows σ. Similarly, we can define (winning) strategies for Player II. A tree is accepted by the automaton $\mathscr{A}$ if Player I has a winning strategy in the game of $\mathscr{A}$ over $\mathscr{T}$.

When dealing with non deterministic automata, we shall represent the transition function as a set of subsets of the set of states: $\delta(q, \lambda) = \{J_1, \ldots, J_k\}$ with $J_i \in Pow(Q)$; an acceptance game from a position (q, t) will then proceeds as follows: first Player I chooses a $J \in \delta(q, \lambda)$, then Player II chooses either a conjunct $\Diamond(q')$ with $q' \in J$ or $\Box(\bigvee_{q \in J} q)$; in the first case, Player I chooses a son t' of t and the play starts again from (q', t'); in the second case, Player II chooses a son t' of t, Player I must choose a $q' \in J$ and the play starts again from (q', t').

A well known result states:

Theorem 1 (Arnold and Niwinski 2001) *The class of alternating automata and the class of non deterministic automata accepts the same tree languages. Moreover, if the alphabet A is equal to $Pow(Prop)$ for a finite set of proposition $Prop$, then automata (alternating or non deterministic) have the same expressive power as the μ-sentences over $Prop$.*

In view of the previous result we shall confuse automata and formulas and write e.g. that a certain implication between automata is valid: $\models \mathscr{A} \rightarrow \mathscr{B}$ or has an interpolant (see Definition 4).

We shall use the following construction of the dual automaton.

Definition 2 The dual automaton $\tilde{\mathscr{A}}$ of the alternating automaton $\mathscr{A}$ is defined as follows.

1. the transition function of $\tilde{\mathscr{A}}$ is dualized: conjunctions in the $\mathscr{A}$-transitions become disjunctions in $\tilde{\mathscr{A}}$, disjunctions become conjunctions, diamonds become boxes and boxes become diamonds, while atomic propositions $q \in Q$ remain the same;
2. the set of states, the initial state, and the alphabet of the dual automaton are the same as the ones in $\mathscr{A}$;
3. the priority function of $\tilde{\mathscr{A}}$ is obtained from the one in $\mathscr{A}$ by adding 1.

Lemma 1 *The language recognized by the dual automaton $\tilde{\mathscr{A}}$ of an alternating automaton $\mathscr{A}$ is the complement of the language recognized by $\mathscr{A}$.*

8.2.2 Alternation Hierarchy

μ-formulas are divided into levels, depending on the alternation of fixed point operators inside the formula:

Definition 3 The *fixpoint alternation-depth hierarchy* of the μ-calculus is the sequence $\Pi_0 = \Sigma_0, \Pi_1, \Sigma_1, \ldots$ of sets of μ-formulas defined inductively as follows.

1. $\Pi_0 = \Sigma_0$ is defined as the set of all modal fixpoint free formulas.
2. Π_{k+1} is the closure of $\Pi_k \cup \Sigma_k$ under the operations described in (a), (b) below.
   (a) Positive Substitution: if $\varphi(P_1, \ldots P_n), \varphi_1, \ldots, \varphi_n$ are in Π_{k+1}, then $\varphi(\varphi_1 \ldots \varphi_n)$ is in Π_{k+1}, provided $P_1, \ldots, P_n$ are positive in φ and no occurrence of a variable which was free in one of the φ_i becomes bound in $\varphi(\varphi_1 \ldots \varphi_n)$.
   (b) If φ is in Π_{k+1}, then $\nu X.\varphi \in \Pi_{k+1}$.
3. Likewise, Σ_{k+1} is the closure of $\Pi_k \cup \Sigma_k$ under positive substitution and the μ-operator.

The previous hierarchy is defined at a syntactical level. In the following we will always consider the levels of the alternation hierarchy modulo semantics, i.e. we shall say that a formula φ belongs to the class Σ_n if it is semantically equivalent to a formula in this class. For each n we define the ambiguous class Δ_n as the set of μ formulas which are equivalent to both a formula in Σ_n and a formula in Π_n, while the class $Comp(\Sigma_n, \Pi_n)$ is defined as the least class closed under positive substitutions and containing Σ_n, Π_n.

We shall use the well known result that $\Delta_2 = Comp(\Sigma_1, \Pi_1)$ (the equality $\Delta_n = Comp(\Sigma_n, \Pi_n)$ is false for $n > 2$).

8.2.2.1 Levels and Automata

Levels of the Alternation Hierarchy are characterizable by automata:

Theorem 2 (Arnold and Niwinski 2001)

1. *A μ-formula φ is equivalent to a formula in the class Π_n if and only if there exists an alternating modal automaton which is equivalent to φ and an even $m \geq n - 1$ such that the range of the priority function of the automaton is contained in $\{m - n + 1, \ldots, m\}$.*
2. *If $n = 0, 1, 2$, then a μ-formula φ is equivalent to a formula in the class Π_n if and only if there exists a* non deterministic *modal automaton which is equivalent to φ and an even $m \geq n - 1$ such that the range of the priority function of the automaton is contained in $\{m - n + 1, \ldots, m\}$. The same result does not hold for $n > 2$.*
3. *A formula φ is equivalent to a formula in the class $Comp(\Sigma_n, \Pi_n)$ if and only if there exists an alternating modal automaton $\mathscr{A} = \langle Q, q_0, \delta, \Omega\rangle$ which is equivalent to φ and a preorder $\preceq$ of the set of states of the automaton such that, if $Q_0, \ldots, Q_k$ are the equivalence classes induced by the preorder, it holds:*
   a. *if $q \in Q_h$ then $\delta(q, a) \in Mod^+(\bigcup_{i \in \{h,\ldots,k\}} Q_i)$, for all $a \in A$;*

b. *for all* $i \in \{0, \ldots, k\}$ *there exists* $m \geq n - 1$ *such that the range of the priority function restricted to an equivalence class* Q_i *is either contained in* $\{m - n + 1, \ldots, m\}$ *or in* $\{m - n, \ldots, m - 1\}$.

In view of the equivalence between automata and μ-formulas we shall be free to say, e.g., that an automaton is in the class Π_n (meaning that it is equivalent to a formula in this class).

8.3 Interpolation and Uniform Interpolation

Let us begin by stating what we mean by ordinary (Craig) interpolation and uniform interpolation.

Definition 4 Let φ and ψ be two μ-sentences such that $\models \varphi \to \psi$. Then θ is an *interpolant* of φ, ψ iff:

1. $\models \varphi \to \theta$ and $\models \theta \to \psi$;
2. $\mathcal{L}(\theta) \subseteq \mathcal{L}(\varphi) \cap \mathcal{L}(\psi)$. □

In words: if $\models \varphi \to \psi$, an interpolant of φ, ψ is a formula in the common language of φ and ψ which sits in between φ and ψ.

Definition 5 Given a μ-sentence φ and a language $\mathcal{L}' \subseteq \mathcal{L}(\varphi)$, the *uniform interpolant* of φ with respect to $\mathcal{L}'$ is a formula θ such that:

1. $\models \varphi \to \theta$;
2. Whenever $\models \varphi \to \psi$ and $\mathcal{L}(\varphi) \cap \mathcal{L}(\psi) \subseteq \mathcal{L}'$ then $\models \theta \to \psi$.
3. $\mathcal{L}(\theta) \subseteq \mathcal{L}'$. □

When we say that the μ-calculus has (uniform) interpolation we mean that we can always find a (uniform) interpolant when the appropriate conditions are satisfied. Clearly, if the μ-calculus has uniform interpolation, it also enjoys Craig interpolation. For if $\models \varphi \to \psi$, simply choose $\mathcal{L}' = \mathcal{L}(\varphi) \cap \mathcal{L}(\psi)$. The interpolant is then the uniform interpolant of φ relative to $\mathcal{L}'$. This explains why we call this formula a *uniform* interpolant: no information is needed about the formula ψ except which non-logical symbols it has in common with φ.

In (D'Agostino and Hollenberg 2000) the uniform interpolation property for the μ-Calculus was proved using non deterministic automata: given a μ-formula φ, a propositional constant $p \in \mathcal{L}(\varphi)$, and a non deterministic automaton $\mathcal{A} = (Q, q_0, \delta, \Omega)$ over the alphabet $Pow(\mathcal{L}(\varphi))$ equivalent to φ, the *Uniform Interpolant Automaton* $UI(\mathcal{A})$ in the alphabet $Pow(\mathcal{L}(\varphi)) \setminus \{p\})$ is defined as follows:

$$UI(\mathcal{A}) := (Q, q_0, \delta', \Omega)$$

with:

$$\delta'(q, \lambda) := \delta(q, \lambda) \cup \delta(q, \lambda \cup \{p\})$$

In (D'Agostino and Hollenberg 2000) it is proved that $UI(\mathscr{A})$ corresponds to a μ-sentence ψ in the language $\mathscr{L}(\varphi) \setminus \{p\}$ which is a uniform interpolant for φ w.r.t. the language $\mathscr{L}(\varphi) \setminus \{p\}$; by iterating this construction we may obtain all uniform interpolants of a formula.

Uniform interpolation also holds for the levels $\Pi_0, \Sigma_1, \Pi_1, \Pi_2$: here, to construct the uniform interpolant, we may use the same automaton $UI(\mathscr{A})$ as above, simply because these levels have the property that an automaton corresponding to a formula in the level is equivalent to a nondeterministic automaton in the same class.

However, uniform interpolation does not hold if we restrict to other levels of the Alternation Hierarchy: in (D'Agostino and Lenzi 2006) it is proved that, except for $\Pi_0, \Sigma_1, \Pi_1, \Pi_2$, the levels of the Alternation Hierarchy are not closed under uniform interpolation:

Lemma 2 (D'Agostino and Lenzi 2006) *The levels of the Alternation Hierarchy are not closed under uniform interpolation, except for $\Pi_0, \Sigma_1, \Pi_1, \Pi_2$. Among uniform interpolants of Σ_2-formulas we may find formulas of arbitrarily high alternations.*

Since $\Sigma_2 \subseteq \Delta_3$, the previous lemma implies that for $n \geq 3$ the level Δ_n does not have the uniform interpolation property. However, this result leaves open the question of whether Δ_2, that is, the Alternation Free Fragment AF, enjoys uniform Interpolation or Craig interpolation, and this is the content of the following section.

8.4 Alternation Free Fragment and Uniform/Craig Interpolation

In this section we investigate uniform/Craig interpolation for the Alternation Free Fragment AF of the μ-calculus, which is defined as $Comp(\Pi_1, \Sigma_1)$. A well known result about the alternation free fragment gives a characterization of AF as the intersection of levels Π_2, Σ_2. This fragment has many interesting properties, e.g. it allows less complex decision procedure for model checking than the full μ-calculus, still maintaining strong expressive power.

In this section we prove that AF does not enjoy the uniform interpolation property, while retaining the Craig Interpolation one.

8.4.1 Failure of Uniform Interpolation in AF

Lemma 3 *There exists a formula $\varphi(p) \in AF$ without a uniform interpolant in AF w.r.t. $\mathscr{L}(\varphi) \setminus \{p\}$.*

Proof Consider a formula $\psi \in \Delta_2$ such that $\nu X\psi \notin \Delta_2$. We claim that the uniform interpolant of the formula $\varphi(p) := p \wedge \Box^*(p \to \psi(p|X))$, w.r.t. the language

$\mathscr{L}(\varphi) \setminus \{p\}$, where $\Box^*(p) := \nu Y(p \wedge \Box(Y))$, is equivalent to $\nu X\psi$. To show this it suffices to prove:

1. $\models \varphi(p) \to \nu X\psi$;
2. if θ is a formula not containing p and $\models \varphi \to \theta$, then $\models \nu X\psi \to \theta$.

Suppose (G, w) is a model with $(G, w) \models \varphi(p)$; then, by definition of φ, the set of points $\{v \in V_G : (G, v) \models \varphi(p)\}$ is a post-fixed point of ψ and contains w. By Tarski–Knaster Theorem it follows that $(G, w) \models \nu X\psi$.

As for the second point, suppose θ does not contain p, $\models \varphi \to \theta$, and (G, w) is a model such that $(G, w) \models \nu X\psi$. Then, if $A := \{v \in V_G : (G, v) \models \nu X\psi\}$ we have $\psi(A) = A$, and $(G, p := A, w) \models p \wedge \Box^*(p \to \psi(p|X))$, that is, $(G, p := A, w) \models \varphi$. From $\models \varphi \to \theta$ it follows $(G, p := A, w) \models \theta$, and we obtain $(G, w) \models \theta$, since p is not contained in θ.

Finally, we notice that the formula $\varphi(p) := p \wedge \Box^*(p \to \psi(p|X))$ belongs to $Comp(\Pi_1, \Sigma_1) = \Delta_2$.

8.4.2 Craig Interpolation for AF

In this section we prove the Craig Interpolation property for the Alternation Free Fragment using automata. More precisely, we obtain the interpolation property for AF by adapting the proof of a separation result from (Santocanale and Arnold 2005). In this paper it is proved that given two Π_{n+1} non deterministic automata $\mathscr{A}, \mathscr{B}$ with no common models, there exists an automaton $\mathscr{C}$ in the class $Comp(\Pi_n, \Sigma_n)$ such that $\models \mathscr{A} \to \mathscr{C}$ and $\mathscr{C}, \mathscr{B}$ have no common models, where all automata $\mathscr{A}, \mathscr{B}, \mathscr{C}$ are over the same alphabet. We consider this result for $n = 1$, but we have to modify it in order to consider as input two non deterministic automata $\mathscr{A}, \mathscr{B}$ in Π_2 over the alphabets $A_1 = Pow(P_1)$, $A_2 = Pow(P_2)$, respectively, and obtain as output an automaton $\mathscr{C}$ in the class $Comp(\Pi_1, \Sigma_1)$ over the alphabet $A_1 \cap A_2$:

Lemma 4 (Santocanale and Arnold 2005) *If $\mathscr{A}, \mathscr{B}$ are non deterministic Π_2 automata in the alphabets A_1, A_2, respectively, such that $L(\mathscr{A}) \cap L(\mathscr{B}) = \emptyset$, there exists an automaton $\mathscr{C}$ in the class $Comp(\Pi_1, \Sigma_1)$ over the alphabet $A_1 \cap A_2$ with $\models \mathscr{A} \to \mathscr{C}$ and $\models \mathscr{C} \to \tilde{\mathscr{B}}$, for the dual automaton $\tilde{\mathscr{B}}$ of $\mathscr{B}$.*

We give a proof of this Lemma in the next section, but first we prove Craig Interpolation from it.

Theorem 3 *Craig Interpolation for AF*.
The Alternation Free Fragment of the modal μ-Calculus enjoys the Craig Interpolation Property.

Proof Suppose $\models \varphi \to \psi$ with $\varphi, \psi \in AF$. Since AF is closed under negation and any automaton in Π_2 is equivalent to a non deterministic automaton in the same class, both $\varphi, \neg\psi$ are equivalent to non deterministic automata $\mathscr{A}, \mathscr{B}$ in Π_2, for which the previous lemma applies. Hence, there exists an automaton $\mathscr{C}$ in Δ_2 over the alphabet $Pow(\mathscr{L}(\varphi) \cap \mathscr{L}(\psi))$, with $\models \mathscr{A} \to \mathscr{C}$ and $\models \mathscr{C} \to \tilde{\mathscr{B}}$, for the dual automaton $\tilde{\mathscr{B}}$ of $\mathscr{B}$. If θ is the formula which is equivalent to the automaton $\mathscr{C}$, then θ is an AF formula in $\mathscr{L}(\varphi) \cap \mathscr{L}(\psi)$ with $\models \varphi \to \theta$ and $\models \theta \to \psi$, that is, $\theta \in AF$ is the required interpolant between $\varphi, \psi \in AF$.

8.5 The Interpolation Automaton

Given two non deterministic automata $\mathscr{A}, \mathscr{B}$ in Π_2 (over the alphabets $A_1 = Pow(P_1)$, $A_2 = Pow(P_2)$, respectively) such that $L(\mathscr{A}) \cap L(\mathscr{B}) = \varnothing$, we will define the interpolation automaton $INT(\mathscr{A}, \tilde{\mathscr{B}})$ between $\mathscr{A}$ and the dual $\tilde{\mathscr{B}}$ of $\mathscr{B}$ using $\mathscr{A}, \mathscr{B}$ and a winning strategy for Player II in the satisfiability game defined below.

Definition 6 (Niwinski and Walukiewicz 1996; Santocanale and Arnold 2005) Let $\mathscr{A} = (Q, q_0, \delta_{\mathscr{A}}, \Omega_{\mathscr{A}})$ and $\mathscr{B} = (R, r_0, \delta_{\mathscr{B}}, \Omega_{\mathscr{B}})$ be non deterministic modal automata over the same alphabet A. The satisfiability game $\mathscr{G}(\mathscr{A}, \mathscr{B})$ is defined as follows:

1. Positions for Player I:

$$Pos_1 = (Q \times R) \cup \{(q, K) : q \in Q, K \in Pow(R)\} \cup \{(J, r) : r \in R, J \in Pow(Q)\};$$

2. Position for Player II:

$$Pos_2 = \{(J, K) : J \in Pow(Q), K \in Pow(R)\};$$

3. $Pos = Pos_1 \cup Pos_2$;
4. the initial position is $p_0 = (q_0, r_0)$.
5. from a position $(q, r) \in Q \times R$, Player I chooses $a \in A$ and move to $(J, K) \in \delta_{\mathscr{A}}(q, a) \times \delta_{\mathscr{B}}(r, a)$;
6. from a position $(J, K) \in Pow(Q) \times Pow(R)$, Player II can move either to (q, K) with $q \in J$ or to (J, r) with $r \in K$;
7. from a position (q, K) with $q \in Q$, $K \subseteq R$, Player I can move to (q, r) with $r \in K$; from a position (J, r) with $r \in R$, $J \subseteq Q$, Player I can move to (q, r) with $q \in J$.

If a Player cannot move, it looses. Player I wins a play $(q_0, r_0)(J_0, K_0) \ldots (q_1, r_1) \ldots$ if and only if both $limsup_i \Omega_{\mathscr{A}}(q_i)$ and $limsup_i \Omega_{\mathscr{B}}(r_i)$ are even.

Consider now two non deterministic modal automata $\mathscr{A}, \mathscr{B}$ such that $L(\mathscr{A}) \cap L(\mathscr{B}) = \varnothing$; by a result in (Niwinski and Walukiewicz 1996), Player II has a winning strategy in the satisfiability game $\mathscr{G}(\mathscr{A}, \mathscr{B})$. Notice that this kind of *biparity game* can be easily transformed into a Muller game, and since Muller games have finite memory winning strategies, it follows:

Lemma 5 *If $L(\mathscr{A}) \cap L(\mathscr{B}) = \varnothing$ then Player II has a finite memory winning strategy σ in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$*

$$\sigma = \begin{cases} nxt : Pos_2 \times M \to Pos; \\ updt : Pos \times M \to M; \\ init : Pos \to M, \end{cases}$$

(where M is a finite set, the memory), such that for all positions (q, r) reached by a play in which Player II follows σ it holds $L(\mathscr{A}, q) \cap L(\mathscr{B}, r) = \varnothing$.

The strategy of the above Lemma can be presented as a finite *strategy* graph $\mathscr{S}$ defined as follows. Consider the following binary relation on $Pos \times M$:

1. if $p \in Pos_1$, $p' \in Pos$ there is an edge from (p, m) to (p', m) iff $p \to p'$ is a possible move for Player I;
2. if $p \in Pos_2$ there is an edge from (p, m) to $(nxt(p, m), updt(p, m))$.

The finite strategy graph $\mathscr{S}$ is defined as the set of vertices and edges which are reachable from the vertex $(p_0, init(p_0))$, where p_0 is the initial position of the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$.

In order to find the interpolant between AF formulas, we concentrate now on non deterministic modal automata over power set alphabets of the form $A = Pow(P)$, where P is a finite set of *propositions*. If $P \subseteq P'$, where P' is another set of propositions, a $Pow(P)$ automaton $\mathscr{A}$ will be also considered as an automaton over $A' = Pow(P')$, by extending the transition function as follows, for $a' \in A'$:

$$\delta_{\mathscr{A}}(q, a') = \delta_{\mathscr{A}}(q, a' \cap P)$$

If (T, λ) is an A'-labelled tree, then it holds:

$$(T, \lambda) \text{ is accepted by (the extension of)} \mathscr{A} \Leftrightarrow (T, \lambda \cap P) \text{ is acceped by } \mathscr{A},$$

where $\lambda \cap P(t) := \lambda(t) \cap P$. In the following, we shall use this kind of extension freely, every time an automaton originally defined over an alphabet $A = Pow(P)$ is considered as an automaton over a bigger alphabet $A' = Pow(P')$, with $P \subseteq P'$.

Definition 7 Let $\mathscr{A} = (Q, q_0, \delta_{\mathscr{A}}, \Omega_{\mathscr{A}})$, $\mathscr{B} = (R, r_0, \delta_{\mathscr{B}}, \Omega_{\mathscr{B}})$ be two non deterministic Π_2 modal automata over the alphabets $A_1 = Pow(P_1)$, $A_2 = Pow(P_2)$, respectively, with $\Omega_{\mathscr{A}}(Q), \Omega_{\mathscr{B}}(R) \subseteq \{1, 2\}$ and $L(\mathscr{A}) \cap L(\mathscr{B}) = \varnothing$ (considering both as automata over the alphabet $A = Pow(P_1 \cup P_2)$). Then if $\mathscr{S}$, σ are the strategy graph and the winning strategy for Player II in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$, as defined above, the automaton $INT(\mathscr{A}, \tilde{\mathscr{B}}) = (S, s_0, \delta, \Omega)$ over the alphabet $A_1 \cap A_2 = Pow(P_1 \cap P_2)$ is defined as follows:

1. S is the vertex set of the strategy graph $\mathscr{S}$.
2. $s_0 = (p_0, init(p_0))$, where p_0 is the initial position of the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$;

3. if $a \in A_1 \cap A_2 = Pow(P_1 \cap P_2)$ then

$$\delta((q,r,m),a) = \bigvee_{\left\{\substack{a_1 \in A_1,\ a_1 \cap P_2 = a \\ J \in \delta_{\mathscr{A}}(q,a_1)}\right\}} \bigwedge_{\left\{\substack{a_2 \in A_2,\ a_2 \cap P_1 = a \\ K \in \delta_{\mathscr{B}}(r,a_2)}\right\}} F_{J,K,m},$$

where, if σ is defined as in Lemma 5, the formula $F_{J,K,m}$ is defined as

$$F_{J,K,m} = \begin{cases} \Diamond(\bigwedge_{r \in K}(q,r,updt(J,K,m))), & \text{if } nxt(J,K,m) = (q,K); \\ \Box(\bigvee_{q \in J}(q,r,updt(J,K,m))), & \text{if } nxt(J,K,m) = (J,r). \end{cases}$$

4. To define Ω, consider a strongly connected component $C(q,r,m)$ of the strategy graph $\mathscr{S}$. Then we consider the following cases:

   a. $|C(q,r,m)| = 1$;
   b. $|C(q,r,m)| > 1$ and $\forall (q',r',m') \in C(q,r,m)$ we have $\Omega_{\mathscr{A}}(q') = 1$;
   c. $|C(q,r,m)| > 1$ and $\exists (q',r',m') \in C(q,r,m)$ with $\Omega_{\mathscr{A}}(q') = 2$.

   Accordingly, we define:

$$\Omega(q,r,m) = \begin{cases} 1 \text{ if case (a) or (b) applies;} \\ 2 \text{ if case (c) applies.} \end{cases}$$

Consider the partial order $\preceq$ over the set of states S of the automaton $INT(\mathscr{A}, \tilde{\mathscr{B}})$ defined by: $s \preceq s'$ iff there is a path from s to s' in the strategy graph S; then we easily see that the automaton $INT(\mathscr{A}, \tilde{\mathscr{B}})$ defined above is in Δ_2 by using the preorder $\preceq$: if we are in a non trivial strongly connected component of the graph and there is a (q,r,m) in the component with $\Omega_{\mathscr{A}}(q) = 2$ then by definition the priority function is always 2 on the component, otherwise it is always 1.

We are now ready to prove:

Lemma 6 *Let $\mathscr{A}, \mathscr{B}$ be non deterministic Π_2 automata in the alphabets $A_1 = Pow(P_1)$, $A_2 = Pow(P_2)$, respectively, such that $L(\mathscr{A}) \cap L(\mathscr{B}) = \varnothing$.*
If $\mathscr{C} = INT(\mathscr{A}, \tilde{\mathscr{B}})$ is the automaton over $A_1 \cap A_2$ defined above we have:

$$\models \mathscr{A} \to \mathscr{C} \text{ and } \models \mathscr{C} \to \tilde{\mathscr{B}}$$

where $\tilde{\mathscr{B}}$ is the dual automaton of $\mathscr{B}$.

Proof We first prove that every A_1-labeled tree (T, λ) which is accepted by $\mathscr{A}$, is accepted by $\mathscr{C}$ (i.e. $\mathscr{C}$ accepts the tree $(T, \lambda \cap P_2)$ where $\lambda \cap P_2(t) = \lambda(t) \cap P_2$). If (T, λ) is accepted by $\mathscr{A}$, let τ be a winning strategy for Player I in the game of $\mathscr{A}$ over T. Using τ and a winning strategy σ for Player II in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$ we shall define a strategy η for Player I in the game of $INT(\mathscr{A}, \tilde{\mathscr{B}})$ over $(T, \lambda \cap P_2)$, and prove it is a winning one.

Claim We claim that a strategy η can be defined in such a way to preserve the following invariant: any η-play of $\mathscr{C}$ over $(T, \lambda \cap P_2)$,

$$(q_0, r_0, m_0, t_0), \ldots, (q_1, r_1, m_1, t_1), \ldots, (q_i, r_i, m_i, t_i) \ldots,$$

leaves a trace

$$(q_0, t_0), \ldots, (q_1, t_1), \ldots, (q_i, t_i) \ldots$$

which is a τ-play of $\mathscr{A}$ over (T, λ), and a trace

$$(q_0, r_0, m_0), \ldots, (q_1, r_1, m_1), \ldots, (q_i, r_i, m_i) \ldots,$$

which is a σ-play in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$.

The $\mathscr{C}$-play over $(T, \lambda \cap P_2)$ starts form the initial position (q_0, r_0, m_0, t_0), where t_0 is the root of T, while the play of $\mathscr{A}$ over (T, λ) starts from the initial position (q_0, t_0), and the play of the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$ starts from (q_0, r_0, m_0). From the position (q_0, t_0) of the $\mathscr{A}$ play, the strategy τ suggests to Player I a move $J \in \delta_{\mathscr{A}}(q_0, \lambda(t_0))$; since $\lambda(t_0) \cap P_2 := \lambda \cap P_2(t_0)$, this J is also a possible move for Player I in the game of $\mathscr{C}$ over $(T, \lambda \cap P_2)$; hence we define $\eta(q_0, r_0, m_0, t_0) = J$. In the $\mathscr{C}$-play it is now Player II's turn, and he can choose a $K \in \delta_{\mathscr{B}}(r_0, a_2)$, with $a_2 \in A_2$ and $a_2 \cap P_1 = \lambda(t_0) \cap P_2$. At this point we shift to the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$, starting from (q_0, r_0, m_0), where Player I can choose to move to position (J, K, m_0), which is a legal position for him. Since (J, K, m_0) is a position for Player II in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$, we may use the σ strategy. We consider two cases:

1. $nxt((J, K), m_0) = (q_1, K)$, with $q_1 \in J$. In this case

$$F_{J,K,m_0} = \Diamond\left(\bigwedge_{r \in K} (q_1, r, updt((J, K), m_0))\right),$$

   and it is Player I's turn in the $\mathscr{C}$-play; to decide the value of the strategy η, we consider the $\mathscr{A}$ play, where now in position J it is Player II's turn; since $q_1 \in J$, the conjunct $(\Diamond(q_1), t_0)$ is a legal move for Player II in the $\mathscr{A}$ play; remaining in this play the winning strategy τ will suggest a son t_1 of t_0, and Player I will move to the position (q_1, t_1). We choose this t_1 also to be the suggestion that the strategy η gives to Player I in the $\mathscr{C}$-play over $(T, \lambda \cap P_2)$, and move to the position $(\bigwedge_{r \in K}(q_1, r, m_1), t_1)$, where $m_1 = updt((J, K), m_0))$. Now it is Player II's turn in the $\mathscr{C}$-play, say he chooses $r_1 \in K$ and moves to the position (q_1, r_1, m_1, t_1). Since $r_1 \in K$, (q_1, r_1, m_1) is also a possible move for Player I in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$, and the invariant is satisfied.

2. $nxt((J,K),m_0)) = (J, r_1)$, with $r_1 \in K$. In this case

$$F_{J,K,m_0} = \Box\left(\bigvee_{q\in J}(q, r_1, updt((J,K),m_0))\right),$$

and it is Player II's turn in the $\mathscr{C}$-play: he will choose a son t_1 of t_0 in T and move to position $(\bigvee_{q\in J}(q, r_1, m_1), t_1)$, with $m_1 = updt((J,K),m_0)$. We consider again the $\mathscr{A}$ play, and let Player II choose the position $(\Box(\bigvee_{q\in J} q), t_0)$ in this play; from this position it is still Player II turn and he can choose position $(\bigvee_{q\in J} q, t_1)$, which is now a position for Player I. Using the strategy τ for Player I, we find $q_1 \in J$ so that (q_1, t_1) is a legal position in the τ-play over (T, λ); in the $\mathscr{C}$-play we choose this q_1 to be the suggestion that the strategy η gives to Player I in position $(\bigvee_{q\in J}(q, r_1, m_1), t_1)$ of the $\mathscr{C}$-play over $(T, \lambda \cap P_2)$, and move to (q_1, r_1, m_1, t_1); since $q_1 \in J$, (q_1, r_1, m_1) is also a legal move for Player I in the $\mathscr{G}(\mathscr{A}, \mathscr{B})$, and the invariant is satisfied.

Since the same pattern can be applied to a position (q_i, r_i, m_i, t_i) already reached by the η play, the claim is proved.

To conclude the proof of $\models \mathscr{A} \to \mathscr{C}$ we have to check that the strategy η described above is winning for Player I in the game of $\mathscr{C}$ over $(T, \lambda \cap P_2)$. By construction, any η-play

$$(q_0, r_0, m_0, t_0), \ldots, (q_1, r_1, m_1, t_1), \ldots, (q_i, r_i, m_i, t_i) \ldots,$$

leaves a trace

$$(q_0, t_0), \ldots, (q_1, t_1), \ldots, (q_i, t_i) \ldots$$

which is a τ-play of $\mathscr{A}$ over T, and a trace

$$(q_0, r_0, m_0), \ldots, (q_1, r_1, m_1), \ldots, (q_i, r_i, m_i) \ldots,$$

which is a σ-play in the game $\mathscr{G}(\mathscr{A}, \mathscr{B})$. Since τ is winning for I in the $\mathscr{A}$ play, $limsup_i(\Omega_{\mathscr{A}}(q_i)) = 2$. Moreover, the σ play

$$(q_0, r_0, m_0), \ldots, (q_1, r_1, m_1), \ldots, (q_i, r_i, m_i) \ldots,$$

will sooner or later get trapped in a non trivial strongly connected component $\mathscr{C}$ of the strategy graph $\mathscr{S}$. Hence, we must be in case (c) in the definition of Ω; hence, from a certain point on $\Omega(q_i, r_i, m_i) = 2$, and η is winning. This concludes the proof that $\models \mathscr{A} \to \mathscr{C}$.

Finally, to prove that $\models \mathscr{C} \to \tilde{\mathscr{B}}$, consider the pair of automata $(\mathscr{B}, \mathscr{A})$. This pair satisfies the hypothesis of Definition 7; by the previous part of the proof, we have $\models \mathscr{B} \to INT(\mathscr{B}, \tilde{\mathscr{A}})$, where $INT(\mathscr{B}, \tilde{\mathscr{A}}) = (S, (q_0, r_0, m_0), \delta', \Omega')$ and δ', Ω' are defined as follows:

1. if $a \in A_1 \cap A_2$ then

$$\delta'((q,r,m),a) = \bigvee_{\left\{\substack{a_2 \in A_2,\ a_2 \cap P_1 = a \\ K \in \delta_{\mathscr{B}}(r,a_2)}\right\}} \bigwedge_{\left\{\substack{a_1 \in A_1,\ a_1 \cap P_2 = a \\ J \in \delta_{\mathscr{A}}(q,a_1)}\right\}} G_{K,J,m},$$

where, if σ is as in Lemma 5, the formula $G_{J,K,m}$ is defined as

$$G_{J,K,m} = \begin{cases} \Diamond(\bigwedge_{q \in J}(q,r,updt((J,K),m))), & \text{if } \sigma(J,K,m) = (J,r,updt((J,K),m); \\ \Box(\bigvee_{r \in K}(q,r,updt((J,K),m))), & \text{if } \sigma(J,K,m) = (q,K,updt((J,K),m). \end{cases}$$

2.

$$\Omega'(q,r,m) = \begin{cases} 1 \text{ if case (a) or (b) applies;} \\ 2 \text{ if case or (c) applies,} \end{cases}$$

where

a. $|\mathscr{C}(q,r,m)| = 1$;
b. $|\mathscr{C}(q,r,m)| > 1$ and $\forall (q',r',m') \in \mathscr{C}(q,r,m)$ we have $\Omega_{\mathscr{B}}(r') = 1$;
c. $|\mathscr{C}(q,r,m)| > 1$ and $\exists (q',r',m') \in \mathscr{C}(q,r,m)$ with $\Omega_{\mathscr{B}}(r') = 2$.

From the first part of the proof we already know that $\models \mathscr{B} \to INT(\mathscr{B}, \tilde{\mathscr{A}})$ and hence $\models I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}}) \to \tilde{\mathscr{B}}$, for the dual automaton $I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ of $INT(\mathscr{B}, \tilde{\mathscr{A}})$, to conclude the proof of the Lemma we only have to show that $\models INT(\mathscr{A}, \tilde{\mathscr{B}}) \to I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$.

Notice that the transition function ε of $I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ is

$$\varepsilon((q,r,m),a) = \bigwedge_{\left\{\substack{a_2 \subseteq A_2,\ a_2 \cap A_1 = a \\ K \in \delta_{\mathscr{B}}(r,a_2)}\right\}} \bigvee_{\left\{\substack{a_1 \subseteq A_1,\ a_1 \cap A_2 = a \\ J \in \delta_{\mathscr{A}}(q,a_1)}\right\}} \tilde{G}_{K,J,m},$$

where

$$\tilde{G}_{J,K,m} = F_{J,K,m} = \begin{cases} \Box(\bigvee_{q \in J}(q,r,updt((J,K),m))), & \text{if } \sigma(J,K,m) = (J,r,updt((J,K),m); \\ \Diamond(\bigwedge_{r \in K}(q,r,updt((J,K),m))), & \text{if } \sigma(J,K,m) = (q,K,updt((J,K),m). \end{cases}$$

Calling δ the transition function of $INT(\mathscr{A}, \tilde{\mathscr{B}})$, for propositional reasoning one can easily check that for all $a \in A$ it holds:

$$\models \delta((q,r,m),a) \to \varepsilon((q,r,m),a). \tag{8.1}$$

Moreover, if Ω'' is the priority function of $I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ then

$$\Omega''(q,r,m) = \begin{cases} 2 \text{ if case (a) or (b) of } \Omega' \text{ definition applies;} \\ 3 \text{ if case or (c) of } \Omega' \text{ definition applies.} \end{cases}$$

Suppose that $INT(\mathscr{A}, \tilde{\mathscr{B}})$ accepts a tree, via a winning strategy η for Player I; then:

1. since $\models \delta((q, r, m), a) \rightarrow \varepsilon((q, r, m), a)$, the strategy η is also a possible strategy for Player I in the game of $I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ over the same tree;
2. sooner or later the η-play will get trapped in a strongly connected component $\mathscr{C}(q, r, m)$ of the strategy graph $\mathscr{S}$ such that both $|\mathscr{C}(q, r, m)| > 1$ and $\exists(q', r', m') \in \mathscr{C}(q, r, m)$ with $\Omega_{\mathscr{A}}(q') = 2$ hold. Since the strategy graph $\mathscr{S}$ is built using a winning strategy for Player II, we must have $\Omega_{\mathscr{B}}(r') = 1$, $\forall(q', r', m') \in C(q, r, m)$. Hence, from some point in the play the priority function Ω'' will always assume value 2.

This proves that η is also winning for Player I in the game of $I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ over the tree and hence $\models INT(\mathscr{A}, \mathscr{B}) \rightarrow I\tilde{N}T(\mathscr{B}, \tilde{\mathscr{A}})$ follows.

8.6 Conclusion

In this paper we considered the property of interpolation for levels of the μ-Calculus hierarchy. We proved that a uniform version of interpolation does not hold in levels Δ_n, for $n \geq 2$, and in particular uniform interpolation does not hold in the Alternation Free Fragment AF of the μ-Calculus. However, a Craig interpolant for a valid implication in AF can always be found in AF, and we provided a construction of the interpolant via automata. This construction uses non deterministic automata for AF-formulas, and it is not directly generalizable to the other ambiguous classes Δ_n with $n > 2$. Hence, the problem of the validity of ordinary interpolation in these levels is left open in this paper.

Acknowledgements This paper has been partially supported by the GNCS-INDAM Project 'Algoritmica per il model checking e la sintesi di sistemi safety-critica'.
The author wish to thank Michael T. Vanden Boom for pointing out various mistakes in a previous version of the paper, and the anonimous referees for being very patient and positive in spite of all these mistakes.

References

Arnold, A., & Niwinski, D. (2001). *Rudiments of the mu-Calculus*. Amsterdam: North Holland.

D'Agostino, G., & Hollenberg, M. (2000). Logical questions concerning the μ-calculus: Interpolation. Lyndon and Łoś-Tarski, Journal of Symbolic Logic, *65*(1), 310–332.

D'Agostino, G., & Lenzi, G. (2006). On modal mu-calculus with explicit interpolants. *Journal of Applied Logic*, *4*(3), 256–278.

Facchini, A., Venema, Y., & Zanasi, F. (2013). A characterization theorem for the alternation-free fragment of the modal μ-calculus. In *LICS '13 Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, pp. 478–487.

Gutierrez, J., & Klaedtke, F. (2014). The μ-calculus alternation hierarchy collapses over structures with restricted connectivity. *Theoretical Computer Science*, *560*, 292–306.

Lutz, C., Seylan, I., & Wolter, F. (2012). An automata-theoretic approach to uniform interpolation and approximation in the description logic EL. *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR*. Rome, Italy: AAAI Press.

Niwinski, D., & Walukiewicz, I. (1996). Games for the μ-calculus. *Theoretical Computer Science*, *163*(1–2), 99–116.

Santocanale, L., & Arnold, A. (2005). Ambiguous classes in μ-calculi hierarchies. *Theoretical Computer Science*, *333*(1–2), 265–296.