The Parametric Ordinal-Recursive Complexity of Post Embedding Problems*

Prateek Karandikar[1,2] and Sylvain Schmitz[2]

[1] CMI, Chennai, India
[2] LSV, ENS Cachan & CNRS, Cachan, France

Abstract. *Post Embedding Problems* are a family of decision problems based on the interaction of a rational relation with the subword embedding ordering, and are used in the literature to prove non multiply-recursive complexity lower bounds. We refine the construction of Chambart and Schnoebelen (LICS 2008) and prove parametric lower bounds depending on the size of the alphabet.

1 Introduction

Ordinal Recursive functions and subrecursive hierarchies [24, 12] are employed in computability theory, proof theory, Ramsey theory, rewriting theory, etc. as tools for bounding derivation sizes and other objects of very high combinatory complexity. A standard example is the ordinal-indexed *extended Grzegorczyk hierarchy* $\mathscr{F}_\alpha$ [21], which characterizes classical classes of functions: for instance, $\mathscr{F}_2$ is the class of elementary functions, $\bigcup_{k<\omega} \mathscr{F}_k$ of primitive-recursive ones, and $\bigcup_{k<\omega} \mathscr{F}_{\omega^k}$ of multiply-recursive ones. Similar tools are required for the classification of decision problems arising with verification algorithms and logics, prompting the investigation of "natural" decision problems complete for *fast-growing complexity* classes $\mathbf{F}_\alpha$ [14, 27].

Post Embedding Problems. (PEPs) have been introduced by Chambart and Schnoebelen [7] as a tool to prove the decidability of safety and termination problems in unreliable channel systems. The most classical instance of a PEP is called "regular" by Chambart and Schnoebelen [7], but we will follow Barceló et al. [4] and rather call it *rational* in this paper:

Rational Embedding Problem (EP[Rat])

Input. A rational relation R in $\Sigma^* \times \Sigma^*$.
Question. Is the relation $R \cap \sqsubseteq$ empty?

Here, the $\sqsubseteq$ relation denotes the *subword embedding* ordering, which relates two words w and w' if $w = c_1 \cdots c_n$ and $w' = w_0 c_1 w_1 \cdots w_n c_n w_{n+1}$ for some symbols

* Research partially funded by the ANR ReacHard project (ANR 11 BS02 001 01). The first author is partially funded by Tata Consultancy Services. Part of this research was conducted while the second author was visiting the Department of Computer Science at Oxford University thanks to a grant from the ESF *Games for Design and Verification* activity.

F. Pfenning (Ed.): FOSSACS 2013, LNCS 7794, pp. 273–288, 2013.

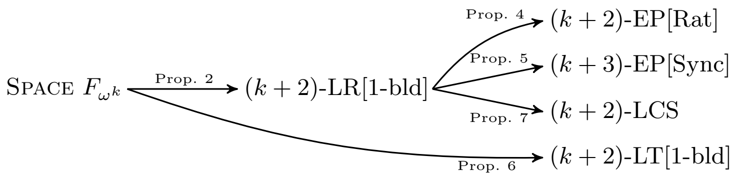

Fig. 1. Relationships between PEPs and similar decision problems

c_i in Σ and words w_i in Σ^*; in other words, w can be obtained from w' by "losing" some symbol occurrences (maybe none).

Although PEPs appear naturally in relation with channel systems [7, 8, 16] and queries on graph databases [4], their main interest lies in their use in lower bound proofs for other, sometimes seemingly distantly related problems [23, 19, 3]: in spite of their simple formulation, they are known to be of non multiply-recursive complexity in general. In fact, this motivation has been present from their inception in [7]: find a "master" decision problem complete for $\mathbf{F}_{\omega^\omega}$, the class of *hyper-Ackermannian* problems, solvable with non multiply-recursive complexity, but no less—much like SAT is often taken as the canonical NPTIME-complete problem, or the Post Correspondence Problem for Σ^0_1. This has also prompted a wealth of research into variants and related questions [10, 4, 18].

In this paper, we revisit and simplify the original proof of Chambart and Schnoebelen [9] that established the hardness of PEPs, and prove tight *parameterized* lower bounds when the size of the alphabet Σ is fixed. More precisely, we show that the $(k+2)$-rational embedding problem, i.e. the restriction of EP[Rat] to alphabets Σ of size at most $k+2$, is hard for $\mathbf{F}_{\omega^k}$ the class of *k-Ackermannian problems* if $k \geq 2$. As the problem can be shown to be in $\mathbf{F}_{\omega^{k+1}+1}$ [26, 18], we argue this to be a rather tight bound. The hyper-Ackermannian lower bound of $\mathbf{F}_{\omega^\omega}$ first proven by Chambart and Schnoebelen then arises when $|\Sigma|$ is not fixed but depends on the instance.

Our main tool to this end is another problem that involves a rational relation together with the subword embedding:

Lossy Rewriting (LR[Rat])

Input. A rational relation R in $\Sigma^* \times \Sigma^*$ and two words w and w' in Σ^*.
Question. Does (w, w') belong to the reflexive transitive closure $R_{\sqsupseteq}^{\circledast}$?

Here $R_{\sqsupseteq}$ denotes the "lossy version" of the relation R, defined formally as the composition $\sqsupseteq \fatsemi R \fatsemi \sqsupseteq$. We prove our lower bounds on this variant of EP[Rat] and then use them to prove lower bounds for EP[Rat] and other embedding problems; Fig. 1 summarizes the lower bounds presented in this paper. In a sense, LR is our own champion for the title of "master" problem for $\mathbf{F}_{\omega^\omega}$. Besides its rather simple statement, note that the related question of whether (w, w') belongs to $R^{\circledast}$ is undecidable by an easy reduction from the acceptance problem for Turing machines.

Overview. Technically, our results rely on an implementation of the computations for the *Hardy functions* $H^{\omega^{\omega^k}}$ and their inverses by successive applications of a relation with a fixed *bounded length discrepancy.* The main difficulty here is that this implementation should be *robust* for the symbol losses associated with the embedding relation. It requires in particular a robust encoding of ordinals below ω^{ω^k} as sequences over an alphabet of $k+2$ symbols, for which we adapt the constructions of [9, 15]; see Sec. 3. Compared with previous work, we make the most of the rational relations framework, leading to simpler and more detailed proofs of robustness.

This allows us to show in Sec. 4 that for $k \geq 2$, $(k+2)$-LR[1-bld], i.e. a version of LR[Rat] over an alphabet of size $|\Sigma| = k+2$ and with a relation R with bounded length discrepancy of 1, is $\mathbf{F}_{\omega^k}$-hard. We also show that this lower bound is quite tight, as $(k+2)$-LR[Rat] is in $\mathbf{F}_{\omega^{k+1}}$.

We then show in Sec. 5 that LR[1-bld] can easily be reduced to EP[Rat] and other (parameterized) embedding problems—including EP[Sync], a restriction of EP[Rat] introduced by Barceló et al. [4] where the relation R is *synchronous* (aka *regular*), and which required a complex lower bound proof.

Let us now turn to the necessary formal background on PEPs in Sec. 2. Due to space constraints, some proof details will be found in the full version of this paper, available as arXiv:1211.5259 [cs.LO].

2 Post Embedding Problems

Rational Relations [11] play an important role in the following, as they provide a notion of finitely presentable relations over strings more powerful than string rewrite systems, and come with a large body of theory and results [see e.g. 25, Chap. IV]. Let us quickly skim over the notations and definitions that will be needed in this paper.

We assume the reader to be familiar with the basic characterizations of *rational relations* R between two finite alphabets Σ and Δ by

closure of the finite relations in $\Sigma^* \times \Delta^*$ under union, concatenation, and Kleene star,[1]

finite transductions defined by normalized transducers $\mathcal{T} = \langle Q, \Sigma, \Delta, \delta, I, F\rangle$ where Q is a finite set of states, $\delta \subseteq Q \times ((\Sigma \times \{\varepsilon\}) \cup (\{\varepsilon\} \times \Delta)) \times Q$ is a transition relation—where ε denotes the empty word, of length $|\varepsilon| = 0$—, initial set of states $I \subseteq Q$, and final set of states $F \subseteq Q$,

decomposition into a regular language L over some finite alphabet Γ and two morphisms $u\colon \Gamma^* \to \Sigma^*$ and $v\colon \Gamma^* \to \Delta^*$ s.t. $R = u^{-1} \mathbin{;} \mathrm{Id}_L \mathbin{;} v$, where Id_L is the identity function over the restricted domain L.

[1] We use different symbols "$*$" and "$+$" for Kleene star and Kleene plus, i.e. iteration of concatenation "$\cdot$" on the one hand, and "$\circledast$" and "$\oplus$" for reflexive transitive closure and transitive closure, i.e. iteration of composition "$;$" on the other hand. Rational relations and length-preserving relations are closed under Kleene star, but none of the classes of relations we consider is closed under reflexive transitive closure.

This last characterization is known as Nivat's Theorem, and shows that EP[Rat] can be stated alternatively as taking as input a regular language L in Γ^* and two morphisms u and v from Γ^* to Σ^* and asking whether there exists some word x in L s.t. $u(x) \sqsubseteq v(x)$ [7]. This justifies the name of "Post Embedding Problem", as the well-known, undecidable *Post Correspondence Problem* asks instead given u and v whether there exists x in Γ^+ s.t. $u(x) = v(x)$.

Synchronous Relations are a restricted class of rational relations, and are closed under intersection and complement, in addition to e.g. the closure under composition and inverse that all rational relations enjoy. A rational relation has *b-bounded length discrepancy* if the absolute value of $|u| - |v|$ is at most b for all (u, v) in R, and has *bounded length discrepancy* (bld) if there exists such a finite b. In particular, it is *length-preserving* if $|u| = |v|$, i.e. if it has bld 0. A *synchronous relation* is a finite union of relations of form $\{(u, vw) \mid (u, v) \in R \wedge w \in L\}$ and $\{(uw, v) \mid (u, v) \in R \wedge w \in L\}$ where R ranges over length-preserving rational relations and L over regular languages. In terms of classes of relations in $\Sigma^* \times \Delta^*$, we have the strict inclusions [25]:

$$\text{lp} = 0\text{-bld} \subsetneq \cdots \subsetneq b\text{-bld} \subsetneq (b+1)\text{-bld} \subsetneq \cdots \subsetneq \text{bld} \subsetneq \text{Sync} \subsetneq \text{Rat} . \qquad (1)$$

Post Embedding Problems, as we have seen in the introduction, are concerned with the interplay of a rational relation R in $\Sigma^* \times \Sigma^*$ with the subword embedding ordering $\sqsubseteq$. The latter is a particular case of a (deterministic) rational relation that is not synchronous. Both EP[Rat] and LR[Rat] are particular instances of more general, undecidable problems: the emptiness of intersection of two rational relations for EP[Rat], and the word problem in the reflexive transitive closure of a rational relation for LR[Rat]. We can add another natural problem to the set of PEPs:

Lossy Termination (LT[Rat])

Input. A rational relation R over Σ and a word w in Σ^*.
Question. Does $R_{\sqsupseteq}^{\circledast}$ terminate from w, i.e. is every sequence $w = w_0 \, R_{\sqsupseteq} \, w_1 \, R_{\sqsupseteq} \cdots R_{\sqsupseteq} \, w_i \, R_{\sqsupseteq} \cdots$ with $w_0, w_1, \ldots, w_i, \ldots$ in Σ^* finite?

Again, this is a variant of the termination problem, which is in general undecidable when the relation is not lossy.

Restrictions. We parameterize PEPs with the subclass of rational relations under consideration for R and the cardinal of the alphabet Σ; for instance, $(k+2)$-EP[Sync] is the variant of EP[Rat] where the relation is synchronous and $|\Sigma| = k+2$. We are interested in this paper in providing $\mathbf{F}_{\omega^k}$ lower bounds with the smallest possible class of relations and smallest possible alphabet size, but we should also mention that some (rather strong) restrictions become tractable:

– Barceló et al. [4] show that EP[Rec]—where a *recognizable relation* is a finite union of products $L \times L'$ where L and L' range over regular languages—is in NLogSpace, because the intersection $R \cap \sqsubseteq$ is rational, and can effectively be constructed and tested for emptiness on the fly,

- Chambart and Schnoebelen [7] show that EP[2Morph]—where a *2-morphic relation* [20] is the composition $R = (u^{-1} \mathbin{⨾} v) \setminus \{(\varepsilon, \varepsilon)\}$ of two morphisms u and v from Γ^* to Σ^*—is in LogSpace, because it reduces to checking whether there exists a in Γ s.t. $u(a) \sqsubseteq v(a)$,
- the case EP[Rewr] of *rewrite relations* is similarly in LogSpace: a rewrite relation R is defined from a *semi-Thue system*, i.e. a finite set Υ of rules (u, v) in $\Sigma^* \times \Sigma^*$, as $\rightarrow_\Upsilon = \{(wuw', wvw') \mid w, w' \in \Sigma^* \wedge (u, v) \in \Upsilon\}$, and EP[Rewr] reduces to checking whether $u \sqsubseteq v$ for some rule (u, v) of Υ,
- the unary alphabet case of 1-EP[Rat] is in NLogSpace: this can be seen using Parikh images and Presburger arithmetic:

Proposition 1. *The problem* 1-*EP[Rat] is in* NLogSpace.

3 Hardy Computations

We use the *Hardy hierarchy* as our main subrecursive hierarchy [21, 24, 12]. Although we will only use the lower levels of this hierarchy, its general definition is worth knowing, as it is archetypal of ordinal-indexed *subrecursive hierarchies*; see [27] for a self-contained presentation.

3.1 The Hardy Hierarchy

Ordinal Terms. Let ε_0 be the smallest solution of the equation $\omega^x = x$. It is well-known that any ordinal $\alpha < \varepsilon_0$ can be written uniquely in Cantor Normal Form (CNF) as a sum

$$\alpha = \omega^{\beta_1} \dotplus \cdots \dotplus \omega^{\beta_n} \tag{2}$$

where $\beta_n \leq \cdots \leq \beta_1 < \alpha$ and each β_i is itself in CNF. This ordinal α is 0 if $n = 0$ in (2), a *successor ordinal* if β_n is 0, and a *limit ordinal* otherwise. In the following, we write $\alpha \dotplus \beta$ to denote a direct sum $\alpha + \beta$ where $\alpha > \beta$ or $\alpha = 0$.

Subrecursive hierarchies are defined through assignments of *fundamental sequences* $(\lambda_n)_{n<\omega}$ for limit ordinals $\lambda < \varepsilon_0$, satisfying $\lambda_n < \lambda$ for all n and $\lambda = \sup_n \lambda_n$. A standard assignment on terms in CNF is defined by:

$$\left(\gamma \dotplus \omega^{\alpha \dotplus 1}\right)_n \stackrel{\text{def}}{=} \gamma \dotplus \omega^\alpha \cdot n, \qquad \left(\gamma \dotplus \omega^\lambda\right)_n \stackrel{\text{def}}{=} \gamma \dotplus \omega^{\lambda_n}, \tag{3}$$

thus verifying $\omega_n = n$. Let $\Omega \stackrel{\text{def}}{=} \omega^{\omega^\omega}$; this yields for instance $\Omega_k = \omega^{\omega^k}$ and, if $k > 0$, $(\Omega_k)_n = \omega^{\omega^{k-1} \cdot n}$.

Hardy Hierarchy. The *Hardy hierarchy* $(H^\alpha)_{\alpha<\varepsilon_0}$ is an ordinal-indexed hierarchy of functions $H^\alpha \colon \mathbb{N} \to \mathbb{N}$ defined by

$$H^0(n) \stackrel{\text{def}}{=} n \qquad H^{\alpha \dotplus 1}(n) \stackrel{\text{def}}{=} H^\alpha(n+1) \qquad H^\lambda(n) \stackrel{\text{def}}{=} H^{\lambda_n}(n)\ . \tag{4}$$

Observe that H^1 is simply the successor function, and more generally H^α is the αth iterate of the successor function, using diagonalisation to treat limit ordinals. A related hierarchy is the *fast growing hierarchy* $(F_\alpha)_{\alpha<\varepsilon_0}$, which can be defined by $F_\alpha \stackrel{\text{def}}{=} H^{\omega^\alpha}$, resulting in $F_0(n) = H^1(n) = n+1$, $F_1(n) = H^\omega(n) = H^n(n) = 2n$, $F_2(n) = H^{\omega^2}(n) = 2^n n$ being exponential, $F_3 = H^{\omega^3}$ being non-elementary, $F_\omega = H^{\omega^\omega} = H^{\Omega_1}$ being an Ackermannian function, $F_{\omega^k} = H^{\Omega_k}$ a k-Ackermannian function, and $F_{\omega^\omega} = H^\Omega$ an hyper-Ackermannian function.

Fast-Growing Complexity Classes. Our intention is to establish the "F_{ω^k} hardness" of Post embedding problems. In order to make this statement more precise, we define the class $\mathbf{F}_{\omega^k}$ of *k-Ackermannian problems* as a specific instance of the *fast-growing complexity classes* defined for $\alpha \geq 3$ by

$$\mathbf{F}_\alpha \stackrel{\text{def}}{=} \bigcup_{p \in \bigcup_{\beta<\alpha} \mathscr{F}_\beta} \mathrm{DT}\textsc{ime}(F_\alpha(p(n))) , \qquad \mathscr{F}_\alpha = \bigcup_{c<\omega} \mathrm{FDT}\textsc{ime}(F_\alpha^c(n)) , \tag{5}$$

where $\mathscr{F}_\alpha$ defined above is the αth level of the *extended Grzegorczyk hierarchy* [21] when $\alpha \geq 2$. The classes $\mathbf{F}_\alpha$ are naturally equipped with $\bigcup_{\beta<\alpha} \mathscr{F}_\beta$ as class of reductions. For instance, because $\bigcup_{k<\omega} \mathscr{F}_{\omega^k}$ is exactly the set of multiply-recursive functions, $\mathbf{F}_{\omega^\omega}$ captures the intuitive notion of hyper-Ackermannian problems closed under multiply-recursive reductions.[2]

Hardy Computations. The fast-growing and Hardy hierarchies have been used in several publications to establish Ackermannian and higher complexity bounds [9, 26, 15, 27]. The principle in their use for lower bounds is to view (4), read left-to-right, as a rewrite system over $\varepsilon_0 \times \mathbb{N}$, and later implement it in the targeted formalism. Formally, a (forward) *Hardy computation* is a sequence

$$\alpha_0, n_0 \rightarrow \alpha_1, n_1 \rightarrow \alpha_2, n_2 \rightarrow \cdots \rightarrow \alpha_\ell, n_\ell \tag{6}$$

of evaluation steps implementing the equations in (4) seen as left-to-right rewrite rules over *Hardy configurations* α, n. It guarantees $\alpha_0 > \alpha_1 > \alpha_2 > \cdots$ and keeps $H^{\alpha_i}(n_i)$ invariant. We say it is *complete* when $\alpha_\ell = 0$ and then $n_\ell = H^{\alpha_0}(n_0)$ (we also consider incomplete computations). A *backward Hardy computation* is obtained by using (4) as right-to-left rules. For instance,

$$\omega^{\omega^k}, n \rightarrow \omega^{\omega^{k-1}\cdot n}, n \rightarrow \omega^{\omega^{k-1}\cdot(n-1) \dotplus \omega^{k-2}\cdot n}, n \tag{7}$$

constitute the first three steps of the forward Hardy computation starting from Ω_k, n if $k > 1$ and $n > 0$.

[2] Note that, at such high complexities, the usual distinctions between deterministic vs. nondeterministic, or time-bounded vs. space-bounded computations become irrelevant. In particular, $\mathscr{F}_2$ is the set of elementary functions, and $\mathbf{F}_3$ the class of problems with a tower of exponentials of height bounded by some elementary function of the input as an upper bound.

Termination of Hardy Computations. Because $\alpha_0 > \alpha_1 > \cdots > \alpha_\ell$ in a forward Hardy computation like (6), it necessarily terminates. For inverse computations, this is less immediate, and we introduce for this a *norm* $\|\alpha\|$ of an ordinal α in ε_0 as its count of "ω" symbols when written as an ordinal term: formally, $\|.\|: \varepsilon_0 \to \mathbb{N}$ is defined by

$$\|0\| \stackrel{\text{def}}{=} 0 \qquad \|\omega^\alpha\| \stackrel{\text{def}}{=} 1 + \|\alpha\| \qquad \|\alpha \dotplus \alpha'\| \stackrel{\text{def}}{=} \|\alpha\| + \|\alpha'\| \,. \tag{8}$$

We can check that, for any limit ordinal λ, $\|\lambda_n\| > \|\lambda\|$ whenever $n > 1$. Therefore, in a backward Hardy computation, the pair $(n, \|\alpha\|)$ decreases for the lexicographic ordering over $\mathbb{N}^2$. As this is a well-founded ordering, we see that backward computations terminate if n remains larger than 1—which is a reasonable hypothesis for the following.

3.2 Encoding Hardy Configurations

Our purpose is now to encode Hardy computations as relations over Σ^*. This entails in particular (1) encoding configurations α, n in $\Omega_k \times \mathbb{N}$ of a Hardy computation as finite sequences using *cumulative ordinal descriptions* or "*codes*", which we do in this subsection, and (2) later in Sec. 3.3 designing a 1-bld relation that implements Hardy computation steps over codes. A constraint on codes is that they should be *robust* against losses, i.e. if $\pi(x)$ and $\pi(x')$ are the ordinals associated to the codes x and x' and $\pi(x) \sqsubseteq \pi(x')$, then $H^{\pi(x)}(n) \leq H^{\pi(x')}(n)$—pending some hygienic conditions on x and x', see Lem. 2.

Finite Ordinals below k can be represented as single symbols $a_0, \ldots, a_{k-1}$ of an alphabet Σ_k along with a bijection

$$\varphi(a_i) \stackrel{\text{def}}{=} i \,. \tag{9}$$

Small Ordinals below ω^k are then easily encoded as finite words over Σ_k: given a word $w = b_1 \cdots b_n$ over Σ_k, we define its associated ordinal in ω^k as

$$\beta(w) \stackrel{\text{def}}{=} \omega^{\varphi(b_1)} + \cdots + \omega^{\varphi(b_n)} \,. \tag{10}$$

Note that β is surjective but not injective: for instance, $\beta(a_0 a_1) = \beta(a_1) = \omega$. By restricting ourselves to *pure* words over Σ_k, i.e. words satisfying $\varphi(b_j) \geq \varphi(b_{j+1})$ for all $1 \leq j < n$, we obtain a bijection between ω^k and $\mathsf{p}(\Sigma_k^*)$ the set of pure finite words in Σ_k^*, because then (10) is the CNF of $\beta(w)$.

Large Ordinals below Ω_k are denoted by *codes* [9, 15], which are #-separated words over the extended alphabet $\Sigma_{k\#} \stackrel{\text{def}}{=} \Sigma_k \uplus \{\#\}$. A code x can be seen as a concatenation $w_1 \# w_2 \# \cdots \# w_p \# w_{p+1}$ where each w_i is a word over Σ_k. Its associated ordinal $\pi(x)$ in Ω_k is then defined as

$$\pi(x) \stackrel{\text{def}}{=} \omega^{\beta(w_1 w_2 \cdots w_p)} \dotplus \cdots \dotplus \omega^{\beta(w_1 w_2)} \dotplus \omega^{\beta(w_1)} \,, \tag{11}$$

or inductively by

$$\pi(w) \stackrel{\text{def}}{=} 0, \qquad \pi(w\#x) \stackrel{\text{def}}{=} \omega^{\beta(w)} \cdot \pi(x) \dotplus \omega^{\beta(w)} \tag{12}$$

for w a word in Σ_k^* and x a code. For instance, $\pi(a_1a_0\#) = \omega^{\omega \dotplus 1} = \pi(a_0a_1a_0\#a_3)$, or, closer to our concerns, the initial ordinal in our computations is $\pi(a_{k-1}^n\#) = (\Omega_k)_n$ when $k > 0$.

Observe that π is surjective, but not injective. We can mend this by defining a *pure* code $x = w_1\#\cdots\#w_p\#w_{p+1}$ as one where $w_{p+1} = \varepsilon$ and every word w_i for $1 \leq i \leq p$ is pure—note that it does not force the concatenation of two successive words w_iw_{i+1} of x to be pure. This is intended, as this is the very mechanism that allows π to be a bijection between Ω_k and $\mathsf{p}(\Sigma_{k\#}^*)$

Lemma 1. *The function π is a bijection from $\mathsf{p}(\Sigma_{k\#}^*)$ to Ω_k.*

We also define $\mathsf{p}(x)$ to be the unique pure code x' verifying $\pi(x) = \pi(x')$; then $\mathsf{p}(x) \sqsubseteq x$, and $x \sqsubseteq x'$ implies $\mathsf{p}(x) \sqsubseteq \mathsf{p}(x')$.

Hardy Configurations α, n are finally encoded as sequences $c = \pi^{-1}(\alpha) \shortmid \#^n$ using a separator "$\shortmid$", i.e. as sequences in the language $\text{Confs} \stackrel{\text{def}}{=} \mathsf{p}(\Sigma_{k\#}^*)\cdot\{\shortmid\}\cdot\{\#\}^*$. This is a regular language over $\Sigma_{k\#} \uplus \{\shortmid\}$, but the most important fact about this encoding is that it is *robust* against symbol losses as far as the corresponding computed Hardy values are concerned. Robustness is a critical part of hardness proofs based on Hardy functions. The main difficulty rises from the fact that the Hardy functions are not monotone in their ordinal parameter: for instance, $H^\omega(n) = H^n(n) = 2n$ is less than $H^{n \dotplus 1}(n) = 2n + 1$. Code robustness is addressed in [9, Prop. 4.3]. Robustness is the limiting factor that prevents us from reducing languages in $\mathbf{F}_\alpha$ for $\alpha > \Omega$ into PEPs.

Lemma 2 (Robustness). *Let $c = x \shortmid \#^n$ and $c' = x' \shortmid \#^{n'}$ be two sequences in* Confs*. If $c \sqsubseteq c'$, then $H^{\pi(x)}(n) \leq H^{\pi(x')}(n')$.*

3.3 Encoding Hardy Computations

It remains to present a 1-bld relation that implements Hardy computations over Hardy configurations encoded as sequences in Confs. We translate the equations from (4) into a relation $R_H = (R_0 \cup R_1 \cup R_2) \cap (\text{Confs} \times \text{Confs})$, which can be reversed for backward computations:

$$R_0 \stackrel{\text{def}}{=} \{(\#x \shortmid \#^n, x \shortmid \#^{n+1}) \mid n \geq 0, x \in \Sigma_{k\#}^*\} \tag{13}$$

$$R_1 \stackrel{\text{def}}{=} \{(wa_0\#x \shortmid \#^n, w\#^n\mathsf{p}(a_0x) \shortmid \#^n) \mid n > 1, w \in \Sigma_k^*, x \in \Sigma_{k\#}^*\} \tag{14}$$

$$R_2 \stackrel{\text{def}}{=} \{(wa_i\#x \shortmid \#^n, wa_{i-1}^n\#\mathsf{p}(a_ix) \shortmid \#^n) \mid n > 1, i > 0, w \in \Sigma_k^*, x \in \Sigma_{k\#}^*\} \tag{15}$$

The relation R_0 implements the successor case, while R_1 and R_2 implement the limit case of (3) for ordinals of form $\gamma \dotplus \omega^{\alpha+1}$ and $\gamma \dotplus \omega^\lambda$ respectively. The restriction to $n > 1$ in R_1 and R_2 enforces termination for backward computations; it is not required for correctness. Because R_H is a direct translation of (4) over Confs:

Lemma 3 (Correctness). *For all* α, α' *in* Ω_k *and* $n, n' > 1$, $(\pi^{-1}(\alpha) \shortmid \#^n)$ $(R_H \cup R_H^{-1})^{\circledast}(\pi^{-1}(\alpha') \shortmid \#^{n'})$ *iff* $H^\alpha(n) = H^{\alpha'}(n')$.

Unfortunately, although R_0 is a length-preserving rational relation, R_1 and R_2 are not 1-bld, nor even rational. However, they can easily be broken into smaller steps, which are rational—as we are applying a reflexive transitive closure, this is at no expense in generality. This requires more complex encodings of Hardy configurations, with some "finite state control" and a working space in order to keep track of where we are in our small steps. Because we do not want to spend new symbols in this encoding, given some finite set Q of states, we work on sequences in

$$\text{Seqs} \stackrel{\text{def}}{=} \{a_0, a_1\}^{\lceil \log |Q| \rceil} \cdot \{\shortmid\} \cdot \mathsf{p}(\Sigma_k^*) \cdot \{\#\}^* \cdot \{\shortmid\} \cdot \mathsf{p}(\Sigma_{k\#}^*) \cdot \{\shortmid\} \cdot \{\#, a_0, a_1\}^* . \quad (16)$$

with four segments separated by "$\shortmid$": a state, a working segment, an ordinal encoding, and a counter. Given a state q in Q, we use implicitly its binary encoding as a sequence of fixed length over $\{a_0, a_1\}$.

We define two relations Fw and Bw with domain and range Seqs that implement forward and backward computations with R_H. A typical case is that of computations with R_1, which can be implemented as the closure of the union:

$$q_{\mathsf{Fw}} \shortmid\shortmid wa_0\#x \shortmid \#^{n+2} \ \mathsf{Fw}_1 \ q_{\mathsf{Fw}_1} \shortmid w\# \shortmid \mathsf{p}(a_0x) \shortmid \#^{n+1}a_0 \quad (17)$$

$$q_{\mathsf{Fw}_1} \shortmid w\#^m \shortmid x \shortmid \#^{n+1}a_0^{p+1} \ \mathsf{Fw}_1 \ q_{\mathsf{Fw}_1} \shortmid w\#^{m+1} \shortmid x \shortmid \#^n a_0^{p+2} \quad (18)$$

$$q_{\mathsf{Fw}_1} \shortmid w\#^{m+1} \shortmid x \shortmid a_0^{n+2} \ \mathsf{Fw}_1 \ q_{\mathsf{Fw}_1} \shortmid\shortmid w\#^{m+1}x \shortmid \#^{n+2} \quad (19)$$

for m, n, p in $\mathbb{N}$, w in $\mathsf{p}(\Sigma_k^*)$, and x in $\mathsf{p}(\Sigma_{k\#}^*)$. Note that $\mathsf{p}(a_0x)$ returns a_0x if x begins with $\#$ or a_0, and x otherwise. The corresponding backward computation for R_1 inverses the relations in (17–19) and substitutes q_{Bw} and q_{Bw_1} for q_{Fw} and q_{Fw_1}. The reader should be able to convince herself that this is indeed feasible in a rational 1-bld fashion; for instance, (18) can be written as a rational expression

$$\begin{bmatrix} q_{\mathsf{Fw}_1}\shortmid \\ q_{\mathsf{Fw}_1}\shortmid \end{bmatrix} \cdot \mathrm{Id}_{\Sigma_k^*} \cdot \begin{bmatrix} \# \\ \# \end{bmatrix}^* \cdot \begin{bmatrix} \varepsilon \\ \# \end{bmatrix} \cdot \begin{bmatrix} \shortmid \\ \shortmid \end{bmatrix} \cdot \mathrm{Id}_{\Sigma_{k\#}^*} \cdot \begin{bmatrix} \shortmid \\ \shortmid \end{bmatrix} \cdot \begin{bmatrix} \# \\ \# \end{bmatrix}^* \cdot \begin{bmatrix} \# \\ \varepsilon \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_0 \end{bmatrix}^+ \cdot \begin{bmatrix} \varepsilon \\ a_0 \end{bmatrix} . \quad (20)$$

Observe that separators "$\shortmid$" are reliable, and that losses cannot pass unnoticed in the constant-sized state segment of a sequence in Seqs; thus we can use lemmas 2 and 3 to prove that $\mathsf{Fw}_{\sqsupseteq}^{\circledast}$ and $\mathsf{Bw}_{\sqsupseteq}^{\circledast}$ are "weak" implementations of H^α and its inverse when α is in Ω_k. Not any reformulation of R_H as the closure of a rational relation would work here: our relation also needs to be robust to losses; see the full paper for details.

Lemma 4 (Weak Implementation). *The relations* Fw *and* Bw *are 1-bld and terminating. Furthermore, if* $k \geq 1$, $m, n > 1$ *and* $\alpha \in \Omega_k$,

$$(q_{\mathsf{Fw}} \shortmid\shortmid \pi^{-1}(\alpha) \shortmid \#^n) \ \mathsf{Fw}_{\sqsupseteq}^{\circledast} \ (q_{\mathsf{Fw}} \shortmid\shortmid\shortmid \#^m) \qquad \textit{implies } m \leq H^\alpha(n)$$

$$(q_{\mathsf{Bw}} \shortmid\shortmid\shortmid \#^m) \ \mathsf{Bw}_{\sqsupseteq}^{\circledast} \ (q_{\mathsf{Bw}} \shortmid\shortmid \pi^{-1}(\alpha) \shortmid \#^n) \qquad \textit{implies } m \geq H^\alpha(n)$$

and there exists rewrites verifying $m = H^\alpha(n)$ *in both of the above cases.*

4 The Parametric Complexity of LR[1-bld]

Now equipped with suitable encodings for Hardy computations, we can turn to the main result of the paper: Prop. 2 below shows the $\mathbf{F}_{\omega^k}$-hardness of $(k+2)$-LR[1-bld]. As we obtain almost matching upper bounds in Sec. 4.2, we deem this to be rather tight.

4.1 Lower Bound

Thanks to the relations over $\Sigma_{k\#} \uplus \{\shortmid\}$ defined in Sec. 3, we know that we can weakly compute with Fw a "budget space" as a unary counter of size $F_{\omega^k}(n)$, and later check that this budget has been maintained by running through Bw. We are going to insert the simulation of an $\mathbf{F}_{\omega^k}$-hard problem between these two phases of budget construction and budget verification, thereby constructing $\mathbf{F}_{\omega^k}$-hard instances of $(k+2)$-LR[1-bld].

Proposition 2. *Let $k \geq 2$. Then $(k+2)$-LR[1-bld] is $\mathbf{F}_{\omega^k}$-hard.*

Bounded Semi-Thue Reachability. The problem we reduce from is a space-bounded variant of the *semi-Thue reachability problem* (aka *semi-Thue word problem*): as already mentioned in Sec. 2, a *semi-Thue system* Υ over an alphabet is a finite set of rules (u, v) in $\Sigma^* \times \Sigma^*$, defining a *rewrite relation* $\rightarrow_\Upsilon$. The semi-Thue reachability problem, or R[Rewr], is a reliable version of the lossy reachability problem. This problem is in general undecidable, as one can express the "next configuration" relation of a Turing machine as a semi-Thue system. Its F_{ω^k}*-bounded* version for some $k \geq 1$ takes as input an instance $\langle \Upsilon, y, y' \rangle$ of size n where, if $y \rightarrow_\Upsilon^{\circledast} x$, then $|x| \leq F_{\omega^k}(n)$. This is easily seen to be hard for $\mathbf{F}_{\omega^k}$, even for a binary alphabet Σ.

Reduction. Let $\langle \Upsilon, y, y' \rangle$ be an instance of size $n > 1$ of F_{ω^k}-bounded R[Rewr] over the two-letters alphabet $\{a_0, a_1\}$. We build a $(k+2)$-LR[1-bld] instance in which the rewrite relation R performs the following sequence:

1. Weakly compute a budget of size $F_{\omega^k}(n)$, using Fw described in Sec. 3.
2. In this allocated space, simulate the rewrite steps of Υ starting from y.
3. Upon reaching y', perform a reverse Hardy computation using Bw and check that we obtain back the initial Hardy configuration. This check ensures that the lossy rewrites were in fact reliable (i.e., no symbols were lost).

For Phase 2, we define a #-padded version Sim of $\rightarrow_\Upsilon$ that works over Seqs:

$$\mathsf{Sim} \stackrel{\text{def}}{=} \{(q_{\mathsf{Sim}} \shuffle u\#^p, q_{\mathsf{Sim}} \shuffle v\#^q) \mid u \rightarrow_\Upsilon v, |u| + p = |v| + q\} . \tag{21}$$

This is a length-preserving rational relation. We define two more length-preserving rational relations Init and Fin that initialize the simulation with y on the budget space, and launch the verification phase if y' appears there, allowing to move from Phase 1 to Phase 2 and from Phase 2 to Phase 3, respectively:

$$\mathsf{Init} \stackrel{\text{def}}{=} \{(q_{\mathsf{Fw}} \shuffle \#^{\ell+|y|}, q_{\mathsf{Sim}} \shuffle y\#^\ell) \mid \ell \geq 0\} , \tag{22}$$

$$\mathsf{Fin} \stackrel{\text{def}}{=} \{(q_{\mathsf{Sim}} \shuffle y'\#^\ell, q_{\mathsf{Bw}} \shuffle \#^{\ell+|y'|}) \mid \ell \geq 0\} . \tag{23}$$

Finally, because $F_{\omega^k}(n) = H^{(\Omega_k)_n}(n)$, we define our source and target by

$$w \stackrel{\text{def}}{=} q_{\mathsf{Fw}} \shortparallel a_{k-1}^n \# \mid \#^n , \qquad w' \stackrel{\text{def}}{=} q_{\mathsf{Bw}} \shortparallel a_{k-1}^n \# \mid \#^n , \tag{24}$$

and we let R be the 1-bld rational relation $\mathsf{Fw} \cup \mathsf{Init} \cup \mathsf{Sim} \cup \mathsf{Fin} \cup \mathsf{Bw}$.

Claim. The given R[Rewr] instance is positive if and only if $\langle R, w, w' \rangle$ is a positive instance of the $(k+2)$-LR[1-bld] problem.

Proof. Suppose $w\ R_{\sqsupseteq}^{\circledast}\ w'$. It is easy to see that the separator symbol "$\mid$" and the encodings of states from Q are reliable. Because of the way the relations treat the states, we in fact get

$$w\ \mathsf{Fw}_{\sqsupseteq}^{\circledast}\ (q_{\mathsf{Fw}} \mid\mid\mid \#^{\ell_1})\ \mathsf{Init}_{\sqsupseteq}\ (q_{\mathsf{Sim}} \mid\mid\mid z_1)\ \mathsf{Sim}_{\sqsupseteq}^{\circledast}\ (q_{\mathsf{Sim}} \mid\mid\mid z_2)\ \mathsf{Fin}_{\sqsupseteq}\ (q_{\mathsf{Sim}} \mid\mid\mid \#^{\ell_2})\ \mathsf{Bw}_{\sqsupseteq}^{\circledast}\ w'$$

for some strings z_1, z_2 and naturals $\ell_1, \ell_2 \in \mathbb{N}$. By Lem. 4, we have $\ell_1 \leq F_{\omega^k}(n)$ and $\ell_2 \geq F_{\omega^k}(n)$. Since Init, Sim, and Fin are length-preserving, we get

$$F_{\omega^k(n)} \geq \ell_1 \geq |z_1| \geq |z_2| \geq \ell_2 \geq F_{\omega^k(n)} \tag{25}$$

Thus equality holds throughout, and therefore the lossy steps of $\mathsf{Sim}_{\sqsupseteq}$ in Phase 2 were actually reliable, i.e. were steps of Sim. This allows us to conclude that the original R[Rewr] instance was positive.

Suppose conversely that the R[Rewr] instance is positive. We can translate this into a witnessing run for $w\ R_{\sqsupseteq}^{\circledast}\ w'$, in particular, for $w\ \mathsf{Fw}^{\circledast} \mathbin{⨾} \mathsf{Init} \mathbin{⨾} \mathsf{Sim}^{\circledast} \mathbin{⨾} \mathsf{Fin} \mathbin{⨾} \mathsf{Bw}^{\circledast}\ w'$, because any successful run from the R[Rewr] instance can be plugged into the $\mathsf{Sim}^{\circledast}$ phase; Lem. 4 and the fact that the configurations of Υ are bounded by $F_{\omega^k}(n)$ together ensure that this can be done. ☐

4.2 Upper Bound

Well-Structured Transition Systems. As a preliminary, let us show that the lossy rewriting problem is decidable. Indeed, the relation $R_{\sqsupseteq}$ can be viewed as the transition relation of an infinite transition system over the state space Σ^*. Furthermore, by Higman's Lemma, the subword embedding ordering $\sqsubseteq$ is a *well quasi ordering* (wqo) over Σ^*, and the relation $R_{\sqsupseteq}$ is *compatible* with it: if $u\,R_{\sqsupseteq}\,v$ and $u \sqsubseteq u'$ for some u, v, u' in Σ^*, then there exists v' in Σ^* s.t. $u'\,R_{\sqsupseteq}\,v'$: here it suffices to use $v' = v$ by transitivity of $\sqsupseteq$.

A transition system $\mathcal{S} = \langle S, \rightarrow, \leq \rangle$ with a wqo $(S, \leq)$ as state space and a compatible transition relation $\rightarrow$ is called a *well-structured transition system* (WSTS), and several problems are decidable on such systems under very light effectiveness assumptions [1, 13], among which the *coverability problem*, which asks given a WSTS $\mathcal{S}$ and two states s and s' in S whether there exists $s'' \geq s'$ s.t. $s \rightarrow^{\circledast} s''$. The lossy rewrite problem when $w \not\sqsupseteq w'$ can be restated as a coverability problem for the WSTS $\langle \Sigma^*, R_{\sqsupseteq}, \sqsubseteq \rangle$ and w and w', since if there exists $w'' \sqsupseteq w'$ with $w\,R_{\sqsupseteq}^{\circledast}\,w''$, then $w\,R_{\sqsupseteq}^{\circledast}\,w'$ also holds by transitivity of $\sqsupseteq$.

Parameterized Upper Bound. In many cases, a *combinatory algorithm* can be employed instead of the classical backward coverability algorithm for WSTS: we can find a particular coverability witness $w' = w_0 \sqsubseteq \fatsemi R^{-1}\, w_1 \cdots w_{\ell-1} \sqsubseteq \fatsemi R^{-1}$ $w_\ell \sqsubseteq w$ of length ℓ *bounded* by a function akin to $F_{\omega^{k-1}}$ using the Length Function Theorem of [26]. This is a generic technique for coverability explained in [27], and the reader will find it instantiated for $(k+2)$-LR[Rat] in the long version of this paper:

Proposition 3 (Upper Bound). *The problem $(k+2)$-LR[Rat] is in $\mathbf{F}_{\omega^{k+1}}$.*

The small gap of complexity we witness here with Prop. 2 stems from the encoding apparatus, which charges us with one extra symbol. We have not been able to close this gap; for instance, the encoding breaks if we try to work without our separator symbol "$\shortmid$".

5 Applications

We apply in this section the proof of Prop. 2 to prove parametric complexity lower bounds for several problems. In three cases (propositions 4, 5, and 7 below), we proceed by a reduction from the LR problem, but take advantage of the specifics of the instances constructed in the proof Prop. 2 to obtain tighter parameterized bounds. The hardness proof for the LT problem in Prop. 6 requires more machinery, which needs to be incorporated to the construction of Sec. 4.1 in order to obtain a reduction.

Rational Embedding. We first deal with the classical embedding problem: We reduce from a $(k+2)$-LR[Rat] instance and use Prop. 2. The issue is to somehow convert an iterated composition into an iterated concatenation—the idea is similar to the one typically used for proving the undecidability of PCP.

Proposition 4. *Let $k \geq 2$. Then $(k+2)$-EP[Rat] is $\mathbf{F}_{\omega^k}$-hard.*

Proof. Assume without loss of generality that $w \neq w'$ in a $(k+2)$-LR[Rat] instance $\langle R, w, w' \rangle$. We consider sequences of consecutive configurations of $\sqsupseteq \fatsemi (R \fatsemi \sqsupseteq)^{\oplus}$ of form

$$w = v_0 \sqsupseteq u_0\, R\, v_1 \sqsupseteq u_1\, R\, v_2 \sqsupseteq \cdots R\, v_n \sqsupseteq u_n = w' \tag{26}$$

that prove the LR instance to be positive. Let \$ be a fresh symbol; we construct a new relation R' that attempts to read the u_i's on its first component and the v_i's on the second, using the \$'s for synchronization:

$$R' \stackrel{\text{def}}{=} \begin{bmatrix} \$w'\$ \\ \$ \end{bmatrix} \cdot \left(R \cdot \begin{bmatrix} \$ \\ \$ \end{bmatrix} \right)^{+} \cdot \begin{bmatrix} \varepsilon \\ w\$ \end{bmatrix} \tag{27}$$

Observe that in any pair of words (u, v) of R', one finds the same number of occurrences of the separator \$ in u and v, i.e. we can write $u = \$u_n\$ \cdots \$u_0\$$ and

$v = \$v_n\$ \cdots \$v_0\$$ with $n > 0$, verifying $v_0 = w$, $u_n = w'$, and $u_i \, R \, v_{i+1}$ for all i. Assume $u \sqsubseteq v$: the embedding ordering is restricted by the \$ symbols to the factors $u_i \sqsubseteq v_i$. We can therefore exhibit a sequence of form (26). Conversely, given a sequence of form (26), the corresponding pair (u, v) belongs to $R' \cap \sqsubseteq$.

In order to conclude, observe that we can set $\$ \stackrel{\text{def}}{=} \mathsf{I}$ in the proof of Prop. 2 and adapt the previous arguments accordingly, since "I" is preserved by R and appears in both w and w' in the particular instances we build. □

Synchronous Embedding. Turning now to the case of synchronous relations, we proceed as in the previous proof, but employ an extra padding symbol $\perp$ to construct a length-preserving version of the relation R in an instance of $(k+2)$-LR[Sync], allowing us to apply the Kleene star operator while remaining regular.

Proposition 5. *Let $k \geq 2$. Then $(k+3)$-EP[Sync] is $\mathbf{F}_{\omega^k}$-hard.*

Proof. Let $\langle R, w, w' \rangle$ be an instance of $(k+2)$-LR[Sync] with $w \neq w'$ and let \$ and $\perp$ be two fresh symbols. Because $R \cdot \{(\$, \$)\}$ is a synchronous relation, we can construct a padded length-preserving relation

$$R_\perp \stackrel{\text{def}}{=} \{(u\$\perp^m, v\$\perp^p) \mid m, p \geq 0 \wedge (u, v) \in R \wedge |u\$\perp^m| = |v\$\perp^p|\} \tag{28}$$

and define a relation similar to (27):

$$R'_\perp \stackrel{\text{def}}{=} \begin{bmatrix} \$w'\$ \\ \$ \end{bmatrix} \cdot R_\perp^+ \cdot \begin{bmatrix} \varepsilon \\ w\$ \end{bmatrix} \cdot \begin{bmatrix} \varepsilon \\ \perp \end{bmatrix}^* . \tag{29}$$

Let us show that $R'_\perp$ is regular: $\{(\$w'\$, \$)\}$ and $\{(\varepsilon, w\$)\}$ are relations with bounded length discrepancy and $R_\perp^*$ is length preserving, thus their concatenation has bounded length discrepancy, and can be effectively computed by *resynchronization* [25]. Suffixing $\{(\varepsilon, \perp)\}^*$ thus yields a synchronous relation.

As in the proof of Prop. 4, $R'_\perp$ preserves the \$ separators, thus if (u, v) belongs to $R'_\perp$, then we can write

$$\begin{aligned} u &= \$ \; u_n \; \$ \perp^{m_n} \; u_{n-1} \; \$ \perp^{m_{n-1}} \cdots \; \$ \perp^{m_1} \; u_0 \; \$ \perp^{m_0} , \\ v &= \$ \; v_n \; \$ \perp^{p_n} \; v_{n-1} \; \$ \perp^{p_{n-1}} \cdots \; \$ \perp^{p_1} \; v_0 \; \$ \perp^{p_0} . \end{aligned} \tag{30}$$

with $n > 0$ and $m_n = 0$. Furthermore, $v_0 = w$, $u_n = w'$, and $(u_i\$\perp^{m_i}, v_{i+1}\$\perp^{p_{i+1}})$ belongs to $R_\perp$, thus $u_i \, R \, v_{i+1}$ for all i. If the EP instance is positive, i.e. if $u \sqsubseteq v$, then $u_i \sqsubseteq v_i$ and $m_i \leq p_i$ for all i, and we can build a sequence of form (26) proving the LR instance to be positive. Conversely, if the LR instance is positive, there exists a sequence of form (26), and we can construct a pair (u, v) as in (30) above by guessing a sufficient padding amount p_0 that will allow to carry the entire rewriting. Finally, as in the proof of Prop. 4, we can set $\$ \stackrel{\text{def}}{=} \mathsf{I}$. □

Lossy Termination. In contrast with the previous cases, our hardness proof for the LT problem does not reduce from LR but directly from a semi-Thue word problem, by adapting the proof of Prop. 2 in such a way that $R_{\sqsupseteq}^{\circledast}$ is *guaranteed* to terminate. The main difference is that we reduce from a semi-Thue system where the length of *derivations* is bounded, rather than the length of configurations—this is still $\mathbf{F}_{\omega^k}$-hard since the distinction between time and space complexities is insignificant at such high complexities. The simulation of such a system then builds two copies of the initial budget in Phase 1: a *space* budget, where the derivation simulation takes place, and a *time* budget, which gets decremented with each new rewrite of Phase 2, and enforces its termination even in case of losses. See the full paper for details.

Proposition 6. *Let* $k \geq 2$*. Then* $(k+2)$*-LT[1-bld] is* $\mathbf{F}_{\omega^k}$*-hard.*

Lossy Channel Systems. By over-approximating the behaviours of a channel system by allowing uncontrolled, arbitrary message losses, Abdulla, Cécé, et al. [6, 2] obtain decidability results on an otherwise Turing-complete model. Many variants of this model have been studied in the literature [7, 8, 16], but our interest here is that LCSs were originally used as the formal model for the $\mathbf{F}_{\omega^\omega}$ lower bound proof of Chambart and Schnoebelen [9], rather than a PEP.

Formally, a *lossy channel system* (LCS) is a finite labeled transition system $\langle Q, \Sigma, \delta \rangle$ where transitions in $\delta \subseteq Q \times \{?, !\} \times \Sigma \times Q$ read and write on an unbounded channel. An channel system defines an infinite transition system over its set of configurations $Q \times \Sigma^*$—holding the current state and channel content—, with transition relation $q, x \to q', x'$ if either δ holds a read $(q, ?m, q')$ and $x = mx'$, or if it holds a write $(q, !m, q')$ and $xm = x'$. The operational semantics of an LCS then use the lossy version $\to_{\sqsupseteq}$ of this transition relation. In the following, we consider a slightly extended model, where transitions carry sequences of instructions instead, i.e. δ is a finite set included in $Q \times (\{?, !\} \times \Sigma)^* \times Q$. The natural decision problem associated with a LCS is its reachability problem:

Lossy Channel System Reachability (LCS)

Input. A LCS $\mathcal{C}$ and two configurations (q, x) and (q', x') of $\mathcal{C}$.
Question. Is (q', x') reachable from (q, x) in $\mathcal{C}$, i.e. does $q, x \to_{\sqsupseteq}^{\circledast} q', x'$?

The lossy rewriting problem easily reduces to a reachability problem in a LCS: the LCS *cycles* through the channel contents thanks to a distinguished symbol, and applies the rational relation at each cycle; see the full version for details.

Proposition 7. *Let* $k \geq 2$*. Then* $(k+2)$*-LCS is* $\mathbf{F}_{\omega^k}$*-hard.*

6 Concluding Remarks

Post embedding problems provide a high-level packaging of hyper-Ackermannian decision problems—and thanks to our parametric bounds, for k-Ackermannian problems—, compared to e.g. reachability in lossy channel systems (as used in [9]). The lossy rewriting problem is a prominent example: because it is stated in terms of a rational relation instead of a machine definition, it benefits automatically from the theoretical toolkit and multiple characterizations associated with rational relations. For a simple example, the *increasing* rewriting problem, which employs $R_{\sqsubseteq} \stackrel{\text{def}}{=} \sqsubseteq ; R ; \sqsubseteq$ instead of $R_{\sqsupseteq}$, is immediately seen to be equivalent to LR, by substituting R^{-1} for R and exchanging w and w'.

Interestingly, this inversion trick allows to show the equivalence of the lossy and increasing variants of all our problems, except for lossy termination:

Increasing Termination (IT[Rat])

Input. A rational relation R over Σ and a word w in Σ^*.
Question. Does $R_{\sqsubseteq}^{\circledast}$ terminate from w?

A related problem, termination of increasing channel systems with emptiness tests, is known to be in $\mathbf{F}_3$ [5] instead of $\mathbf{F}_{\omega^\omega}$ for LCS termination, but IT[Rat] is more involved. Like LR[Rat] or EP[Rat], it provides a high-level description, this time of *fair termination* problems in increasing channel systems, which are known to be equivalent to satisfiability of *safety metric temporal logic* [23, 22, 17]. The exact complexity of IT[Rat] is open, with a gigantic gap between the $\mathbf{F}_{\omega^\omega}$ upper bound provided by WSTS theory, and an $\mathbf{F}_4$ lower bound by Jenkins [17].

Acknowledgements. The authors thank Philippe Schnoebelen and the anonymous reviewers for their insightful comments.

References

1. Abdulla, P.A., Čerāns, K., Jonsson, B., Tsay, Y.K.: Algorithmic analysis of programs with well quasi-ordered domains. Inform. and Comput. 160, 109–127 (2000)
2. Abdulla, P.A., Jonsson, B.: Verifying programs with unreliable channels. Inform. and Comput. 127(2), 91–101 (1996)
3. Atig, M.F., Bouajjani, A., Burckhardt, S., Musuvathi, M.: On the verification problem for weak memory models. In: POPL 2010, pp. 7–18. ACM (2010)
4. Barceló, P., Figueira, D., Libkin, L.: Graph logics with rational relations and the generalized intersection problem. In: LICS 2012, pp. 115–124. IEEE Press (2012)
5. Bouyer, P., Markey, N., Ouaknine, J., Schnoebelen, P., Worrell, J.: On termination and invariance for faulty channel machines. Form. Asp. Comput. 24(4), 595–607 (2012)
6. Cécé, G., Finkel, A., Purushothaman Iyer, S.: Unreliable channels are easier to verify than perfect channels. Inform. and Comput. 124(1), 20–31 (1996)
7. Chambart, P., Schnoebelen, P.: Post Embedding Problem Is Not Primitive Recursive, with Applications to Channel Systems. In: Arvind, V., Prasad, S. (eds.) FSTTCS 2007. LNCS, vol. 4855, pp. 265–276. Springer, Heidelberg (2007)
8. Chambart, P., Schnoebelen, P.: Mixing Lossy and Perfect Fifo Channels. In: van Breugel, F., Chechik, M. (eds.) CONCUR 2008. LNCS, vol. 5201, pp. 340–355. Springer, Heidelberg (2008)
9. Chambart, P., Schnoebelen, P.: The ordinal recursive complexity of lossy channel systems. In: LICS 2008, pp. 205–216. IEEE Press (2008)
10. Chambart, P., Schnoebelen, P.: Pumping and Counting on the Regular Post Embedding Problem. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6199, pp. 64–75. Springer, Heidelberg (2010)

11. Elgot, C.C., Mezei, J.E.: On relations defined by generalized finite automata. IBM Journal of Research and Development 9(1), 47–68 (1965)
12. Fairtlough, M., Wainer, S.S.: Hierarchies of provably recursive functions. In: Handbook of Proof Theory, ch. III, pp. 149–207. Elsevier (1998)
13. Finkel, A., Schnoebelen, P.: Well-structured transition systems everywhere! Theor. Comput. Sci. 256(1-2), 63–92 (2001)
14. Friedman, H.M.: Some decision problems of enormous complexity. In: LICS 1999, pp. 2–13. IEEE Press (1999)
15. Haddad, S., Schmitz, S., Schnoebelen, P.: The ordinal-recursive complexity of timed-arc Petri nets, data nets, and other enriched nets. In: LICS 2012, pp. 355–364. IEEE Press (2012)
16. Jančar, P., Karandikar, P., Schnoebelen, P.: Unidirectional Channel Systems Can Be Tested. In: Baeten, J.C.M., Ball, T., de Boer, F.S. (eds.) TCS 2012. LNCS, vol. 7604, pp. 149–163. Springer, Heidelberg (2012)
17. Jenkins, M.: Synthesis and Alternating Automata over Real Time. Ph.D. thesis, Oxford University (2012)
18. Karandikar, P., Schnoebelen, P.: Cutting through Regular Post Embedding Problems. In: Hirsch, E.A., Karhumäki, J., Lepistö, A., Prilutskii, M. (eds.) CSR 2012. LNCS, vol. 7353, pp. 229–240. Springer, Heidelberg (2012)
19. Lasota, S., Walukiewicz, I.: Alternating timed automata. ACM Trans. Comput. Logic 9(2), 10 (2008)
20. Latteux, M., Leguy, J.: On the Composition of Morphisms and Inverse Morphisms. In: Díaz, J. (ed.) ICALP 1983. LNCS, vol. 154, pp. 420–432. Springer, Heidelberg (1983)
21. Löb, M., Wainer, S.: Hierarchies of number theoretic functions, I. Arch. Math. Logic 13, 39–51 (1970)
22. Ouaknine, J., Worrell, J.B.: Safety Metric Temporal Logic Is Fully Decidable. In: Hermanns, H., Palsberg, J. (eds.) TACAS 2006. LNCS, vol. 3920, pp. 411–425. Springer, Heidelberg (2006)
23. Ouaknine, J.O., Worrell, J.B.: On the decidability and complexity of Metric Temporal Logic over finite words. Logic. Meth. in Comput. Sci. 3(1), 8 (2007)
24. Rose, H.E.: Subrecursion: Functions and Hierarchies, Oxford Logic Guides, vol. 9. Clarendon Press (1984)
25. Sakarovitch, J.: Elements of Automata Theory. Cambridge University Press (2009)
26. Schmitz, S., Schnoebelen, P.: Multiply-Recursive Upper Bounds with Higman's Lemma. In: Aceto, L., Henzinger, M., Sgall, J. (eds.) ICALP 2011, Part II. LNCS, vol. 6756, pp. 441–452. Springer, Heidelberg (2011)
27. Schmitz, S., Schnoebelen, P.: Algorithmic Aspects of WQO Theory. Lecture Notes (2012), http://cel.archives-ouvertes.fr/cel-00727025