Aspects of Layer Systems in IsaFoR*

Bertram Felgenhauer and Franziska Rapp

Department of Computer Science, University of Innsbruck, Austria
{bertram.felgenhauer|franziska.rapp}@uibk.ac.at

Abstract

We report on an ongoing formalization of layer systems in Isabelle.

1 Introduction

Toyama's theorem [6, 9] states that confluence is modular, i.e., that the union of two confluent term rewrite systems (TRSs) over disjoint signatures is confluent if and only if the two TRSs themselves are confluent. This opens up a decomposition approach to proving confluence, which is attractive, because different confluence criteria may apply to the constituent TRSs that do not apply to their union. By adapting the modularity proof, several other results have been proved. For example, confluence is preserved by currying [5]. Layer systems [3] were introduced as an abstraction from these proofs, which work by decomposing terms into a maximal top and remaining aliens. A layer system $\mathfrak{L}$ is simply the set of allowed tops; for modularity, those are homogeneous multi-hole contexts, i.e., multi-hole contexts whose function symbols all belong to the signature of only one of the two given TRSs. At the heart of layer systems lies yet another adaptation of the modularity proof in [6]. The main results correspond to the *if* direction of modularity as stated above. When establishing confluence by layer systems, as remaining proof obligations, one has to check that a layer system satisfies so called layer conditions, which is easier than doing a full adaptation of the modularity proof.

This note describes an ongoing effort to formalize layer systems, which, once complete, will enable certification of confluence proofs based on persistence and currying. In fact, the prospect of formalization was one of the selling points of layer systems; whereas adapting existing proofs is convenient on paper, it becomes a burden when done in a formalization; as with any code duplication in software engineering, it would increase maintenance costs and should therefore be avoided. We use Isabelle [7] for our formalization[1], building on top of IsaFoR [8].

Notation. We use notation from term rewriting. Let $\mathcal{F}$, $\mathcal{V}$ be a signature. Then $\mathcal{T}(\mathcal{F},\mathcal{V})$ is the set of terms over that signature; $\mathcal{C}(\mathcal{F},\mathcal{V})$ is the set of multihole contexts (which may contain occurrences of an extra constant $\square$, denoting holes.) On multihole contexts, we have a partial order $\sqsubseteq$ which is generated by $\square \sqsubseteq C$ and closure under contexts. The corresponding partial supremum operation is denoted by $\sqcup$; intuitively it merges multi-hole contexts.

2 Layer Conditions

Let $\mathfrak{L}$ be a set of multi-hole contexts, which we intend to use for decomposing terms. We recall the definitions of a layer system, and (weakly) layered TRSs.

Definition 1 ([3, Definition 3.1])**.** Let $\mathfrak{L} \subseteq \mathcal{C}(\mathcal{F},\mathcal{V})$ be a set of multi-hole contexts over $\mathcal{F}$. Then $L \in \mathfrak{L}$ is called a *top* of a context $C \in \mathcal{C}(\mathcal{F},\mathcal{V})$ (according to $\mathfrak{L}$) if $L \sqsubseteq C$. A top is a *max-top* of C if it is maximal with respect to $\sqsubseteq$ among the tops of C.

*This work is supported by FWF (Austrian Science Fund) project P27528.

[1] The theories can be viewed at http://cl-informatik.uibk.ac.at/software/lisa/iwc2017/

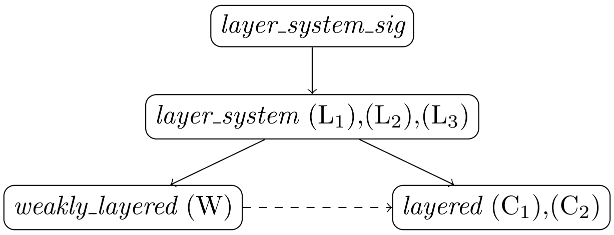

Figure 1: Hierarchy of locales.

Definition 2 ([3, Definition 3.3])**.** Let $\mathcal{F}$ be a signature. A set $\mathfrak{L} \subseteq \mathcal{C}(\mathcal{F}, \mathcal{V})$ of contexts is called a *layer system* if it satisfies properties (L_1), (L_2), and (L_3). The elements of $\mathfrak{L}$ are called *layers*. A TRS $\mathcal{R}$ over $\mathcal{F}$ is *weakly layered (according to a layer system $\mathfrak{L}$)* if condition (W) is satisfied for each $\ell \to r \in \mathcal{R}$. It is *layered (according to a layer system $\mathfrak{L}$)* if conditions (W), (C_1), and (C_2) are satisfied. The conditions are as follows:

(L_1) Each term in $\mathcal{T}(\mathcal{F}, \mathcal{V})$ has a non-empty top.

(L_2) If $x \in \mathcal{V}$ and $C \in \mathcal{C}(\mathcal{F}, \mathcal{V})$ then $C[x]_p \in \mathfrak{L}$ if and only if $C[\Box]_p \in \mathfrak{L}$.

(L_3) If $L, N \in \mathfrak{L}$, $p \in \mathcal{P}os_{\mathcal{F}}(L)$, and $L|_p \sqcup N$ is defined then $L[L|_p \sqcup N]_p \in \mathfrak{L}$.

(W) If M is a max-top of s, $p \in \mathcal{P}os_{\mathcal{F}}(M)$, and $s \to_{p,\ell\to r} t$ then $M \to_{p,\ell\to r} L$ for some $L \in \mathfrak{L}$.

(C_1) In (W) either L is a max-top of t or $L = \Box$.

(C_2) If $L, N \in \mathfrak{L}$ and $L \sqsubseteq N$ then $L[N|_p]_p \in \mathfrak{L}$ for any $p \in \mathcal{P}os_{\Box}(L)$.

In Isabelle, we bundle these assumptions in locales [1]. For example, the first three layer conditions are formalized as follows:

```
locale layer_system_sig = fixes 𝓕 :: 'f sig and 𝔏 :: ('f, 'v) mctxt set

locale layer_system = layer_system_sig 𝓕 𝔏 for 𝓕 :: 'f sig and
  𝔏 :: ('f, 'v :: infinite) mctxt set +
  assumes 𝔏_sig: 𝔏 ⊆ 𝓒
  and L₁: t ∈ 𝓣 ⟹ ∃ L ∈ 𝔏. L ≠ MHole ∧ L ≤ mctxt_of_term t
  and L₂: p ∈ poss_mctxt C ⟹
    mreplace_at C p (MVar x) ∈ 𝔏 ⟷ mreplace_at C p MHole ∈ 𝔏
  and L₃: L ∈ 𝔏 ⟹ N ∈ 𝔏 ⟹ p ∈ funposs_mctxt L ⟹
    (subm_at L p, N) ∈ comp_mctxt ⟹ mreplace_at L p (subm_at L p ⊔ N) ∈ 𝔏
```

The first locale, *layer_system_sig*, is used to define $\mathcal{T}$ and $\mathcal{C}$, the set of terms and multi-hole contexts over $\mathcal{F}$, and the concept of max-tops. Actually max-tops are defined separately for terms and for multi-hole contexts, because while on paper, multi-hole contexts are just terms, in IsaFoR they have their own type. In total, four locales are defined, capturing the layer conditions, cf. Figure 1. Note that condition (W) is not part of the *layered* locale; it would be redundant because (C_1) implies (W). In Isabelle we have encoded this fact by proving that *layered* is a sublocale of *weakly_layered*, as indicated by the dashed arrow.

Using the layer system to decompose terms from the top yields the following notion of rank.

Definition 3 ([3, Definition 3.6])**.** Let $t = M[t_1, \dots, t_n]$ with M the max-top of t. We define $\mathrm{rank}(t) = 1 + \max\{\mathrm{rank}(t_i) \mid 1 \leqslant i \leqslant n\}$, where $\max(\varnothing) = 0$ ($t_1, \dots, t_n$ are the *aliens* of t).

The main theorems of [3] can be stated as follows (we omit [3, Theorem 4.3] to save space).

Theorem 4 ([3, Theorem 4.1])**.** *Let $\mathcal{R}$ be a weakly layered TRS that is confluent on terms of rank one. If $\mathcal{R}$ is left-linear then $\mathcal{R}$ is confluent.*

Theorem 5 ([3, Theorem 4.6])**.** *Let $\mathcal{R}$ be a layered TRS that is confluent on terms of rank one. Then $\mathcal{R}$ is confluent.*

Within the formalization, Theorem 4 will be established inside the *weakly_layered* locale, whereas Theorem 5 is expected to hold in the *layered* locale. The proofs of these main results correspond to Section 4 of [3], which we have fully formalized up to Lemma 4.18. The section goes up to Lemma 4.36, so a big chunk remains to be done. Nevertheless, we could already work on the applications like modularity and currying, because they are merely instantiations of these locales, which can be established independently of the main results.

3 Currying

Here we consider currying as one application of the layer framework, which we formalized in Isabelle. Instead of applying functions to several arguments at once, currying introduces a binary function symbol that is used for applying arguments to functions one by one. In functional programming, currying turns a function of type $A_1 \times \cdots \times A_n \to B$ into one of type $A_1 \to \cdots \to A_n \to B$, enabling partial application. For term rewriting systems (TRSs) we introduce a fresh function symbol $\bullet$ to denote application, whereas every other function symbol becomes a constant. By convention we write f_n to denote a function symbol of arity n. Moreover, we denote the arity of a function symbol f with respect to the signature $\mathcal{F}$ by $\mathsf{a}_{\mathcal{F}}(f)$. We identify $f_{\mathsf{a}_{\mathcal{F}}(f)}$ with f.

Definition 6. Given a TRS $\mathcal{R}$ over a signature $\mathcal{F}$, its *curried version* $\mathsf{Cu}(\mathcal{R})$ consists of rules $\{\mathsf{Cu}(l) \to \mathsf{Cu}(r) \mid \ell \to r \in \mathcal{R}\}$, where $\mathsf{Cu}(t) = t$ if t is a variable and $\mathsf{Cu}(f(t_1, \dots, t_n)) = f_0 \bullet \mathsf{Cu}(t_1) \bullet \cdots \bullet \mathsf{Cu}(t_n)$. Here $\bullet$ is a fresh left-associative function symbol.

Currying is useful for deciding properties such as confluence [2] or termination [4]. For analyzing confluence by currying, the following result is important.

Theorem 7. *Let $\mathcal{R}$ be a TRS. If $\mathcal{R}$ is confluent, then $\mathsf{Cu}(\mathcal{R})$ is confluent.*

This result was proved by Kahrs [5]. Rather than working directly with $\mathsf{Cu}(\mathcal{R})$, Kahrs works with the *partial parametrization* of $\mathcal{R}$, which is given by $\mathsf{PP}(\mathcal{R}) = \mathcal{R} \cup \mathcal{U}_{\mathcal{F}}$, where $\mathcal{U}_{\mathcal{F}}$ is the set of uncurrying rules for $\mathcal{F}$ (see Definition 8). Confluence of $\mathsf{PP}(\mathcal{R})$ and $\mathsf{Cu}(\mathcal{R})$ are closely related, cf. Lemma 9.

Definition 8. Given a signature $\mathcal{F}$, the *uncurrying* rules $\mathcal{U}_{\mathcal{F}}$ are rules $f_i(x_1, \dots, x_i) \bullet x_{i+1} \to f_{i+1}(x_1, \dots, x_{i+1})$ for every function symbol $f \in \mathcal{F}$ and $1 \leqslant i < \mathsf{a}_{\mathcal{F}}(f)$.

Lemma 9 ([5, Proposition 3.1])**.** *Let $\mathcal{R}$ be a TRS. Then $\mathsf{Cu}(\mathcal{R})$ is confluent if $\mathsf{PP}(\mathcal{R})$ is.*

Hence in order to prove Theorem 7 it suffices to prove that $\mathsf{PP}(\mathcal{R})$ is confluent. To this end, we make use of Theorem 5. Hence we need to show that $\mathsf{PP}(\mathcal{R})$ is layered according to some set of layers $\mathfrak{L}$ and confluent on terms of rank one. First of all we have to define a suitable set of layers. We choose $\mathfrak{L} = \mathfrak{L}_1 \cup \mathfrak{L}_2$ where $\mathfrak{L}_1$ is the smallest extension of $\mathcal{V}_\Box = \mathcal{V} \cup \{\Box\}$ such that $f_m(s_1, \ldots, s_m) \bullet s_{m+1} \bullet \cdots \bullet s_n \in \mathfrak{L}_1$ for all $f \in \mathcal{F}$, $1 \leqslant m \leqslant n \leqslant \mathsf{a}_\mathcal{F}(f)$ and $s_1, \ldots, s_n \in \mathfrak{L}_1$, and $\mathfrak{L}_2 = \{x \bullet t \mid x \in \mathcal{V}_\Box \text{ and } t \in \mathfrak{L}_1\}$. This definition realizes a separation between well-formed terms ($\mathfrak{L}_1$), whose $\mathcal{U}_\mathcal{F}$-normal form contains no $\bullet$ symbol, and ill-formed terms ($\mathfrak{L}_2$), whose $\mathcal{U}_\mathcal{F}$-normal form contains exactly one $\bullet$ symbol at the root. As required for condition (L_1), variables and holes are treated interchangeably.

Whereas for Lemma 9 we could follow the lines of the paper proof, the formalization of the fact that $\mathsf{PP}(\mathcal{R})$ is layered according to $\mathfrak{L}$ turned out to be much more tedious. First of all, we found it convenient to define functions that *compute* the max-top of a term, since the abstract definition of max-tops in the layer framework is not really suitable for proofs in Isabelle.

Definition 10. The following function checks whether the number of arguments applied to the first non-$\bullet$ function symbol f is at most the arity $\mathsf{a}_\mathcal{F}(f)$ according to the original signature $\mathcal{F}$

$$\mathsf{check}(t, m) = \begin{cases} \mathit{false} & \text{if } t \in \mathcal{V} \\ \mathsf{check}(t_1, m+1) & \text{if } t = t_1 \bullet t_2 \\ \mathsf{a}_\mathcal{F}(f) \geqslant m + n & \text{if } t = f_n(t_1, \ldots, t_n) \end{cases}$$

Let $\mathcal{F}^\bullet = \mathcal{F} \cup \{\bullet\}$. The max-top mt_Cu of a term $t \in \mathcal{T}(\mathcal{F}^\bullet, \mathcal{V})$ with respect to $\mathfrak{L}$ is defined as

$$\mathsf{mt}_\mathsf{Cu}(t) = \begin{cases} \mathrm{t} & \text{if } t \in \mathcal{V} \\ f(\mathsf{mt}_1(t_1, 0), \ldots, \mathsf{mt}_1(t_n, 0)) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } (\mathsf{check}(t, 0) \text{ or } t_1 \in \mathcal{V}) \\ \Box \bullet \mathsf{mt}_1(t_2, 0) & \text{otherwise (in which case } t = t_1 \bullet t_2) \end{cases}$$

Here $\mathsf{mt}_1(t, m)$ computes the max-top of t with respect to $\mathfrak{L}_1$, where m is the number of already applied arguments:

$$\mathsf{mt}_1(t, m) = \begin{cases} \mathrm{t} & \text{if } t \in \mathcal{V} \\ \mathsf{mt}_1(t_1, m+1) \bullet \mathsf{mt}_1(t_2, 0) & \text{if } t = t_1 \bullet t_2 \text{ and } \mathsf{check}(t, m) \\ f(\mathsf{mt}_1(t_1, 0), \ldots, \mathsf{mt}_1(t_n, 0)) & \text{if } t = f(t_1, \ldots, t_n), f \neq \bullet \text{ and } \mathsf{check}(t, m) \\ \Box & \text{otherwise} \end{cases}$$

Note that there is some redundancy, since the check function does the same counting several times. However the definition is easier like this.

After proving the correctness of mt_1 and mt_Cu, the main difficulty was the proof of condition (C_1) for $\mathfrak{L}$ and $\mathsf{PP}(\mathcal{R})$. We sketch a constructive proof here, since this gives the best intuition and is also easiest to formalize in our opinion. In order to establish (C_1), we need to analyze a rewrite step $s = C[l\sigma]_p \to C[r\sigma]_p$ with p a function position of the max-top M of s; our goal is to obtain a rewrite step $M = D[l\sigma']_p \to D[r\sigma']_p$. The following two lemmas allow pushing the computation of the max-top (using mt_1) all the way to the substitution σ.

Lemma 11. *Let s be a term and p the hole position of context C such that $C[s]_p \in \mathcal{T}(\mathcal{F}^\bullet, \mathcal{V})$ and $p \in \mathcal{P}os_{\mathcal{F}^\bullet}(\mathsf{mt}_1(C[s], j))$. Then there exists a context D and a natural number k such that $\mathsf{mt}_1(C[s], j) = D[\mathsf{mt}_1(s, k)]$, and $\mathsf{mt}_1(C[t], j) = D[\mathsf{mt}_1(t, k)]$ for any term $t \in \mathcal{T}(\mathcal{F}^\bullet, \mathcal{V})$ having the same number of missing arguments as s.*

Lemma 12. *Let $t \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. Then $\mathsf{mt}_1(t \cdot \sigma, 0) = \mathsf{mt}_1(t, 0) \cdot \sigma'$ with $\sigma' = (\lambda x.\, \mathsf{mt}_1(x, 0)) \circ \sigma$.*

Using these two lemmas, we can obtain the desired rewrite step from M by the following computation, where for simplicity we only consider the case $M \in \mathfrak{L}_1$ and $l \to r \in \mathcal{R}$:

$$M = \mathsf{mt}_{\mathsf{Cu}}(s) = \mathsf{mt}_1(C[l \cdot \sigma], 0) \overset{11}{=} D[\mathsf{mt}_1(l \cdot \sigma, k)] \overset{12}{=} D[\mathsf{mt}_1(l, 0) \cdot \sigma'] = D[l \cdot \sigma']$$

$$\to_{p,\ell\to r} D[r \cdot \sigma'] = D[\mathsf{mt}_1(r, 0) \cdot \sigma'] \overset{12}{=} D[\mathsf{mt}_1(r \cdot \sigma, k)] \overset{11}{=} \mathsf{mt}_1(C[r \cdot \sigma], 0)$$

The uses of the previous two lemmas are indicated above the equalities. Note that the number of missing arguments of r is the same as for l, so we can use Lemma 11 in both directions. Furthermore $k = 0$, because $\mathsf{mt}_1(l \cdot \sigma, k) = \square$ otherwise, so the rewrite step would not take place at a function position of M. Hence Lemma 12 is applicable. Furthermore, we use $\mathsf{mt}_1(l, 0) = l$ ($\mathsf{mt}_1(r, 0) = r$), using that l (r) is well-formed. If $C = \square$, r is a variable and $\mathsf{check}(r \cdot \sigma)$ is false, $\mathsf{mt}_1(C[r \cdot \sigma], 0) = \square$. Otherwise, the max-top of $C[r \cdot \sigma]$ is equal to $\mathsf{mt}_1(C[r \cdot \sigma], 0)$.

4 Conclusion

We have presented some aspects of a formalization of layer systems in Isabelle. Let us conclude with some statistics. In [3], the setup of the layer systems and the proof of the main results up to Lemma 4.18 spans approximately 150 lines of text. The corresponding formalization is about 3000 lines in length, corresponding to a *de Bruijn factor* of about 20. The section on currying in the original paper [3, Section 5.4], containing the proof of Theorem 7, covers about 80 lines (including 2 intermediate results in Kahrs [5]). Whereas only 9 definitions were needed to prepare for the proof, in the Isabelle formalization 24 definitions (abbreviations counted half) and 122 lemmas were necessary. Overall the formalization spans about 3200 lines of Isar code, which implies a de Bruijn factor of approximately 40. This high factor may be due to the fact that many case distinctions on the shape of terms were necessary and counting the number of applied arguments was tedious. Moreover, the formalization distinguishes terms from multi-hole contexts and hence several conversions between those types were necessary. Since the ultimate goal of our formalization effort is the certification of confluence proofs exploiting currying, we plan to finish the formalization of the layer framework and formalize further applications, most notably persistence of many-sorted TRSs.

References

[1] C. Ballarin. Locales: A module system for mathematical theories. *JAR*, 52(2):123–153, 2014.

[2] B. Felgenhauer. Deciding confluence of ground term rewrite systems in cubic time. In *Proc. 23rd RTA*, volume 15 of *LIPIcs*, pages 165–175, 2012.

[3] B. Felgenhauer, A. Middeldorp, H. Zankl, and V. van Oostrom. Layer systems for proving confluence. *ACM TOCL*, 16(2:14):1–32, 2015.

[4] N. Hirokawa, A. Middeldorp, and H. Zankl. Uncurrying for termination. In *Proc. 15th LPAR*, pages 667–681, 2008.

[5] S. Kahrs. Confluence of curried term-rewriting systems. *JSC*, 19(6):601–623, 1995.

[6] J.W. Klop, A. Middeldorp, Y. Toyama, and R. de Vrijer. Modularity of confluence: A simplified proof. *IPL*, 49:101–109, 1994.

[7] T. Nipkow, L.C. Paulson, and M. Wenzel. *Isabelle/HOL – A Proof Assistant for Higher-Order Logic*, volume 2283 of *LNCS*. Springer, 2002.

[8] R. Thiemann and C. Sternagel. Certification of termination proofs using CeTA. In *Proc. 22nd TPHOLs*, volume 5674 of *LNCS*, pages 452–468, 2009.

[9] Y. Toyama. On the Church-Rosser property for the direct sum of term rewriting systems. *JACM*, 34(1):128–143, 1987.