Feebly not weakly

Vincent van Oostrom
Utrecht University
Philosophy
Utrecht, The Netherlands
v.vanoostrom@uu.nl

Some rewrite systems are not orthogonal, in that they do have critical peaks, but are very close to being orthogonal, in that for any given object there exists a partial function, called orthogonalisation, mapping the set of all redexes to an orthogonal subset and every multi-step to an equivalent one. Term rewrite systems having only trivial peaks, so-called weakly orthogonal systems with the $\lambda\beta\eta$-calculus as prime example, are known to admit such an orthogonalisation. Here we characterise the term rewrite systems that admit orthogonalisation as those whose critical peaks are feeble, in that at least two out of the three terms in such a peak must be identical (generalising weak orthogonality).

Introduction The $\lambda\beta\eta$-calculus is not orthogonal since there are *overlap* peaks, i.e. peaks that do not arise from multi-steps. In particular, the overlap peaks $@(F,G) \leftarrow_\beta @(\lambda(x.@(F,x)),G) \rightarrow_\eta @(F,G)$ and $\lambda(y.F(y)) \leftarrow_\eta \lambda(x.@(\lambda(y.F(y)),x)) \rightarrow_\beta \lambda(x.F(x))$ taken from [3, Example 4.10], are *critical* since they are minimal with respect to encompassment. Still, looking at these critical peaks one notes that they are *trivial* in that the targets of the steps in each of them are the same, both $@(F,G)$ for the former peak and both $\lambda(y.F(y))$ (up to α-equivalence) for the latter. One could say that in each case the β- and η-steps are extensionally the same but not intensionally so. Term rewrite systems such as the $\lambda\beta\eta$-calculus that are left-linear and have only trivial critical peaks are called *weakly* orthogonal and are known to share many desirable properties with the orthogonal ones, extensional properties such as confluence, multi-steps having the triangle property, and the full-substitution strategy being cofinal. However, there are also properties that do hold in the orthogonal but fail in the weakly orthogonal case, intensional properties such as head-normalisation of outermost-fair strategies, having finitely many permutation equivalent reductions to normal form, and the unique normal form property in case of infinitary term rewriting.[1]

This paper aims to contribute to clarifying what separates both by considering the *orthogonalisation* property, i.e. whether all the multi-steps from a given term can be generated (see Figure 1, cf. [2][2]) from an orthogonal subset of its redexes in this way linking extensional to intensional orthogonality. For instance, all multi-steps from $\lambda(\underline{x.\overline{@(\underline{\lambda(y.@(z,y))},x)}})$ can be generated from the subset comprising only the two underlined η-redexes: the step induced by the overlined β-redex is extensionally the same as the step induced by either η-redex (see also Figure 2). The extensional properties mentioned above factor through orthogonalisation, a satisfactory state of affairs as it is known [1] that weakly orthogonal term rewriting systems admit orthogonalisation.

One may then wonder what rewrite systems admit orthogonalisation and whether it is a decidable property. Clearly the non-weakly-orthogonal TRS with rules $a \to a$ and $a \to b$ admits orthogonalisation, as the first rule is trivial so redexes for it can simply be omitted (steps for that rule can be simulated by the empty multistep). We show the phenomenon displayed in this example is in fact the only extension to

[1] The notion of strong convergence is an intensional property; it refers to depth.

[2] Note however that that paper considers, more generally, instead of multi-steps arbitrary reductions and, less generally, instead of arbitrary left-linear rewrite systems only orthogonal ones rendering the issues dealt with here non-issues.

Submitted to:
HOR 2014

© V. van Oostrom
This work is licensed under the
Creative Commons Attribution License.

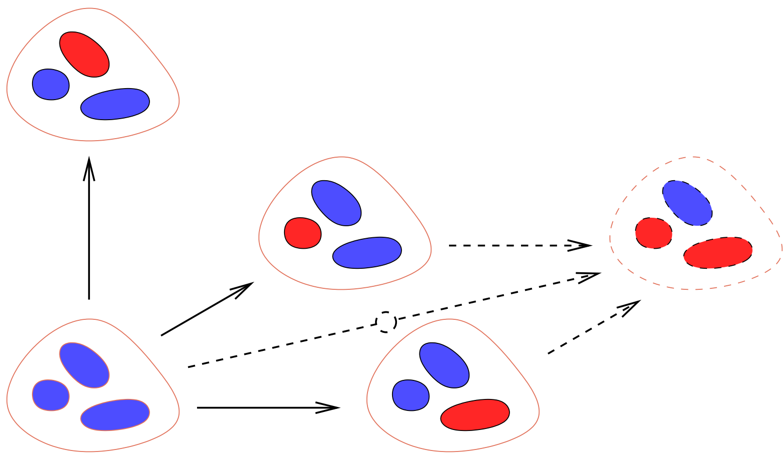

Figure 1: Generating multi-steps from orthogonal steps, by disjoint union

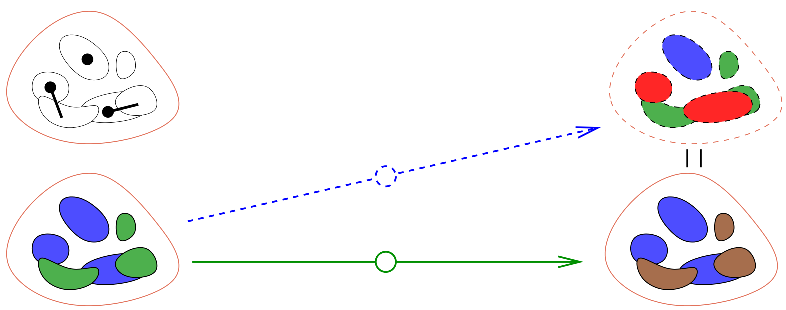

Figure 2: Orthogonalising (simulating all patterns by blue ones as symbolised by the bullet-headed-arrows in the top–left) the bottom multi-step (contracting green patterns into brown) yielding the diagonal multi-step (contracting blue pattern into red), which is extensionally the same (they have the same targets)

weak orthogonality needed to characterise orthogonalisation exactly. More precisely, calling a left-linear term rewrite system *feebly* orthogonal if for every irreducible critical peak $s \leftarrow t \rightarrow r$ at least two out of the thee terms s,t,r in the peak are the same, where irreducible means that the rules involved do not encompass other rules, we show a term rewrite system admits orthogonalisation if and only if it is feebly orthogonal. We establish this result for the class GHRS of *geometric* higher-order term rewrite systems, a subclass of the pattern rewrite systems [3] introduced here, where patterns [4, 5] are further restricted to be linear and *convex*, the idea roughly being that that function symbols occurring in (the Böhm tree of) the pattern can only have other such (no variables!) inbetween (see Figure 3). We assume familiarity with and notation from [7, Chapter 11].

Geometric unification We introduce geometric terms and study their unification.

Definition 1. Terms *are simply typed* λ*-terms modulo* $\alpha\beta\eta$ *over a higher-order signature. We assume term to be in* η*-long form. A term is a* pattern *if every free variablein it occurs as applied to a list of pairwise distinct bound variables in a term of base type. For patterns we assume the bound variables in this list to be in* η*-short form. A term is* convex *if it is a pattern* and every active occurrence of a variable

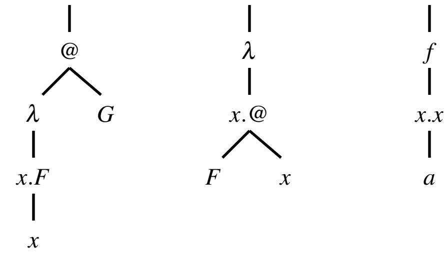

Figure 3: Convexity of the left-hand sides of the β- and η-rules (no bound variable between the function symbols @,λ) and non-convexity of $f(x.x(a))$ (bound variable x between function symbols f,a)

in the pattern is free*, where an occurrence is called* active *if it is the first argument to an application. A* renamer *is a convex term of shape $\vec{x}.F(\vec{y})$. A term is* linear *if each variable occuring freely in it does so exactly once. A term is* geometric *if it is linear and convex. A* geometric *higher-order rewrite system, GHRS, is a PRS [3] for which the left-hand sides of rules are geometric. Notions for terms generalise pointwise to sets of terms. A family of terms is* totally linear *if every free variable occurring in it does so only once.*

We use $t,s,r,\ldots$ to range over terms, $p,q,\ell,g,\ldots$ to range over patterns, $x,y,z,\ldots$ to range over bound variabeles, and $F,G,H,\ldots$ to range over free ones. Total linearity of a family of terms is stronger than linarity of its associated set, e.g., the family comprising F and $f(F)$ is linear (as set) but not totally so (as family).

Remark 2. *Convex terms are patterns and, conversely, all first-order terms and second-order patterns are convex; the former in the absence of bound variables, the latter because the extra constraint convexity puts on bound variables is vacuously satisfied as bound variables in second-order patterns are of base type so cannot be applied. Terms that are not patterns are certainly not convex, e.g. the non-patterns $F(G)$ and $f(x.F(x,x))$ are not convex, but also terms that are patterns may fail to be convex as witnessed by e.g. $x.y.x(y)$ and $f(x.x(a,F))$; the (bound) variable x is active in both. At first approximation one may say convexity corresponds to second-order pattern matching and higher-order parameter passing. We do not know how severe the geometricity restriction is in general, but do note that the class of GHRSs contains both the class of left-linear TRSs, the first-order term rewrite systems, and the class of left-linear CRSs, the combinatory reduction systems. In fact, every higher-order rewrite system given in [6], incuding the example in the introduction here, that is a PRS, is in fact a GHRS.*

Geometric and convex terms being patterns, extant unification algorithms can be deployed on them. We recapitule pattern unification [4] from [5] and show it is better-behaved on geometric terms.

Definition 3. *A* substitution *is a function mapping each variable to a term of the same type. The* domain *of a substitution σ is the set of variables F for which $\sigma(F) \neq F$. The* range *of a substitution is the image of its domain. Notions for terms generalise pointwise to substitutions via their range. The application to a term t of the lifting of the substitution σ from variables to terms, is denoted by t^σ. The* composition *of σ and τ is the substitution $\tau \circ \sigma$ (τ* after *σ) defined by $F \mapsto \sigma(F)^\tau$. A substitution σ is* idempotent *if $\sigma \circ \sigma = \sigma$.* Restricting *a substitution to some set of variables updates it to map variables outside that set to themselves. A substitution is* totally linear on *a set of variables if the corresponding family is so. If it moreover maps those variables to convex terms that are* open*, i.e. containing at least one free variable, the substitution is called a* discriminator on *that set.*

We use σ,τ,υ to range over substitutions, π,θ to range over pattern substitutions, and ρ, ω to denote renamer substitutions, also called simply *renamings*. The proofs of the following are standard.

Proposition 4. *Pattern/convex/renamer substitutions are closed under composition. Linear pattern substitutions are closed under composition if the latter substitution is totally linear on the free variables of the former. A pattern substitution is idempotent iff its set of free variables is disjoint from its domain.*

Remark 5. *Two linear substitutions need not compose to a linear substitution not even in the first-order case as witnessed by, e.g., the linear term $f(F,G)$ and the linear substitution mapping both F and G to H resulting in the non-linear term $f(H,H)$. In the higher-order case, without the pattern restriction even the strengtened requirement of total linearity does not suffice as witnessed, e.g., by the totally linear term $F(G)$ and the linear substitution mapping F to $x.f(x,x)$ resulting in the non-linear term $f(G,G)$. Without the pattern restriction higher-order idempotent substitutions need not have a set of free variables that is disjoint from its domain, e.g., in case of the idempotent substitution $F \mapsto x.F(a)$ both sets are $\{F\}$.*

Substituting discriminable geometric terms into a geometric term is reversible, see Figure 4:

Lemma 6. *For every geometric term p and (arbitrary) term t such that $\mathcal{FV}(p) \supseteq \mathcal{FV}(t)$ and discriminator π on the former, if $p^\pi = t^\pi$, then there exist a renaming ρ and geometric substitution $\overline{\pi}$, such that $p^\rho = t^\rho$ and $\pi = \overline{\pi} \circ \rho$.*

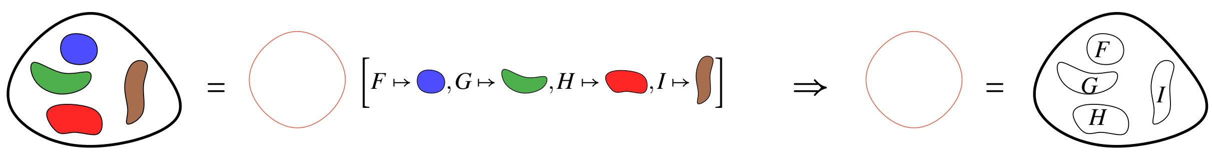

Figure 4: Geometric intuition for Lemma 6: if the result of substituting some discriminable (distinctly coloured) parts contains these uniquely (left), from that result its origin is uniquely reconstructible (right)

Proof. First note that we may assume π is *non-erasing* in that if it maps a free variable F to a term $\vec{x}.\underline{t}$ with $\underline{t}$ of base type, then each bound variable among $\vec{x}$ does occur freely in $\underline{t}$: if, say, x among $\vec{x}$ doesn't, we have $\pi = \underline{\pi} \circ \underline{\rho}$ for $\underline{\rho}$ the identity substitution updated to map F to $\vec{x}.\underline{F}(\underline{\vec{x}})$ and $\underline{\pi}$ identical to π updated to map $\underline{F}$ to $\underline{\vec{x}}.\underline{t}$ for some fresh free variable $\underline{F}$ with $\underline{\vec{x}}$ being obtained from $\vec{x}$ by removing x from it. Since $p^{\underline{\rho}}$ is geometric again, $\mathcal{FV}(p^{\underline{\rho}}) \supseteq \mathcal{FV}(t^{\underline{\rho}})$, and $(p^{\underline{\rho}})^{\underline{\pi}} = p^{\underline{\pi}\circ\underline{\rho}} = p^\pi = t^\pi = t^{\underline{\pi}\circ\underline{\rho}} = (p^{\underline{\rho}})^{\underline{\pi}}$, we conclude by the induction (on the number of such bound variables in the discriminator that do not occur freely) hypothesis for $p^{\underline{\rho}}$, $t^{\underline{\rho}}$ and $\underline{\pi}$ that there exist a renaming ρ and geometric substitution $\overline{\pi}$, such that $(p^{\underline{\rho}})^\rho = (t^{\underline{\rho}})^\rho$ and $\underline{\pi} = \overline{\pi} \circ \rho$, from which we conclude the same for p, t and π by the renaming $\rho \circ \underline{\rho}$ (using closure under composition of renaming, Proposition 4) and geometric substitution $\overline{\pi}$.

Observing that for π non-erasing $\mathcal{FV}(t^\pi) = \mathcal{FV}(\pi(\mathcal{FV}(t)))$, as can be shows by induction on t using that $\mathcal{FV}(q(\vec{t})) = \mathcal{FV}(q) \cup \mathcal{FV}(\vec{t})$ for a non-erasing geometric term q and vector of terms $\vec{t}$, the statement of the lemma is shown to hold by induction on p, distinguishing cases on p,t:

- (flex,flex) if $p = \vec{x}.F(\vec{p})$ and $t = \vec{y}.G(\vec{t})$, then we may assume w.l.o.g. that $\vec{x} = \vec{y}$. That F and G are distinct is not possible, since then by π being a discriminator $\pi(F)$ and $\pi(G)$ could be discriminated by considering their free variables, and since they are geometric terms this is preserved for $\pi(F)(\vec{p}^\pi)$ and $\pi(g)(\vec{t}^\pi)$ (considering free variables at minimal depth). Hence $F = G$ and to apply the induction hypothesis it remains to show that $\mathcal{FV}(p_i) \supseteq \mathcal{FV}(t_i)$ for each i, which must hold since otherwise $\mathcal{FV}(p_i^\pi)$ and $\mathcal{FV}(t_i^\pi)$ could be discriminated by the observation (then $p^\pi \neq t^\pi$).
- (flex,rigid) and (rigid,flex) these cases are shown impossible as in the previous item, by using the free variables occurring (at minimal depth) in the geometric term substituted for the flex-head, to discriminate both.

(rigid,rigid) then both heads must be the same (bound variable or constant) since they are not affected by π, and we conclude by the induction hypothesis for the arguments, as in the 1st case. □

In the first-order case ρ is the identity, e.g. if π maps F to $f(G)$ and G to $g(H)$ in $p = h(f(F), f(G))$, then there is one term t with $\mathcal{FV}(t) \subseteq \{F, G\}$ and $p^\pi = h(f(f(G)), f(g(H))) = t^\pi$, namely p.

Remark 7. *Non-totally-linear geometric substitutions may fail to discriminate; if π maps both F,G to $g(H)$ in $p = f(F,G)$ then there is more than* 1 *possibility for t under the further constraints* (4 *in fact). Similarly, the non-convex (but pattern!) substitution σ mapping F to $x.f(G, x(a))$ in $p = x.F(x)$ allows more than* 1 *possibility for t under the further constraints; except for p for instance also $x.F(y.x(a))$.*

Definition 8. *A* unification problem *is a pair of terms t and s of the same type, and written as $t =^? s$. Notions for terms are extended pointwise to unification problems and lists thereof. A (list of) unification problem(s) is* totally linear *if its family of terms is.*

We use e to range over unification problems and E to range over lists thereof. Although a unification problem is just a pair and will be treated as such formally, the notation $t =^? s$ already suggests that it expresses a question, namely the question whether there exists a substitution σ such that $t^\sigma = s^\sigma$. If such a substitution exists it is called a *unifier* of a (list of) unification problem(s). The pattern unification algorithm [4, 5] transforms states, which combine lists of unification problems and substitutions. The second part of the definition allows to express (the invariants needed for) the stronger property that in the geometric case the unifier computed consists of geometric 'subterms' of the original problem.

Definition 9. *A* state *is a pair $\langle E, \theta \rangle$ with E a list of pattern unification problems and θ an idempotent pattern substitution, such that $\mathcal{FV}(E)$ is disjoint from the domain of θ. The union of both sets with $\mathcal{FV}(\theta)$ is called the* workspace *of the state. The* denotation $\langle\!\langle E, \theta \rangle\!\rangle$ *of the state is the set of all substitutions σ on the workspace that unify E and such that $\sigma = \sigma \circ \theta$. Notions for both components of a state extend naturally to the state itself.*

For (fixed) disjoint (infinite and coinfinite) subsets of left *and* right *variables, a state is* well-split *if (1) E is totally linear; (2) θ is linear; (3)* splitting *variables, i.e. variables that occur freely both in E and θ, may occur in E only in renamers; (4) E may not but θ may contain* unification *variables, i.e. variables that are neither left nor right; and (5) left/right variables only occur freely on the left/right, i.e. either in the left/right term of a problem in E or in the θ-image of a left/right variable.*[3] *For a well-split state its* subspace $(\!|E, \theta|\!)$ *comprises the subterms of base type of the pair $(E^\sigma, \theta^\sigma)$ with σ the* joining *substitution σ mapping H to $\vec{x}.s$, if there is a unification problem for $H(\vec{x})$ and s in E.*

Note that a joining substitution maps variables of one side (left/right) to terms of the other side (right/left). The pattern unification algorithm [4, 5] stepwise transforms a state $\langle E, \{\} \rangle$ into some state $\langle [], \theta \rangle$ in a denotation preserving way by means of transformation steps of shape $\langle e :: E, \theta \rangle \longrightarrow \langle E' \mathbin{@} E^{\theta'}, \theta' \circ \theta \rangle$, if

[3] The notion of occurring on the left or right in a unification problem is not supported by the pattern unfication algorithm in [5]; the left–right order is lost there in the treatment of the flex–rigid and rigid–flex cases. Here we will implicitly assume to have the symmetric rigid–flex version of the flex–rigid rule (3), introducing fresh variables on the right instead of on the left.

$e \to \langle E', \theta' \rangle$for transformation rules, reproduced from [5, Figure 1]:

$$x.s =^? x.t \quad \to \quad \langle [s =^? t], \{\} \rangle \tag{1}$$

$$a(\vec{s}) =^? a(\vec{t}) \quad \to \quad \langle [s_1 =^? t_1, \ldots, s_n =^? t_n], \{\} \rangle \tag{2}$$

$$F(\vec{x}) =^? a(\vec{s}) \quad \to \quad \langle [H_1(\vec{x}) =^? s_1, \ldots, H_n(\vec{x}) =^? s_n], \{F \mapsto \vec{x}.a(\overrightarrow{H(\vec{x})})\} \rangle \tag{3}$$

where F not free in $\vec{s}$ and a a symbol or among $\vec{x}$

$$F(\vec{x}) =^? F(\vec{y}) \quad \to \quad \langle [], \{F \mapsto \vec{x}.H(\vec{z})\} \rangle \tag{4}$$

where $\vec{z}$ is the vector of x_i's for which $x_i = y_i$

$$F(\vec{x}) =^? G(\vec{y}) \quad \to \quad \langle [], \{F \mapsto x.H(\vec{z}), G \mapsto y.H(\vec{z})\} \rangle \tag{5}$$

where $F \neq G$ and $\vec{z}$ the intersection as set of $\vec{x}$ and $\vec{y}$

Linearity and convexity allow to strengthen the correctness statement of unification [5, Theorem 3.1]:

Theorem 10. *A list of pattern unification problems E has a solution iff $\langle E, \{\} \rangle \longrightarrow^* \langle [], \theta \rangle$, in which case $\theta|_{\mathcal{FV}(E)}$ is a most general unifier of E that is*

- *an idempotent pattern substitution;*
- *a convex substitution, if E is a list of convex unification problems; and*
- *a linear substitution, if E is linear, and then it maps free variables occurring on the left/right in E to a renaming of either itself or of a right/left subterm.*

Proof. Transformation steps preserve being a (well-split) state and denotation, and reflect subspace. □

Definition 11. *A redex in a term t is* critically trivial *if it occurs as one of the redexes in a critical peak (that occurs) in t and its induced step in that critical peak is trivial (has the same source and target).*

The action of a critically trivial step is trivial on its 'open' variables, see Figure 5.

Lemma 12. *A step induced by a critically trivial redex in a GHRS has shape $\ell^\pi \to r^\pi$ for some rule $\ell \to r$ and some totally geometric substitution π such that $\ell^\pi = r^\pi$, and there exists a renaming ρ such that $\pi = \overline{\pi} \circ \rho \circ \underline{\pi}$ and $\ell^{\rho \circ \underline{\pi}} = r^{\rho \circ \underline{\pi}}$ with $\overline{\pi}$ the restriction of π to closed terms.*

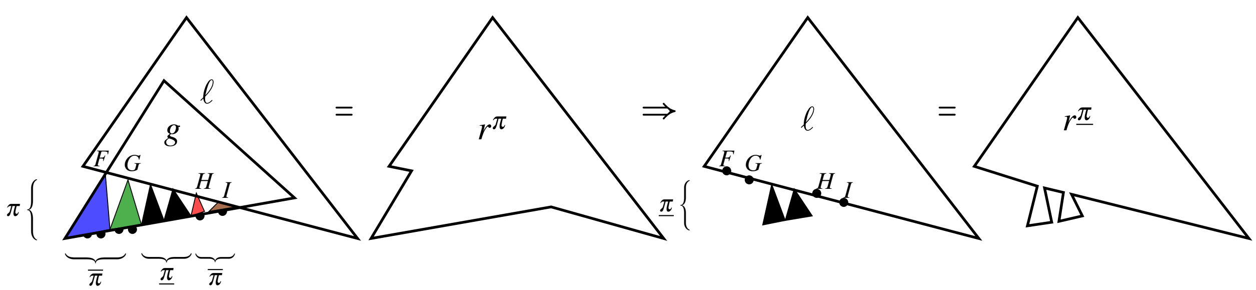

Figure 5: If a step in a critical peak is trivial, it remains so restricting its instantiation to closed terms

Proof. The 1st part follows by definition and by Theorem 10, as left-hand sides of rules of GHRSs are assumed to be geometric. The 2nd part follows from Lemma 6 for ℓ, r, π', after splitting the variables into those mapped by π to closed terms, putting them into $\underline{\pi}$, and the others, into π', yielding $\pi = \pi' \circ \underline{\pi}$. □

Remark 13. *The action of the (non-critically) trivial step obtained by instantiating both F and G with $g(H)$ in the rule $f(F,G) \to f(G,F)$, is non-trivial.*

Orthogonalisation vs. feebly orthogonality We show these conditions to be equivalent.

Definition 14. *An* orthogonalisation *[1, Def. 7.16] of a term t is a partial function $\perp$ from the set W_t of all redexes of t to itself, such that $W_t^{\perp}$ is a multi-redex and $t \multimap_{U^{\perp}} s$ for any multi-step $t \multimap_U s$, where superscripting is used to denote application of $\perp$ lifted to sets of redexes. A GHRS* admits *othogonalisation if every term has an orthogonalisation.*

A rule is redundant *if its source and target encompass the source and target of another rule, by the same context and substitution, and* irredundant *otherwise. A peak is (*ir*)*redundant *if (n)either of its rules is. A peak $s \leftarrow t \to r$ is* feeble *if $|\{s,t,r\}| \leq 2$, and* trivial *if $s = r$. A GHRS is* feebly orthogonal *if it is left-linear and all its irredundant critical peaks are feeble.*

Examples 15. *All orthogonal and weakly orthogonal TRSs admit orthogonalisation [7, Thm. 8.8.27] or [1, Fig. 12] and are feebly orthogonal by definition.*

(1) The TRS $\{f(a,x) \to b, g(f(x,a)) \to g(f(x,x))\}$ is feebly orthogonal: the left step of its only critical peak $g(f(a,a)) \leftarrow g(f(a,a)) \to g(b)$ is trivial. It admits orthogonalisation: let redexes for the left step in the critical peak be undefined, and let $\perp$ be the identity on other redexes.

(2) The TRS $\{a \to b, f(x) \to a', f(a) \to a'\}$ is feebly orthogonal: although its only critical peak $a' \leftarrow f(a) \to f(b)$ is not feeble as $|\{a', f(a), f(b)\}| = 3$, the TRS is *feebly orthogonal since one of the rules ($f(a) \to a'$) used in the critical peak is redundant; a substitution-instance of the rule $f(x) \to a'$.*

(3) The TRS $\{f(x,y) \to f(y,x), g(f(a,a)) \to b\}$ is feebly orthogonal: its critical peak $b \leftarrow g(f(a,a)) \to g(f(a,a))$ is feeble since $|\{b, g(f(a,a)), g(f(a,a))\}| = 2$; the inner step, for the 1st rule, has $g(f(a,a))$ both as source and as target, so is trivial. Note that although the step *is trivial, the* rule *is not; omitting the 1st rule from the TRS would change its induced convertibility.*

We leave it to the reader to check the 2nd, 3rd TRSs admit orthogonalisation. This is no coincidence.

Theorem 16. *A GHRS admits orthogonalisation if and only if it is feebly orthogonal.*

Proof. It suffices to establish the result for GHRSs without redundant or trivial rules, since both properties are preserved and reflected by removing such rules when present. Then, the if-direction is proven by induction on the source of a critical peak using the orthogonalisation for the special case of its doubleton of steps. For the only-if direction we first let $\perp$ map all critically trivial redexes in the term t to undefined. Correctness of this part follows from Lemma 12, as it entails that for any multi-step $t \multimap_U s$ there is another such $t \multimap_{\underline{U}} s$ with $\underline{U}$ the sub-multistep, cf. [7, Def. 8.2.33], obtained from U by discarding its critically trivial steps. The remaining set of redexes constitutes a set of weakly orthogonal redexes (if there is a critical peak between two of them, then they induce the same step), and on these we let $\perp$ be the usual weakly orthogonal orthogonalisation [1, Fig. 12] or [7, Thm. 8.8.27]. □

References

[1] J. Endrullis, C. Grabmayer, D. Hendriks, J.W. Klop & V. van Oostrom: *Unique Normal Forms in Infinitary Weakly Orthogonal Rewriting*. 29 pp. To appear in the special issue of LMCS for RTA 2010.

[2] Z. Khasidashvili & J.R.W. Glauert (1997): *The Geometry of Orthogonal Reduction Spaces*. In: *ICALP'97*, Springer-Verlag, pp. 649–659.

[3] R. Mayr & T. Nipkow (1998): *Higher-order rewrite systems and their confluence*. *TCS* 192(1), pp. 3–29.

[4] D. Miller (1991): *Unification of simply typed lambda-terms as logic programming*. In: *CLP*, pp. 253–281.

[5] T. Nipkow (1993): *Functional Unification of Higher-Order Patterns*. In: *8th LICS*, pp. 64–74.

[6] F. van Raamsdonk (2014): *Examples of Higher-Order Rewriting*. http://www.cs.vu.nl/ femke/ps/examples.ps.

[7] Terese (2003): *Term Rewriting Systems*. Cambridge Tracts in Theoretical Computer Science 55, CUP.