Paths in the lambda-calculus

Three years of communications without understanding

A. Asperti
University of Bologna
Bologna

V. Danos
University Paris 7-CNRS
Paris

C. Laneve
INRIA-CMA
Sophia-Antipolis

L. Regnier
CNRS
Marseille

Abstract

Since the rebirth of λ-calculus in the late sixties, three major theoretical investigations of β-reduction were undertaken: 1) Lévy's analysis of families of redexes (and the associated concept of labeled reductions); 2) Lamping's graph-reduction algorithm; 3) Girard's geometry of interaction.

All these three studies happened to make crucial (if not always explicit) use of a notion of path. Namely and respectively: legal, consistent and regular paths.

Now, these three different notions stand in no obvious relation at first sight. In this paper we prove they are equivalent.

1 Introduction

Let us first survey the three different notions of paths which we want to prove are equivalent.

Lévy took hold in [12] of the difficult notion of two redexes being created in the "same" way during a reduction (in which case they were said to belong to the same "family"). Then he labeled terms and made beta reductions act on labels so that two redexes were in the same family iff they had the same labels.

Labels then slept fifteen years before the awakening in [2] where they were identified with *legal paths*. More precisely: 1) labels of redexes in any reduct N of M denote paths in M; 2) those paths are legal; 3) conversely, any legal path in M denotes a label of a redex to appear somewhere in the set of reducts of M. Legality is a simple and effective condition that intuitively asks for enough symmetry in the path so that the reduction may unfold it into a redex.

In the meantime, people were seeking for a shared reduction faithfully implementing the notion of families, i.e., a reduction where families could be said to be reduced in one step. Such a reduction was discovered by Lamping in [11], and also by Kathail in [10] (important subsequent simplifications were given in [9] and [1]). The invariants which were used to prove the correctness of Lamping's implementation were *consistent paths.*

Finally Girard unveiled in [8] an interpretation of the cut-elimination procedure for linear logic. Again the alternative computation could be defined as the computation of a particular set of paths on proofs, namely *regular paths* which were defined through an algebraic and computational device the *dynamic algebra* (see [5, 14] where this is also extended to pure lambda-calculus).

A fact to marvel at, is that none of the three conditions above seems to bear any relation with beta-reduction. There is yet another equivalent definition which may be the most natural one, but which is also the most uneffective one: *persistent paths.* Call a path persistent if its residuals through any reductions are still connected. In [7] this fourth condition is shown to be equivalent to regularity hence to all the three with which we deal in this paper.

Apart from the satisfaction gained in knowing that there is essentially one notion of paths in lambda-calculus, we also expect these studies to yield some new insights about the (implementation of) beta-reduction based on the unification of the different perspectives.

A word about linear logic. This paper could have been written entirely in the framework of proof-nets. Actually, since proof-nets are a graphical syntax based on a duality (the linear negation) and have a nice geometrical structure given by their correctness condition (the trips), they are much more appealing for working out these investigations on paths. We choose to stick to lambda-calculus because two of the three related works were done for it. However the results presented here may be transposed to linear logic and proof-nets without any difficulty. This point is important since other calculi may be encoded in proof-nets; for instance lambda-calculus-like systems for classical logic such as the $\lambda\mu$-calculus of Parigot [13].

2 The graphical representation of λ-terms

We deal with a graph representation of λ-terms which unifies in a single shot the usual representation (referred to as the *Bourbarki representation*, [3]) linking the bound variables to their lambda, the dynamic graph [5, 14, 6] and the sharing graph representation as defined in [1]. Our graphs are unoriented but have a natural orientation defined on the figures below. The edges are labeled by some *weight* belonging to the dynamic algebra (see section 4). Each node m has a *depth* (an *index* in sharing graphs terminology) which is a positive integer marking m. Similarly an edge is at depth n if its final node (w.r.t. the natural orientation) is at depth n and the depth of a path is the smallest depth of the nodes it traverses. Each node is defined together with an associated set of entering edges (w.r.t. the natural orientation) called its *premises*. There are three kind of nodes:

Communication nodes: the *context (*c*)* and the *variable (*v*)* nodes are zero-ary, i.e., have no associated premises;

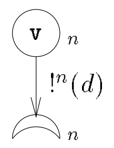
Multiplicative nodes: the *application* and the *lambda* nodes are binary, i.e., have two associated premises which are labeled respectively by $!^k(p)$ and $!^k(q)$ for some k;

Exponential nodes: in sharing graph terminology the *control* nodes; the *croissant* is unary, i.e., has one premise which is labeled by $!^k(d)$ where k is the depth of the node; the *bracket* is unary and its premise is labeled by $!^k(t)$ where k is the depth of the node; the *fan* is binary and its premises are labeled respectively by $!^k(r)$ and $!^k(s)$ where k is the depth of the node; the *weakening* is zero-ary.

For each integer n and each term M we define by induction on M a graph $\mathcal{G}_n(M)$ together with its *boundary* consisting in a single root node (represented on the figure as the topmost node) and a set of control nodes, the *free nodes* of $\mathcal{G}_n(M)$ in one-to-one correspondence with the free variables of M (represented as the bottomost nodes). The translation of a lambda-term M is defined to be $\mathcal{G}(M) = \mathcal{G}_0(M)$.

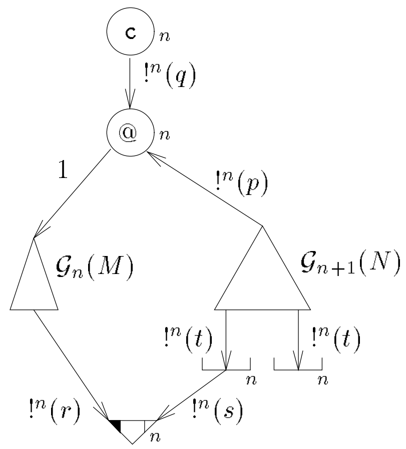
Variable x.

$\mathcal{G}_n(x) =$



Application (MN). Only free nodes corresponding to free variables both in M and N are linked together by means of a fan:

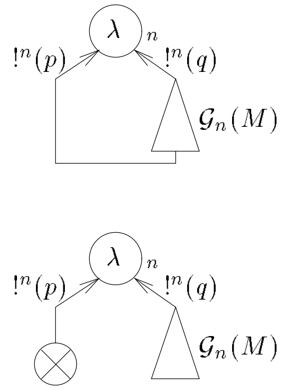
$\mathcal{G}_n(MN) =$



The premises of the application are respectively called the context and the argument edges; the argument node of the application is the initial node of the argument edge. The exiting edge is called the function edge and its final node the function node of the application. Note that the argument node is also the root node of $\mathcal{G}_{n+1}(N)$ whereas the function node is the root node of $\mathcal{G}_n(M)$.

Abstraction $\lambda x.M$. There are two cases whether the variable does or does not appear in M. In the former we link the corresponding node to the lambda node; in the latter we link a weakening node to the lambda node:

$\mathcal{G}_n(\lambda x M) =$



The premises of the lambda are respectively called the variable and the body edges. The initial node of the body edge is the body node of the abstraction.

In the definition of $\mathcal{G}(M)$ we have mentionned a top-down orientation of the graph; this is not meaningful in pure graph theoretic terms (it depends on the drawing). We use it though: the root node is the *topmost* node of the graph; the variable edge of a lambda nodes is bend, it moves *downwards* then up to the lambda. The reason why we feel quiet about it is that there is a way of formalizing "being below" in a graph theoretic manner by means of the correctness condition for proof-nets.

While we are on with proof-nets, let us note that if @ is an application node in $\mathcal{G}(M)$ at depth n, then the set of nodes below @ at depth strictly greater than n corresponds exactly to a !-*box* in the translation of M into a net. Also note that this set is of the form $\mathcal{G}_{n+1}(N)$ where N is the argument of the application.

More generally if $\mathtt{n}$ is a context, variable or lambda node at depth n in $\mathcal{G}(M)$ then the set of nodes below $\mathtt{n}$ at depth greater than n must be of the form $\mathcal{G}_n(N)$ for some subterm N of M. We shall say that $\mathtt{n}$ is the root of N in M. If N is not equal to M than there is an edge on top of $\mathtt{n}$ which will be called the root edge of N in M or the root edge of $\mathtt{n}$.

A path is a sequence of consecutive nodes or equivalently a sequence of oriented edges and reversed oriented edges w.r.t. the natural orientation. We shall use both conventions and even mix them without mention. The reverse of a path φ is denoted φ^r. A path is *straight* if it contains no subpath of the form $\varphi\varphi^r$ nor uv^r where u and v are the two premises of some binary node. From now on and unless specified we assume that all the paths are straight.

To each variable node (occurrence of variable) $\mathtt{v}$ we associate its *discriminant* $\gamma_{\mathtt{v}}$: it is the maximal path starting at $\mathtt{v}$ and moving downwards. Note that it is uniquely determined by $\mathtt{v}$ and that it may contain only control nodes. The last node of $\gamma_{\mathtt{v}}$ will be called its *discriminant node*. The discriminant node $\mathtt{m}$ of $\gamma_{\mathtt{v}}$ may or not have an exiting edge. In the former case this edge must be the variable edge of some λ-node λ and we say that $\mathtt{v}$ and $\gamma_{\mathtt{v}}$ are bound by λ; if $\mathtt{m}$ has no exiting edge than it must be a free node of $\mathcal{G}(M)$.

Let γ be a discriminant; then among all the nodes visited by γ, its variable node has a maximal depth and its discriminant node $\mathtt{m}$ has a minimal depth. The difference between the two will be called the *lift* of γ. Note that if γ is bound by some λ-node λ then the depth of λ is the same than the depth of $\mathtt{m}$.

We will say that a path is of type t_1—t_2 if it starts from a node of type t_1 and ends in a node of type t_2. In particular an edge of type @-λ will be called a redex edge. There is a one to one correspondence between redexes in M and redex edges in $\mathcal{G}(M)$. We say that a path φ crosses a redex if it contains the corresponding redex edge.

If ψ is a path in $\mathcal{G}(M)$ then M has a minimal subterm N such that ψ is a path in $\mathcal{G}(N)$. We define the leftmost outermost redex crossed by ψ to be the leftmost outermost redex of N.

Let M be a term and M' be obtained by some beta-contraction of M. We define a notion of residual of the edges of $\mathcal{G}(M)$ in $\mathcal{G}(M')$ in the obvious way and extend it to the notion of residual of a path along some beta-contraction of M. Note that some paths of $\mathcal{G}(M)$ don't have any residual in $\mathcal{G}(M')$. For a precise definition see [14].

3 Legal paths

In order to express the legality condition, we begin with a definition on paths. *Well balanced paths* (shortly wbp) are inductively defined by the following clauses:

- the function edge of any application node is a wbp.
- let φ_1 be a wbp ending at a context node $\mathtt{c}$ of an application @, u be the context edge of @, φ_2 be a wbp connecting @ to a lambda node λ and u' be the body edge of λ. Then

  $$\varphi_1\, u\, \varphi_2\, u'^r$$

  is a wbp.

- let φ_1 be a wbp terminating at some variable node $\mathtt{v}$, γ be the discriminant starting at $\mathtt{v}$, u be the variable edge starting from the discriminant node of γ and ending in some λ-node λ, φ_2 be a wbp connecting an application node @ to λ and u' be the argument edge of @. Then

$$\varphi_1 \, \gamma \, u \varphi_2^r \, u'^r$$

is a wbp.

Remark.

The following lemma expresses some properties of wbp's that shall be useful later on.

Lemma 3.1

Straight. *If φ is a wbp then φ is straight and has the form $@u\varphi' v^{(r)}\mathtt{m}$ where @ is some application node, u is the function edge of @, $\mathtt{m}$ is some lambda, variable or context node and v is the root edge of $\mathtt{m}$.*

Prefix. *Let φ_1 and φ_2 be two wbp's connecting the same application node @ to respectively the lambda nodes λ_1 and λ_2. If φ_2 is a prefix of φ_1 then $\lambda_2 = \lambda_1$ and $\varphi_2 = \varphi_1$.*

Sub-wbp. *Let φ be a wbp of type @-λ and @ (resp. λ) be an application node (resp. a λ-node) visited by φ. Then φ has a unique subpath ϕ which is a wbp of type @-λ and starts at @ (resp. ends at λ).*

These are more or less immediate consequences of the definition of wbp's.

Actually the "interesting" wbp's are those of type @-λ; they intend to be isomorphical to redexes family. However it is not enough to ask only for well balancing on paths of type @-λ to get this result. The stronger condition of legality to come hereafter is needed. But first let us define by crossed induction two other kinds of paths in a term M: @-cycles and v-cycles.

(@-cycle) Let @ be an application node in $\mathcal{G}(M)$, u its argument edge and denote by N the subterm argument of @. An *@-cycle at @* is a path of the form

$$@ \, u^r \, \psi_1 \, \xi_1 \, \psi_2 \, \cdots \, \psi_n \, \xi_n \, \psi_{n+1} \, u \, @$$

where the ψ_i's are entirely contained in N and the ξ_i's are v-cycles over some free occurrences of variables of N in M. A particular case of @-cycle, which is the very base case of the crossed induction, is $n = 0$; then the @-cycle is just $@u^r \psi u @$ where ψ is a cycle contained in N starting and ending at the root node of N (the agument node of @). Note that a path entirely contained in N is at depth strictly greater than the depth of @.

(v-cycles) Let $\mathtt{v}$ be a variable node corresponding to a bound occurrence of variable in M, γ be its discriminant and u be the variable edge of its binder λ. A *v-cycle over* $\mathtt{v}$ is a path of the form

$$\mathtt{v} \, \gamma \, u \, \lambda \, \varphi^r \, @ \, \psi \, @ \, \varphi \, \lambda \, u^r \, \gamma^r \, \mathtt{v}$$

where @ is some application node, φ is a wbp linking @ to λ, and ψ is an @-cycle at @.

We now state a proposition relating wbp and @-cycles (proved in [2]):

Proposition 3.2 *Let φ be a wbp and $@ \, \psi \, @$ be an @-cycle at @ contained in φ. Then φ can be uniquely decomposed into:*

$$\zeta_1 \, \gamma_1 \, u_1 \, \lambda_1 \, {\varphi_1}^r \, @ \, \psi \, @ \, \varphi_2 \, \lambda_2 \, {u_2}^r \, {\gamma_2}^r \zeta_2$$

where both φ_i's are wbp's linking @ to some lambda nodes λ_1 and λ_2, u_i is the variable edge of λ_i and γ_1, γ_2 are the discriminants of some occurrences of variables bound respectively by λ_1 and λ_2.

In the situation of Proposition 3.2, we will say that φ_1 and φ_2 are the *call* and *return* paths of the @-cycle ψ and that γ_1 and γ_2 are the discriminants of respectively the call and return paths.

Definition 3.3 *A wbp φ is a* legal path *iff for any @-cycle ψ contained in φ, the call and return paths of ψ and their discriminants are pairwise equal.*

Legal paths are related to Lévy's families [12] in a very strong way as expressed by the following theorem (in [2]).

Theorem 3.4 *Given a λ-term M, there exists a bijective correspondence between legal paths of type @-λ in M and all the possible redex families obtained by reduction of M.*

4 LS and Regular paths

We give a presentation of the dynamic algebra LS as an equational theory. Terms of LS will be called *monomials*. We define at the same time the language and the (equational) axioms of LS. Items are:

- a composition function which is associative;

- a neutral 1 and an absorbing 0 for composition;
- an involution U^* satisfying $0^* = 0$, $1^* = 1$ and $(UV)^* = V^*U^*$ for any U and V in LS;
- a morphism ! for composition, 0, 1 and $*$;
- two *multiplicative* coefficients p and q satisfying the *annihilation axioms*:

$$\begin{aligned} p^*p = q^*q &= 1, \\ p^*q = q^*p &= 0. \end{aligned}$$

- Four *exponential* constants r, s, t, d satisfying the annihilation axioms:

$$\begin{aligned} r^*r = s^*s = d^*d = t^*t &= 1, \\ s^*r &= 0 \end{aligned}$$

and the *commutation* equations:

$$\begin{aligned} !(U)r = r!(U) \qquad & !(U)s = s!(U), \\ !(U)t &= t!^2(U), \\ !(U)d &= dU \end{aligned}$$

where U is any monomial.

The weight of the oriented edges is already defined in section 2. The weight $w(\varphi)$ of a path φ is inductively given by: if φ is a null path (a path starting and ending in the same node, crossing no edge) then its weight is 1, if φ is $u\varphi'$ (resp. $u^r\varphi'$) where u is an oriented edge then $w(\varphi) = w(\varphi')w(u)$ (resp. $w(\varphi')w(u)^*$). Note that weights are composed antimorphically w.r.t. paths and that we have $w(\varphi^r) = w(\varphi)^*$.

Definition 4.1 *A path φ is* regular *iff* $\mathsf{LS} \not\vdash w(\varphi) = 0$.

At first sight this definition looks pretty ineffective since it is usually hard to show that something is not provable in an axiomatic theory. However the the theorem AB^* below (proved in [14]) shows that it is actually easy to compute whether a weight of a path is (provably) null or not in LS.

Important convention. From now on, when U and V are monomials and unless otherwise specified we shall write $U = V$ for $\mathsf{LS} \vdash U = V$.

We say that a monomial in which the symbol ! (resp. $*$) doesn't occur is *flat* (resp. *positive*). Positive monomials are interesting in that they satisfy $A^*A = 1$ as can be easily checked by induction on the number of constants occurring in A. We define inductively the *stable forms* of LS to be the monomials of the form AB^* or $A!(M)B^*$ when A and B are positive and flat and M is in turn a stable form. Note that if M is a stable form of LS, then $M = AB^*$ for some positive (but not necessarily flat) A and B.

Proposition 4.2 (Confluence of LSO) *The rewriting system LSO on monomials defined by orienting all the equations from left to right is noetherian. Stable form are normal w.r.t. LSO.*

Let U be monomial:

- *if $U =$ for some stable form V of LS then U rewrites to V w.r.t. LSO;*
- *if $U = 0$ in then U rewrites to 0 w.r.t. LSO.*

Remark. This entails that LSO is confluent on monomials which are either null or equal to some stable form.

Theorem 4.3 (AB^*) *Let M be a term and φ be a path in $\mathcal{G}(M)$ possibly not straight. Then $w(\varphi)$ rewrites w.r.t. LSO into either 0, or a stable form of LS.*

Suppose $AB^* = 0$ for some positive A and B. Since $A^*A = B^*B = 1$ we immediately get $0 = 1$ in LS. But this is contradicted by the fact that LS has some non trivial models (see the next section). Thus rewriting a weight into AB^* form indeed shows that it is provably not null in LS.

The foregoing argument has an interesting corollary concerning models of LS:

Corollary 4.4 (Semantical AB^*.) *Let $\mathcal{M}$ be a non trivial model of LS and M be a term. Then, for every path φ (possibly not straight) in $\mathcal{G}(M)$ we have*

$$\mathsf{LS} \vdash w(\varphi) = 0 \quad \textit{iff} \quad \mathcal{M} \models w(\varphi) = 0.$$

Remark. This property is very specific to weights of paths: generally a model satisfies more equations than the theory it is a model of. For instance one may find some non trivial models of LS in which $t^*d = 0$ which is not provable in LS (there are some other models in which $t^*d \neq 0$). The theorem AB^* and its corollary are only valid for weights of paths. However it is a strong result since it states that any non trivial model of LS is as good as the theory for computing weights of paths.

5 A model of LS

In order to help with some of the weight computations to come, we give a first interpretation of the equational theory LS (the second one is the context semantic in the last section). The verification that the interpretation satisfies the equations of LS is left to the reader.

Put $\mathcal{T}$ the set of partial transformations of $\mathbb{N}$; that is, an element f of $\mathcal{T}$ is a one-to-one mapping of a subset of $\mathbb{N}$ (the *domain* of f) onto a subset of $\mathbb{N}$ (the *codomain* or *range* of f). Composition of partial transformations is defined in the obvious way (the domain of fg is the set of n such that n is in the domain of g and $g(n)$ is in the domain of f). We interpret each monomial U of LS by a partial transformation $|U|_{\mathcal{T}}$ of $\mathcal{T}$. The interpretation is defined by induction on the number of symbols appearing in U.

We denote by $0_{\mathcal{T}}$ the nowhere defined transformation, by $1_{\mathcal{T}}$ the identity on $\mathbb{N}$ and set $|0|_{\mathcal{T}} = 0_{\mathcal{T}}$ and $|1|_{\mathcal{T}} = 1_{\mathcal{T}}$. If f is a partial transformation then f^* is the inverse transformation from the codomain to the domain of f so that for any monomial U we define $|U^*|_{\mathcal{T}} = |U|_{\mathcal{T}}{}^*$.

Let $|p|_{\mathcal{T}}$ and $|q|_{\mathcal{T}}$ be two transformations of $\mathcal{T}$ with full domain and disjoint codomains; for instance take $|p|_{\mathcal{T}}(n) = 2n$ and $|q|_{\mathcal{T}}(n) = 2n+1$.

Let $(m,n) \mapsto \lceil m,n \rceil$ be a one-to-one mapping of $\mathbb{N}^2$ onto $\mathbb{N}$. We shall denote $\lceil n_1, \ldots, n_k \rceil$ the integer $\lceil n_1, \lceil n_2, \ldots, \lceil n_{k-1}, n_k \rceil \cdots \rceil \rceil$. For any transformation f we define $!_{\mathcal{T}}(f)$ by:

$$!_{\mathcal{T}}(f)\lceil m,n \rceil = \lceil m, f(n) \rceil$$

whenever n is in the domain of f. For any monomial U we set $|!(U)|_{\mathcal{T}} = !_{\mathcal{T}}(|U|_{\mathcal{T}})$. Note that if f has a full domain, since any integer may be written $\lceil m,n \rceil$ for some m and n, then $!_{\mathcal{T}}(f)$ has full domain. In particular $!_{\mathcal{T}}(1_{\mathcal{T}}) = 1_{\mathcal{T}}$. Also $!_{\mathcal{T}}(0_{\mathcal{T}}) = 0_{\mathcal{T}}$.

Let k be an integer, ρ, σ and τ be three transformations with full domains. Assume that the codomains of ρ and σ are disjoint. We define $|d|_{\mathcal{T}}$, $|r|_{\mathcal{T}}$, $|s|_{\mathcal{T}}$ and $|t|_{\mathcal{T}}$ by:

$$\begin{aligned} |d|_{\mathcal{T}}(n) &= \lceil k,n \rceil, \\ |r|_{\mathcal{T}}\lceil m,n \rceil &= \lceil \rho(m), n \rceil, \\ |s|_{\mathcal{T}}\lceil n,m \rceil &= \lceil \sigma(n), m \rceil, \\ |t|_{\mathcal{T}}\lceil l, \lceil m,n \rceil \rceil &= \lceil \tau\lceil l,m \rceil, n \rceil \end{aligned}$$

This interpretation is subjected to a lot of arbitrary choices, beginning with the choice of the one-to-one mapping from $\mathbb{N}^2$ onto $\mathbb{N}$. The effect of this is that we can easily build some interpretation satisfying some additional equations.

As a matter of fact, $\mathcal{T}$ satisfies the *inverse semigroup (isg)* [4] equations:

$$\begin{aligned} (f^*)^* &= f \\ ff^*f &= f \\ ff^*gg^* &= gg^*ff^* \end{aligned}$$

Actually, $\mathcal{T}$ is a universal isg in the sense that any countable isg is isomorphical with a sub-isg of $\mathcal{T}$. Isg's enjoy a lot of properties some of which we shall use later, namely: for any f in an isg, ff^* is an idempotent; if π and π' are two idempotents of an isg, then $\pi\pi' = \pi'\pi$; if π is an idempotent and f an element of an isg, then $\pi f = ff^*\pi f$ and $f^*\pi f$ is in turn an idempotent; if π is an idempotent and f, g are elements of an isg such that $fg = 0$ then $f\pi g = 0$. All these properties are immediate in $\mathcal{T}$ once remarked that idempotents are just identities on some subset of $\mathbb{N}$; they may be more tricky to prove in general.

The following property is a kind of converse of the semantical version of AB^* (corollary 4.4).

Proposition 5.1 *A monomial U rewrites w.r.t. LSO into an AB^* form iff it is non null in each non trivial model of LS.*

Proof. (sketch) The only if part is given by the semantical AB^* corollary 4.4. For the if part, let U_0 be some normal form of U w.r.t. LSO (which exists since, by the confluence proposition 4.2, LSO is noetherian) and suppose that U_0 is not an AB^* form. Then there must be a configuration $x^*!^k(y)$ or $!^k(x^*)y$ in U_0 where x and y are two coefficients of LS, k is a positive or null integer and such that no oriented equation can be applied. By symmetry we may consider only the first case. With a bit of work, one can find some interpretation of the constants of LS in $\mathcal{T}$ such that $|x^*!^k(y)|_{\mathcal{T}}$ is the null transformation. Since $x^*!^k(y)$ is occurring in U this entails that $\mathcal{T} \models U = 0$. ∎

6 More about regular paths

If γ is a discriminant of lift c in $\mathcal{G}(M)$, then its weight is $!^k(G)$ for a monomial G of the form:

$$G = \omega_1 t_{i_1}!(\omega_2 t_{i_2}!(\ldots\omega_c t_{i_c}!(\omega_{c+1}d)))$$

where the ω_i's are products of r's and s's only. Such a G will be called a *w-discriminant* of lift c. A *w*-

discriminant G of lift c satisfies the *generalized commutation equation*:

$$!(U)G = G!^c(U)$$

for any monomial U. If γ and γ' are two discriminants bound by the same lambda node λ, then their weights are respectively $!^k(G)$ and $!^k(G')$ where k is the depth of λ and G and G' are some w-discriminants. Furthermore G and G' satisfy the *generalized annihilation equation*:

$$\begin{aligned} G^*G' &= 1 \qquad \text{if } \gamma = \gamma' \\ &= 0 \qquad \text{otherwise.} \end{aligned}$$

Proposition 6.1 (Rendez-vous property) *Let φ be a path linking two nodes* `m` *and* `n` *and let m and n be the respective depths of* `m` *and* `n`. *If φ is a wbp then its weight satisfies the* rendez-vous equation*:*

$$!^n(X)w(\varphi) = w(\varphi)!^m(X).$$

PROOF. The proof is by induction on the definition of wbp's.

(φ is an edge). φ is the function edge of some application node so $U = 1$ and $n = m$. There is not much more to say.

($\varphi = \varphi_1\, u\, \varphi_2\, u'^r$). By definition φ_1 is a wbp ending at the context node `c` of an application node @ whose context edge is u and φ_2 is a wbp starting at @ and ending in some λ-node λ whose body edge u' ends in the node `n`. Let U_1 and U_2 be their respective weights and k be the depth of @. Notice that `c` being the context node of @ is also at depth k and that λ has the same depth n than `n`. Let $U = w(\varphi)$; with these notations we have:

$$U = !^n(q^*)U_2!^k(q)U_1$$

By induction hypothesis on φ_2 (which begins at depth k) we get $!^n(q^*)U_2!^k(q) = !^n(q^*)!^n(q)U_2$, which is equal to U_2 by the annihilation equations of q so that finally $U = U_2U_1$ Now again by induction on φ_1 and φ_2 we have:

$$\begin{aligned} !^n(X)U &= !^n(X)U_2U_1 \\ &= U_2!^k(X)U_1 \\ &= U_2U_1!^m(X) \\ &= U!^m(X) \end{aligned}$$

which shows that φ in turn satisfies the rendez-vous equation.

($\varphi = \varphi_1\, \gamma\, u\, (\varphi_2)^r\, u'^r$). Both φ_1 and φ_2 are wbp's, φ_1 ends in some variable node `v`, γ is the discriminant of `v` bound by some λ-node λ, u is the variable edge of λ, φ_2 starts in some application node @ and ends in λ and u' is the argument edge of @. We denote by k the depth of λ and by c the lift of γ. By definition `v` is therefore at depth $k + c$. Since φ ends in the argument node `n` of @ which is at depth n it must be that n is strictly positive and that the depth of @ is $n - 1$. Let U_i be the weight of φ_i and G be the w-discriminant such that $w(\gamma) = !^k(G)$. With these notation we have

$$U = !^{n-1}(p^*)U_2^*!^k(p)!^k(G)U_1$$

Now since φ_1 starts at depth m and ends at depth $k + c$ and φ_2 starts at depth k and ends at depth $n - 1$, the induction hypothesis tells us that $!^{k+c}(X)U_1 = U_1!^m(X)$ and $!^k(X)U_2 = U_2!^{n-1}(X)$. Thus we have $!^{n-1}(p^*)U_2^*!^k(p) = !^{n-1}(p^*)!^{n-1}(p)U_2^*$ so that by the annihilation equation of p this is equal to U_2^* and therefore $U = U_2^*!^k(G)U_1$. Again the rendez-vous equation applied to U_2 gives $U = !^{n-1}(G)U_2^*U_1$. Finally we may write

$$\begin{aligned} !^n(X)U &= !^n(X)!^{n-1}(G)U_2^*U_1 \\ &= !^{n-1}(!(X)G)U_2^*U_1 \\ &= !^{n-1}(G!^c(X))U_2^*U_1 \\ &= !^{n-1}(G)!^{n-1+c}(X)U_2^*U_1 \\ &= !^{n-1}(G)U_2^*!^{k+c}(X)U_1 \\ &= !^{n-1}(G)U_2^*U_1!^m(X) \end{aligned}$$

so that φ indeed satisfies the rendez-vous property. ∎

REMARK. The proof uses the equations of LS in an unoriented fashion. In other words the rendez-vous equation cannot be obtained only by using the rewriting system LSO.

Lemma 6.2 (Subweight) *Let ϕ be a regular path such that $w(\phi) = U\pi V$ for some monomials U, π and V. There are some positive monomial A and B such that π rewrites into AB^* w.r.t. LSO.*

If π is idempotent in LS then $A = B$ so that $\pi = AA^*$. If $\pi = \pi_1\pi_2$ where π_1 and π_2 are two idempotents of LS, then π is an idempotent of LS.

PROOF. Since ϕ is regular, $w(\phi)$ and thus π are not null in any model of LS. Hence by the proposition 5.1, π has an AB^* form.

Now suppose that $\pi^2 = \pi$, thus we have $AB^*AB^* = AB^*$. Since A and B are positive, we have $A^*A = B^*B = 1$ so that $B^*A = 1$. Now it is fairly easy to show by induction on the length of A and B that this is possible only if $A = B$ so that we finally get $\pi = AA^*$.

If $\pi = \pi_1\pi_2$ with π_1 and π_2 idempotent, then we have just shown that there are some positive A_1 and A_2 in LS such that $\pi_i = A_iA_i^*$. Therefore $A_1^*A_2$ is a subweight of $w(\phi)$ thus by applying the first part of the lemma we get two positive A_i' so that $A_1^*A_2 = A_2'A_1'^*$. Hence $\pi = A_1A_2'A_1'^*A_2^*$ and we have:

$$\begin{array}{rcl}\pi^2 & = & A_1A_2'A_1'^*A_2^*A_1A_2'A_1'^*A_2^* \\ & = & A_1A_2'A_1'^*A_1'A_2'^*A_2'A_1'^*A_2^* \\ & = & A_1A_2'A_1'^*A_2^* \\ & = & \pi\end{array}$$

showing that π is an idempotent. ■

7 Regular well balanced paths are legal

The *legality* of a well balanced path is a necessary condition for its regularity, as shown by Theorem 7.2. This statement relies on the following proposition:

Proposition 7.1 (@-cycle property) *Let @ be an application node in $\mathcal{G}(M)$ at depth k, u be its application edge and $u^r\psi u$ be an @-cycle at @. There is two monomials U and π such that:*

$$w(\psi) = !^{k+1}(U)\pi,$$

*and π is a k-*commuting *idempotent, i.e. π satisfies:*

$$\begin{array}{rcl}\pi^2 & = & \pi, \\ !^k(X)\pi & = & \pi !^k(X),\end{array}$$

for any monomial X;

Remark. This proposition expresses that the weight of any @-cycle is essentially similar to the weight of an elementary @-cycle. The difference only lies in some idempotent which is interpreted in the model $\mathcal{T}$ by a partial identity; furthermore the k-commutation property says that π is in some sense "invisible", at least it doesn't interact with U.

Proof. If the weight of ψ is null then the proposition is true by taking $\pi = 0$ and any monomial. So we may suppose that $w(\psi)$ is non null in LS. Let N be the subterm argument of the application @. By definition of @-cycles, ψ has the form:

$$\psi_0\, \mathbf{v}_1\, \gamma_1\, v_1\, \lambda_1\, \varphi_1^r\, @_1\, u_1^r\, \phi_1\, u_1\, @_1\, \varphi_1\, \lambda_1\, v_1^r\, \gamma_1^r\, \mathbf{v}_1\, \psi_1 \ldots \psi_n$$

where for each i, ψ_i is a path entirely contained in N, γ_i is the discriminant of some free occurrence of variable in N corresponding to the variable node $\mathbf{v}_i$, λ_i is the lambda node binding $\mathbf{v}_i$ and v_i is its variable edge, φ_i is a wbp linking an application node $@_i$ to the lambda node λ_i, u_i is the argument edge of $@_i$ and $u_i^r\, \phi_i\, u_i$ is an @-cycle at the application node $@_i$.

We shall prove by induction on n then on ψ that any path with the shape of ψ (defined in the foregoing paragraph) has the @-cycle property. Note that this is a bit more general than the statement of the proposition since for the sake of induction loading, we don't suppose that the starting node of ψ_0 and the ending node of ψ_n is the argument node of @.

The base case is $n = 0$. Then $\psi = \psi_0$ is entirely contained in N. But N being the argument of @ which is at depth k, the weight of ψ must be of the form:

$$w(\psi) = w(\psi_0) = !^{k+1}(W_0)$$

for some monomial W_0. Thus the proposition is proved with $\pi = 1$.

If $n > 0$ let l_1 be the depth of λ_1 so that the weight of γ_1 is of the form $!^{l_1}(G_1)$ for some w-discriminant G_1 of lift c_1. Furthermore let U_1 be the weight of φ_1 and k_1 the depth of $@_1$. Since $u_1^r\, \phi_1\, u_1$ is an @-cycle, ϕ_1 has the good shape so we may suppose by induction on ϕ_1 that its weight is of the form

$$w(\phi_1) = !^{k_1+1}(V_1)\pi_1$$

for some monomial V_1 and some k_1-commuting idempotent π_1. Furthermore the suffix $\psi_1 \ldots \psi_n$ of ψ again has the right shape so that its weight is by induction on n

$$w(\psi_2 \ldots \psi_n) = !^{k+1}(V)\pi$$

for some V and some k-commuting idempotent π of LS. Thus the weight of ψ is equal to:

$$\begin{array}{l} w(\psi) = !^{k+1}(V)\pi \\ \quad !^{l_1}(G_1^*)!^{l_1}(p^*)U_1\, !^{k_1}(p)!^{k_1+1}(V_1)\pi_1!^{k_1}(p^*)U_1^*!^{l_1}(p)\, !^{l_1}(G_1) \\ \quad !^{k+1}(W_0) \end{array}$$

Now since φ_1 is a wbp linking two nodes whose respective depths are k_1 and l_1 the rendez-vous property of φ_1 states that for any monomial X,

$$!^{l_1}(X)U_1 = U_1!^{k_1}(X).$$

Thus we have $!^{l_1}(p^*)U_1!^{k_1}(p) = !^{l_1}(p^*)!^{l_1}(p)U_1$ so that by the morphism equations of ! and the annihilation equation of p this is equal to U_1. The weight of ψ is therefore equal to:

$$!^{k+1}(V)\pi!^{l_1}(G_1^*)U_1!^{k_1+1}(V_1)\pi_1U_1^*!^{l_1}(G_1)!^{k+1}(W_0)$$

If we let X be $!^{l_1}(G_1^*)U_1!^{k_1+1}(V_1)\pi_1U_1^*!^{l_1}(G_1)$ then we have:

$$\begin{aligned}
X &= U_1!^{k_1}(G_1^*)!^{k_1+1}(V_1)\pi_1U_1^*!^{l_1}(G_1)\\
&= U_1!^{k_1}(G_1^*!(V_1))\pi_1U_1^*!^{l_1}(G_1)\\
&= U_1!^{k_1}(!^{c_1}(V_1)G_1^*)\pi_1U_1^*!^{l_1}(G_1)\\
&= U_1!^{k_1+c_1}(V_1)!^{k_1}(G_1^*)\pi_1U_1^*!^{l_1}(G_1)\\
&= U_1!^{k_1+c_1}(V_1)\pi_1!^{k_1}(G_1^*)U_1^*!^{l_1}(G_1)\\
&= U_1!^{k_1+c_1}(V_1)\pi_1U_1^*!^{l_1}(G_1^*)!^{l_1}(G_1)\\
&= U_1!^{k_1+c_1}(V_1)\pi_1U_1^*\\
&= !^{l_1+c_1}(V_1)U_1\pi_1U_1^*
\end{aligned}$$

by using the rendez-vous equation of U_1, the generalized commutation equation of G_1, the k_1-commutativity of π_1, the rendez-vous equation of U_1, the generalized annihilation equation of G_1 and finally the rendez-vous equation of U_1. Thus we get:

$$w(\psi) = !^{k+1}(V)\pi!^{l_1+c_1}(V_1)U_1\pi_1U_1^*!^{k+1}(W_0).$$

Now note that l_1 being the depth of λ_1, it is also the depth of the discriminant node of γ_1 thus we have $l_1 \le k$. In other words there is a positive or null d_1 such that $k = l_1 + d_1$. Let X be any monomial. By using (again) the rendez-vous property of φ_1 and the k_1-commuting property of π_1 we have that:

$$\begin{aligned}
U_1\pi_1U_1^*!^k(X) &= U_1\pi_1U_1^*!^{l_1+d_1}(X)\\
&= U_1\pi_1!^{k_1+d_1}(X)U_1^*\\
&= U_1!^{k_1+d_1}(X)\pi_1U_1^*\\
&= !^{l_1+d_1}(X)U_1\pi_1U_1^*\\
&= !^k(X)U_1\pi_1U_1^*
\end{aligned}$$

so that $U_1\pi_1U_1^*$ is k-commuting. We have to show that it is an idempotent. But by the subweight lemma (6.2), π_1 being an idempotent there is a positive monomial A_1 such that $\pi_1 = A_1A_1^*$. Furthermore the same lemma states that there are some positive B and C such that $U_1A_1 = BC^*$. Thus we have

$$\begin{aligned}
U_1\pi_1U_1^* &= BC^*CB^*\\
&= BB^*
\end{aligned}$$

which by positivity of B is clearly idempotent.

So we are in position to write

$$w(\psi) = !^{k+1}(V)\pi!^{l_1+c_1}(V_1)!^{k+1}(W_0)\pi_1'$$

where $\pi_1' = U_1\pi_1U_1^*$ is a k-commuting idempotent of LS. But c_1 is the lift of the discriminant γ_1 which starts at depth strictly greater than k and ends at depth l_1. Thus $l_1 + c_1$ is strictly greater than k; in other words there is a positive or null d_1' such that $l_1 + c_1 = k + 1 + d_1'$ so that by the k-commutation of π we may write:

$$\begin{aligned}
w(\psi) &= !^{k+1}(V)\pi!^{k+1+d_1'}(V_1)!^{k+1}(W_0)\pi_1'\\
&= !^{k+1}(V)\pi!^{k+1}(!^{d_1'}(V_1)W_0)\pi_1'\\
&= !^{k+1}(V)!^{k+1}(!^{d_1'}(V_1)W_0)\pi\pi_1'\\
&= !^{k+1}(V!^{d_1'}(V_1)W_0)\pi\pi_1'.
\end{aligned}$$

Since π and π_1' are k-commuting their product is also k-commuting and being both idempotent, by the subweight lemma their product is an idempotent. ■

Theorem 7.2 *Every regular wbp is legal.*

Proof. Let φ be a regular wbp. We have to check that any @-cycle in φ satisfies the legality condition; so let $u^r\psi u$ be an @-cycle at some application node @ (whose argument edge is u) and for $i = 1$ or 2, φ_i be a wbp linking @ to a lambda node λ_i, u_i be the variable edge of λ_i, $\mathtt{v}_i$ be a variable node bound by λ_i, γ_i be the discriminant of $\mathtt{v}_i$ and suppose that the path

$$\phi = \mathtt{v}_1\,\gamma_1\,u_1\,\lambda_1\,{\varphi_1}^r\,@\,u^r\,\psi\,u\,@\,\varphi_2\,\lambda_2\,{u_2}^r\,{\gamma_2}^r\,\mathtt{v}_2$$

is entirely contained in φ. We are to show that $\psi_1 = \mathtt{v}_1\,\gamma_1\,u_1\,\lambda_1\,{\varphi_1}^r\,@\,u^r$ and $\psi_2 = \mathtt{v}_2\,\gamma_2\,u_2\,\lambda_2\,{\varphi_2}^r\,@\,u^r$ are equal.

Let k and k_i be the respective depth of @ and λ_i; the weights of φ_i and γ_i are respectively of the form U_i and $!^{k_i}(G_i)$ where G_i is some w-discriminant of lift c_i. Then the weight of ϕ is:

$$!^{k_2}(G_2^*)!^{k_2}(p^*)U_2!^k(p)w(\psi)!^k(p^*)U_1^*!^{k_1}(p)!^{k_1}(G_1)$$

By the rendez-vous property of φ_1 and φ_2, and the annihilation equations of p this is equal to:

$$w(\phi) = !^{k_2}(G_2^*)U_2w(\psi)U_1^*!^{k_1}(G_1)$$

But the @-cycle property tells us that $w(\psi) = !^{k+1}(U)\pi$ for some U and some idempotent π. Furthermore G_i is a w-discriminant with lift c_i. So by the rendez-vous property again, the morphism equation of ! and the generalized commutation equation of

G_i we have:

$$\begin{array}{rcl} w(\phi) & = & !^{k_2}(G_2^*)U_2\,!^{k+1}(U)\,\pi U_1^*!^{k_1}(G_1) \\ & = & !^{k_2}(G_2^*)\,!^{k_2+1}(U)\,U_2\pi U_1^*!^{k_1}(G_1) \\ & = & !^{k_2}(G_2^*\,!(U))\,U_2\pi U_1^*!^{k_1}(G_1) \\ & = & !^{k_2}(!^{c_2}(U)\,G_2^*)\,U_2\pi U_1^*!^{k_1}(G_1) \\ & = & !^{k_2+c_2}(U)\,!^{k_2}(G_2^*)U_2\pi U_1^*!^{k_1}(G_1) \end{array}$$

Let $W_i = !^k(p^*)U_i^*!^{k_i}(p)!^{k_i}(G_i)$ be the weight of ψ_i. By the rendez-vous property and the annihilation equations of p we have $W_i = U_i^*!^{k_i}(G_i)$. Thus $w(\phi) = !^{k_2+c_2}(U)\,W_2^*\pi W_1$.

Now by lemma 3.1 (straight property) we easily get that ψ_1 and ψ_2 are straight paths ending in the same node. Since they are straight, we must have either $\psi_i = \phi_i v_i \phi$ where v_1 and v_2 are two distinct premises of a binary node, or one ψ_i is a suffix of the other.

In the first case let x_i be the weight of v_i, V_i and V be the respective weights of ψ_i and ψ so that $W_i = V x_i V_i$. Since the v_i's are distinct premises of a binary node we have $x_2^* x_1 = 0$ in LS, thus they are interpreted by partial transformations with disjoint codomains in $\mathcal{T}$. But

$$w(\phi) = !^{k_2+c_2}(U)V_2^* x_2^* V^* \pi V x_1 V_1.$$

$V^*\pi V$ being an idempotent of $\mathcal{T}$ we get that $x_2^* V^* \pi V x_1$ and therefore $w(\phi)$ are null in $\mathcal{T}$. By the semantical AB^* corollary 4.4 we deduce that $w(\phi)$ is null in LS which contradicts the hypothesis that ϕ is regular.

Hence we have for example that ψ_2 is a suffix of ψ_1. This entails that φ_2 is a prefix of φ_1. But both are wbp's of type @-λ thus by the lemma 3.1 (prefix property) are equal. Thus we have $\psi_i = \mathbf{v}_i\,\gamma_i\,u_1\,\lambda_1\ \ {\varphi_1}^r\,@\,u^r$; joined to the fact that ψ_2 is a suffix of ψ_1 we obtain that $\psi_2 = \psi_1$. ∎

8 Legal paths are regular

We shall prove in this section the coincidence of legal paths and (well-balanced) regular paths. For the regularity of every legal path, we need two lemmas. The former is proved in [2], the latter in [14].

Lemma 8.1 *Let M be a term, φ a legal path in $\mathcal{G}(M)$, ρ the leftmost outermost redex crossed by φ and M' the term obtained by firing ρ. Then φ has a* unique *residual φ' in M' and φ' is in turn legal.*

Lemma 8.2 (The Lifting Lemma) *Let φ be a straight path in $\mathcal{G}(M)$ whose weight is P, ρ be the leftmost outermost redex crossed by φ and u the corresponding redex edge in $\mathcal{G}(M)$, @ and λ be respectively the application and the lambda node linked by u. Suppose φ has a residual φ' by the reduction of ρ. Then we have:*

$$w(\varphi) = A(w(\varphi')B^*$$

for some positive A and B in LS.

Proof (sketch). We use some key properties of φ w.r.t. its leftmost outermost redex, namely that under the hypotheses of the lemma φ may be decomposed into:

$$\begin{array}{l} \varphi = \varphi_0\,\gamma_0\,v\,u\,{v'}^* \\ \qquad \psi_0\,v'\,u\,v^*\,{\gamma_0}^*\,\varphi_1\,\cdots\,\varphi_n\,\gamma_n\,v\,u\,{v'}^*\,\psi_n \\ \qquad v'\,u\,v^*\,{\gamma_n}^*\,\varphi_{n+1}) \end{array}$$

where v and v' are respectively the variable edge of λ and the argument edge of @, γ_i's are discriminants of occurrences of the variable bound by λ, φ_i's are subpaths of φ lying completely outside the argument of @ and ψ_i's are subpaths of φ lying completely inside the argument of @. The parenthesized subpaths may possibly be empty. From this a straightforward computation of the weight gives the result. ∎

Theorem 8.3 *Every legal path φ is regular.*

Proof. We prove by induction on the length of φ that its weight is in the AB^* form thus giving also a proof of the AB^* theorem for legal paths. If φ is the function edge of an application node then its weight is 1 so it is in AB^* form. Otherwise by definition of wbp, φ must cross some redex. Let ρ be the leftmost outermost redex of φ and φ' be the residual of φ by the contraction of ρ which uniquely exists and is legal by lemma 8.1. Then by induction hypothesis, the weight of φ' is AB^* for some positive elements A and B of LS. But by the lifting lemma, there are some positive C and D such that $w(\varphi) = Cw(\varphi')D^* = CAB^*D^*$.
Remark. Actually the proof of the AB^* theorem is essentially similar. One has just to take care of the fact that if φ is not legal then its weight may possibly be 0.

8.1 Another proof.

We shall give here the outline of another proof of the foregoing theorem. We call *cycle* a straight path of the form $\psi \equiv u\psi' u^r$ where u is the variable edge of a lambda node λ. Let $\gamma_1, \ldots \gamma_n$ be the discriminants bound by λ. We say that the cycle ψ is *deterministic* if ψ satisfies:

$$w(\gamma_i \psi \gamma_j) = 0 \iff \gamma_i \neq \gamma_j .$$

With this definition, it is possible to show the following proposition:

Proposition 8.4 *Let φ be a wbp and ψ be a subpath of φ. If ψ is a deterministic cycle at some lambda node λ, then for any discriminant γ bound by λ the path $\gamma\psi\gamma^r$ is a v-cycle.*

REMARK. This proposition mirrors the @-cycle property 7.1 of the foregoing section. The @-cycle property states that a path having such geometrical property (being an @-cycle) has such weight property; the deterministic cycles property state that a path having such weight property has such geometrical property (it is a v-cycle). Also note that a v-cycle is a path of the form $\gamma u \varphi^r \psi \varphi u^r \gamma^r$ where γ is a discriminant bound by some lambda node λ, u is the variable edge of λ, φ is a wbp linking some application node @ to λ and ψ is an @-cycle at @; now the @-cycle property of ψ and the rendez-vous-property of φ show that $u\varphi^r\psi\varphi u$ is a deterministic cycle. This allows us to see the deterministic cycle property as a kind of converse of the @-cycle property.

Let now φ be a non regular wbp. Then one may show that φ has a subpath of the form $\gamma_1\psi\gamma_2$ where γ_1 and γ_2 are two *distinct* discriminants bound by the same lambda node λ and ψ is a deterministic cycle at λ. Therefore φ cannot be legal since this would entail that $\gamma_1 = \gamma_2$.

9 The sharing graph implementation

Paths provide a basic semantics of sharing graphs [11, 9]. Let us rephrase the notions in [9] in $\mathcal{G}(M)$.

The (finite) *contexts* are terms defined by the following grammar:

$$a \ ::= \ \Box \ | \ \circ \cdot a \ | \ \star \cdot a \ | \ \sharp \cdot a \ | \ \natural \cdot a \ | \ \langle a, \, b \rangle$$

Definition 9.1 *A* consistent path *along a $\mathcal{G}(M)$ is an undirected path that can be consistently labeled by contexts. By "consistently" we mean that any pair of consecutive edges satisfies the corresponding constraint in the figure below, where $A^n[b]$ denotes a context of the form $\langle \cdots \langle b, \, a_n \rangle \, \cdots, \, a_1 \rangle$. In the cases of the context and variable nodes, the constraint is just to have the same context on top and below.*

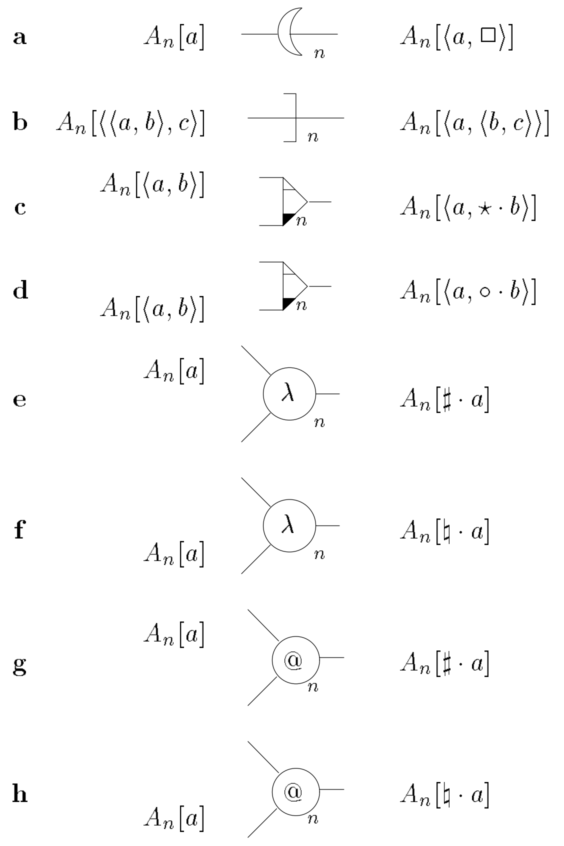


Figure 1: Context transformations

9.1 Regular paths and consistent paths

The property we are going to prove states that a path is consistent if and only if it is regular. The "only if" direction will follow by showing that the partial transformations of contexts based on the eight operations illustrated in Figure 1 are a model of dynamic algebras. The "if" direction is a consequence of corollary 4.4

Actually, in order to prove that context transformations are a model of dynamic algebras, we must consider *infinite contexts*. Let $\mathcal{C}^\infty$ be the class of functions from natural numbers to contexts; we shall denote $C \in \mathcal{C}^\infty$ as $\langle\langle \cdots C(1) \rangle, C(0) \rangle$. Definition 9.1 can be easily generalized to infinite contexts. Note that any infinite context may be written $\langle C, a \rangle$ where C is in turn an infinite context and a is a finite context.

We remark that we can correctly shift the reasonings to infinite contexts because every finite path is consistent if and only if it can be consistently labeled with infinite contexts.

Definition 9.2 (transformations of contexts) *We denote by $\mathcal{H}$ the set of* partial transformations *over $\mathcal{C}^\infty$, i.e. of one-to-one functions from some subset of $\mathcal{C}^\infty$ into $\mathcal{C}^\infty$. Composition of partial transformations is defined in the obvious way.*

Among the transformations of $\mathcal{H}$, we shall recognize the following ones:

- if $\mathtt{h} \in \mathcal{H}$ then $\mathtt{h}^*$ is the reverse transformation: its domain is the codomain of $\mathtt{h}$ and its codomain is the domain of $\mathtt{h}$;
- $\mathtt{0}, \mathtt{1} \in \mathcal{H}$, $\mathtt{0}$ is the *erroneous transformation* nowhere defined; $\mathtt{1}$ is the identity on $\mathcal{H}$;
- each constraint of figure 1 may be seen as a partial transformation: namely associate the context labelling the premise to the context labelling the non premise edge of the node. In particular the transformations associated to the context and variable nodes both are the identity and the ones corresponding to cases (e) and (h) (resp. (f) and (g)) are equal. We denote by $\mathtt{d}^n$, $\mathtt{t}^n$, $\mathtt{r}^n$, $\mathtt{s}^n$, $\mathtt{p}^n$ and $\mathtt{q}^n$ the transformations defined respectively by cases (a), (b), (c), (d), (e), (f). These are the *basic transformations* of level n;
- if $\mathtt{h}$ is a partial transformation then we denote by $!(\mathtt{h})$ the transformation defined by:

$$!(\mathtt{h})\langle C, a\rangle = \langle \mathtt{h}(C), a\rangle.$$

Theorem 9.3 *The family $\mathcal{H}$ is a non trivial model of LS.*

Proof. The constants 0, 1 are interpreted by $\mathtt{0}$ and $\mathtt{1}$; p and q by $\mathtt{p}^0$ and $\mathtt{q}^0$; for each i the exponential coefficients d, t, r and s are interpreted respectively by $\mathtt{d}^0$, $\mathtt{t}^0$, $\mathtt{r}^0$, $\mathtt{s}^0$; the involution $*$ by the inversion of partial transformation and the morphism ! (in LS) by ! (in $\mathcal{H}$). When it is not ambiguous, we shall drop the superscript 0 from the interpretations of the constants.

Note that if $\mathtt{b}^n$ is a basic transformation of level n then we have $!(\mathtt{b}^n) = \mathtt{b}^{n+1}$. Thus $\mathtt{b}^n = !^n(\mathtt{b}^0)$.

Let us check some of the axioms. The facts that the ! of $\mathcal{H}$ is a morphism and that the inversion is an antimorphism for composition are immediate. Let $C = \langle C_0, a\rangle$ be any context; then $\mathtt{p}^*\mathtt{p}(C) = \mathtt{p}^*\langle C_0, \sharp \cdot a\rangle = \langle C_0, a\rangle = C$. Furthemore $\mathtt{p}^*\mathtt{q}(C) = \mathtt{p}^*\langle C_0, \natural \cdot a\rangle$; but $\langle C_0, \natural \cdot a\rangle$ is not in the codomain of $\mathtt{p}$ thus not in the domain of $\mathtt{p}^*$, thus $\mathtt{p}^*\mathtt{q}$ is nowhere defined, i.e. equal to $\mathtt{0}$. Now C_0 may be written $\langle C_1, b\rangle$ so that $C = \langle\langle C_1, b\rangle, a\rangle$. Let $\mathtt{h}$ be any partial transformation; then $\big(!(\mathtt{h})\mathtt{t}\big)(C) = !(\mathtt{h})\langle C_1, \langle a, b\rangle\rangle = \langle \mathtt{h}(C_1), \langle a, b\rangle\rangle$ and $\big(\mathtt{t}!^2(\mathtt{h})\big)(C) = \mathtt{t}\langle\langle \mathtt{h}(C_1), b\rangle, a\rangle = \langle \mathtt{h}(C_1), \langle a, b\rangle\rangle$ so they are indeed equal; this shows that $\mathtt{t}$ satisfies the commutation axiom of t. The computations for the other axioms are similar. ■

To any path φ in $\mathcal{G}(M)$ we can associate a context transformation $\mathtt{h}_\varphi$ by (anti)composing the basic transformations of the edges crossed by φ. This construction immediately yields the following lemma:

Lemma 9.4

$$\mathcal{H} \models \mathtt{h}_\varphi = w(\varphi).$$

Theorem 9.5 *A path φ in $\mathcal{G}(M)$ is consistent iff it is regular.*

Proof. By definition, φ is consistent iff $\mathtt{h}_\varphi$ is not the $\mathtt{0}$ transformation. By the foregoing lemma this is equivalent to $w(\varphi) \neq 0$ in $\mathcal{H}$. The only if part of the theorem follows from the fact that $\mathcal{H}$ is a model of LS. The if part is a consequence of the corollary 4.4 of the AB^* theorem. ■

Remark. The model $\mathcal{H}$ is not initial in the category of dynamic algebras. For instance d^*t is not provably equal to 0 in LS. But it is $\mathtt{0}$ in $\mathcal{H}$, since the transformation $\mathtt{d}^*\mathtt{t}$ is nowhere defined. What expresses the theorem is that $\mathcal{H}$ is a good model for computing weights of paths (see the remark after corollary 4.4).

References

[1] A. Asperti. Linear Logic, Comonads, and Optimal Reductions. To appear on *Fundamenta Informaticae*, 1993.

[2] A. Asperti and C. Laneve. Paths, Computations and Labels in the λ-calculus. In *RTA '93*, volume 690 of *Lecture Notes in Computer Science*, pages 152 – 167. Springer-Verlag, 1993.

[3] N. Bourbaki. *Théorie des ensembles.* Hermann & C. Editeurs, 1954.

[4] A.H. Clifford and G.B. Preston. The Algebraic Theory of Semi-groups. In *Mathematical Surveys*, vol. 7, A.M.S., 1961.

[5] V. Danos. *La Logique Linéare appliquée à l'étude de divers processus de normalisation (principalement du λ-calcul).* PhD thesis, Université Paris VII, 1990.

[6] V. Danos and L. Regnier. Local and asynchronous beta-reduction. In *Proceedings* 8^{th} *Annual Symposium on Logic in Com puter Science,* Montreal, pages 296 – 306, 1993.

[7] V. Danos and L. Regnier. GOI Recollection. *submitted*, 1994.

[8] J. Y. Girard. Geometry of Interaction I: Interpretation of system F. In Ferro, Bonotto, Valentini, and Zanardo, editors, *Logic Colloquium '88*, pages 221 – 260. North Holland, 1988.

[9] G. Gonthier, M. Abadi, and J.J. Lévy. The geometry of optimal lambda reduction. In *Proceedings* 19^{th} *ACM Symposium on Principles of Programmining Languages*, pages 15 – 26, 1992.

[10] V. Katail. *Optimal interpreters for lambda-calculus based functional languages. Interaction nets.* PhD thesis, MIT. 1990

[11] J. Lamping. An algorithm for optimal lambda calculus reductions. In *Proceedings* 17^{th} *ACM Symposium on Principles of Programmining Languages*, pages 16 – 30, 1990.

[12] J.J. Lévy. *Réductions correctes et optimales dans le lambda calcul.* PhD thesis, Université Paris VII, 1978.

[13] M. Parigot. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In *LCS vol 624, proceedings LPAR 92*, St-Petersburg, 1992.

[14] L. Regnier. *Lambda Calcul et Réseaux.* PhD thesis, Université Paris VII, 1992.