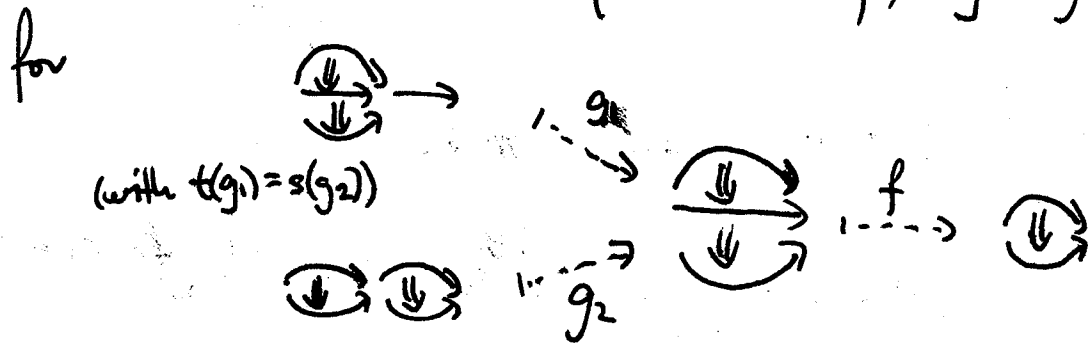


→ Recall: a globular operad is: sets $P(\pi)$ of π -ary operations^{with symbols} for each pasting diag. π , with source & target maps $s: P(\pi) \rightarrow P(s(\pi))$, $t: \dots$

draw as $f \in P(\text{diag})$ as $\text{diag} \xrightarrow{f} \text{target}$;

now have ... also, composition maps, giving eg



A right-ular to describe explicitly in these terms, so that you can, economically, a glob. set $P \rightarrow TI$ with maps $1 \rightarrow P$ $\downarrow \downarrow$ $T1 \rightarrow P$ $\downarrow \downarrow$ $T1 \rightarrow P$ $\downarrow \downarrow$ $T1 \rightarrow P$ $\downarrow \downarrow$ $T1 \rightarrow P$ $\downarrow \downarrow$ $T1 \rightarrow P$

→ Want to be able to define a weak w-category as an alg. for ~~the~~ such an operad. (All operads today: globular).

Not just any operad: two essential problems:

- desired composition laws might not exist!
- or might have multiple composition laws for a diag. (Failure of associativity.) We want all comp'n laws for a diag. to agree "up to homotopy", i.e. up to comp'n laws giving cells of the next dim'n, & these should be coherent, etc...

→ Solve both problems at once: CONTRACTIBILITY!

Def'n:
 \rightarrow A \forall glob operad contraction α

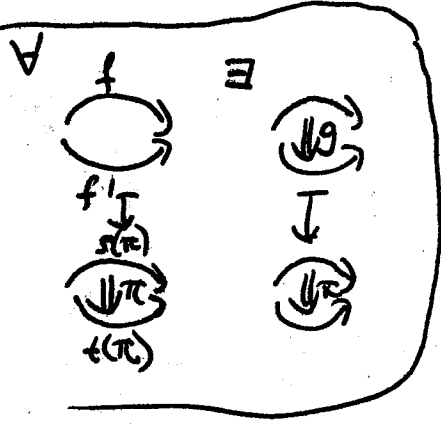
$P \xrightarrow{\alpha} \mathbb{T}\mathbb{L}$ is a map χ

giving, for ~~each~~ any $\pi \in \mathbb{T}\mathbb{L}(n)$

& any parallel pair $f, f' \in P(n-1)$

with $a(f) = s(\pi)$, $a(f') = t(\pi)$,

a lifting $g = \chi(f, f', \pi) \in P(n)$ with $f \xrightarrow{g} f'$ (i.e. $s(g) = f$, $t(g) = f'$)
 $a(g) = \pi$.



(No other axioms on χ .)

\rightarrow This ensures we do have comp'n laws: ^{eg.} get an elt $g \in P(\cdot \rightarrow \cdot \rightarrow \cdot)$ by lifting $\pi = (\cdot \rightarrow \cdot \rightarrow \cdot)$ to the pair $f = f' = e(1) \in P(\cdot)$.

\rightarrow Also ensures coherence, assoc: e.g. given $g \in P(\cdot \rightarrow \cdot \rightarrow \cdot)$, construct the two associations $f, f' \in P(\cdot \rightarrow \cdot \rightarrow \cdot)$ (using composition map)
~~write~~ ^{Get} $h: f \rightarrow f'$ in $P(\cdot \rightarrow \cdot \rightarrow \cdot)$

since $\cdot \rightarrow \cdot \rightarrow \cdot$ is a 2-cell from $\cdot \rightarrow \cdot \rightarrow \cdot$ to itself in $\mathbb{T}\mathbb{L}$.

Digress: explain degenerate p.d.s, i.e. $\rightarrow \cdot \in \mathbb{T}\mathbb{L}(1)$, also $\rightarrow \cdot \in \mathbb{T}\mathbb{L}(2)$ etc. from ids in $\mathbb{T}\mathbb{L}$.

\rightarrow Theorem: there is an initial operad-with-contraction, L .

\rightarrow Def'n (Leinster, 2004): A weak w-cat. is Initial in cat where maps must preserve the contract
 an L -algebra, i.e. $X \in \hat{\mathcal{C}}$ with operad map $L \rightarrow \text{End}(X)$.

\rightarrow Note: by def'n of L , to give wk-w-cat struc on X , it suffices to give action of some contractible operad on X .
 Hallelujah! At last! A def'n!

→ Back to type theory. (Will recall things as we need them.)

A ^(dependent) any type $(in\ context)$. Then let Δ_n ^(dept) be the context

$$x_0, y_0 : A, x_1, y_1 : Id(x_0, y_0), \dots, z_n : Id(x_{n-1}, y_{n-1}).$$

Recall: for contexts Δ, Δ' , where $\Delta' \stackrel{!}{=} B_0, x_1 : B_1(x_0), \dots, x_n : B_n(x_0, \dots, x_{n-1})$

a context map $\sigma : \Delta \rightarrow \Delta'$ is a seq. of terms $\sigma_0, \dots, \sigma_n$
 st. $\Delta \vdash \sigma_0 : B_0, \Delta \vdash \sigma_i : B_i(\sigma_0), \dots$

→ Theorem: T any extension of ML_I , Γ any closed context of T , $\Gamma \vdash A$ type.
 Then for any dept ext Δ over Γ ,
 the ^{glob. set} ~~set~~ of context maps $\Delta \xrightarrow{\sigma} \Delta_n$ carries a canonical weak cat. structure.
 "Terms of type A & its higher identities"

→ Pf: we will give an operad P_{ML} (not depending on T, Γ, A),
 show it acts on these glob. sets,
 & show it is contractible.

→ Recall: Any type theory T (with at least the structural rules of ML)
 has a syntactic category $\mathcal{C}(T)$, with objects: closed contexts of T ,
 maps: context maps as above.

Take $T = ML_I[X]$, i.e. ML_I plus a new "generic" base type X ,
 X , no term formation rules for it.

For a p.d. π , define Γ_π by: well, e.g. $\pi = \cdot \circledast \cdot$,
 $\Gamma_\pi = x, y, z: X, f, f': Id(x, y), g, g': Id(y, z), p: Id(f, f'),$
 $q: Id(g, g').$

Generally: use the fact there's a canonical ordering on the n -cells of any p.d.

There are just ~~what~~ ^{the limits} we need to define the endo operad of the ~~result~~ globular object given by the contexts \mathbb{M}_n^X in $\mathcal{C}(ML[X])$

Def'n P_{ML} is this endo. operad.

Now, for any ext'n T of ML_I , any closed type A of T ,
 have translation $ML_I[X] \rightarrow T$ taking X to A ,
 inducing functor $\mathcal{C}(ML_I[X]) \rightarrow \mathcal{C}(T)$ taking \mathbb{M}_n^X to \mathbb{M}_n^A ,
 preserving enough limits to give map of operads $End(\mathbb{M}_n^X) \rightarrow End(\mathbb{M}_n^A)$
 i.e. giving action of P_{ML} on \mathbb{M}_n^A .
 and hence on the "globular hom-sets" $\mathcal{C}(T)[\Delta, \mathbb{M}_n^A]$ as desired

(Adapt for dependent types A over Γ by using "slice type theory" T/Γ .)

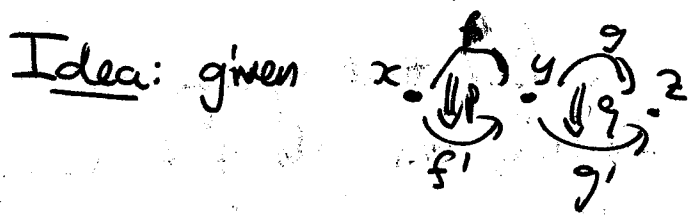
Now, the heart of the matter: contractibility of P_{ML} .

Unwind def'n of ^{endo.} ~~what~~ operad:

what is an elt $\vec{\beta}$ of $P_{ML}(\cdot \circledast \cdot)$??

→ It turns out to be terms $\sigma_0, \tau_0, \sigma_1, \tau_1, \rho$, with

$$(*) \left\{ \begin{array}{l} x: X \vdash \sigma_0(x) : X \\ z: X \vdash \tau_0(z) : X \\ x, y, z: X, f: Id(x, y), g: Id(y, z) \vdash \sigma_1(x, y, z, f, g) : Id(\sigma_0(x), \sigma_1(z)) \\ x, y, z: \dots, f': \dots, g': \dots \vdash \tau_1(x, y, z, f', g') : Id(\sigma_0(x), \sigma_1(z)) \\ x, y, z, f, f', g, g', p, q \vdash \rho(\dots) : Id(\sigma_1, \tau_1) \end{array} \right.$$



there are terms

σ_0, τ_0 for composing the 0-dim' source & tgt,
 σ_1, τ_1 for the 1-d src & tgt,
 ρ for the whole thing.

→ Contraction? Need to show: for any $\pi \in \mathbb{P}_n$ $\vec{\rho}' \in P(s(\pi))$, $\vec{\rho}'' \in P(t(\pi))$, with $\vec{\rho}', \vec{\rho}''$ parallel, need filler $\vec{\rho} \in P(\pi)$, $s(\vec{\rho}) = \vec{\rho}'$, $t(\vec{\rho}) = \vec{\rho}''$.

Now, $\vec{\rho}'$ is a seq. of terms $\sigma_0', \tau_1', \dots, \sigma_{n-2}', \tau_{n-1}', \rho'$ as in $\textcircled{*}$ for $s(\pi)$
 $\vec{\rho}''$ is $\dots \dots \dots \sigma_0'', \tau_1'', \dots, \tau_{n-2}'', \rho'' \dots$

saying these are parallel is saying $\sigma_i' = \sigma_i'', \tau_i' = \tau_i''$.

so finding setting $\sigma_i := \sigma_i', \tau_i := \tau_i' (i < n-1), \sigma_n = \rho', \tau_n = \rho''$,

a parallel pair needing a filler is exactly a seq. of terms

$\sigma_0, \tau_0, \dots, \sigma_{n-1}, \tau_{n-1}$ as in $\textcircled{*}$ for π , & a filler for this is exactly just a term ρ to complete the seq. $((\sigma_i, \tau_i), \rho)$ of $\textcircled{*}$.

eg
So: have terms $x:X \vdash \sigma_0(x):X$, $z:X \vdash \tau_0(z):X$,

$$x,y,z:X, f:Id(x,y), g:Id(y,z) \vdash \sigma_1(x,y,z,f,g):Id(\sigma_0(x), \tau_0(z))$$

$$x,y,z:X, f':Id(x,y), g':Id(y,z) \vdash \tau_1(x,y,z,f',g'):Id(\sigma_0(x), \tau_0(z))$$

& need a term

$$x,y,z:X, f,f':Id(x,y), g,g':Id(y,z) \vdash p:Id(f,f'), q:Id(g,g')$$

$$\vdash p:Id(\sigma_1(x,y,z,f,g), \tau_1(x,y,z,f',g'))$$

So, working bottom-up as usual; we find that by repeatedly using Id-elim, on cells of the highest dimension "first", it's enough to give the filler in the core where x & iterated reflexivity terms $r(x), r(r(x)), \dots$ have been plugged in for all the variables.

What can we do now? How do we know these now agree?

$$x \vdash ? : Id(\sigma_0(x,x,x,r(x),r(x)), \tau_0(x,x,x,r(x),r(x)))$$

$$x,y,f \vdash ? : Id(\sigma_1(x,y,y,f,r(y)), \tau_1(x,y,y,f,r(y))) \quad [Id-elim]$$

$$x,y,z,f,g \vdash ? : Id(\sigma_2(x,y,z,f,g), \tau_2(x,y,z,f,g)) \quad [Id-elim]$$

$$x,y,z,f,f',g,p \vdash ? : Id(\sigma_1(x,y,z,f,g), \tau_1(x,y,z,f',g)) \quad [Id-elim]$$

$$x,y,z,f,f',g,g',p,q \vdash ? : Id(\sigma_1(x,y,z,f,g), \tau_1(x,y,z,f',g'))$$

Normalisation!

This is the punchline!*

→ Normalisation ensures that any term $x:X \vdash \pi(x):X$ must be ~~the~~ of canonical form, i.e. just equal to x itself, (equal to one) so $x:X \vdash \sigma_0(x) = \tau_0(x) = x:X$,

next, it ensures that any term $x:X \vdash \pi(x):Id(x,x)$ must be equal to one of canonical form, i.e. to $r(x)$,

$$\text{so } x:X \vdash \sigma_1(x,x,x, r(x), r(x)) = \tau_1(x,x,x, r(x), r(x)) = r(x).Id(x,x).$$

In the general case, we can show, ~~that~~ by working up thru the dimensions, that after plugging in all iterated $r(-)$ terms, each σ_i, τ_i must become equal to iterated reflexivity terms $r^i(x)$;

so now we can just take the desired filler to be $\rho(x) := r^n(x) : Id(r^{n-1}(x), r^{n-1}(x))$

so we are done!

= This completes the proof of the theorem. □

* Actually, I haven't yet checked carefully that existing normalization statements do give this; if they don't, though, I have an alternative proof for this key step, ~~but~~ by building Pmc ~~from~~ the fragment with just Id types (no Π 's, Σ 's, 1's) & showing (by struct'l indu) that the context $x:X$ is initial in this fragment. However, ~~that~~ is much more cumbersome!

→ Obvious questions: are given wk-cats $\mathcal{C}(T)(\Delta, \underline{A}_\bullet)$,
for a type A & ctxt Δ in a t.t. T .

→ This is easily functorial in T & Δ , i.e. translations
 $T \rightarrow T'$ & ctxt maps $\Delta' \rightarrow \Delta$ induce strict maps of
wk w-cats in the obvious way.

→ What about terms $a: A \vdash \tau(a): B$? These should
induce "weak-w-functors", "weak maps of w-cats" ~~from~~
 $\tau^*: \mathcal{C}(T)(\Delta, \underline{A}_\bullet) \rightarrow \mathcal{C}(T)(\Delta, \underline{B}_\bullet)$

But "weak w-functor" is a slippery concept, & this doesn't
seem simple to show! This is an important question...



Further directions:

→ Can we use this to give a syntactic presentation of
wk-w-gpds? i.e. do all wk-w-gpds (up to some notion
of weak equivalence) arise in this way?

→ We can formalise much maths in extensions of ML_I !

So, what ~~are~~ are the "fund'l wk w-cats" of various natural
types, e.g. Bool, Nat, ...? Their geometric realisations
are top spaces; what do these look like? Are ~~the~~ nice descriptions
attainable at all?? ~~of them~~