Universitat de Barcelona
Facultat de Matemàtiques i Informàtica

GRAU DE MATEMÀTIQUES

Treball final de grau

TEORIA HOMOTÒPICA DE TIPUS

Autor: David Martinez Carpena

Director: Dr. Carles Casacuberta
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 19 de gener de 2020

Resum

La teoria homotòpica de tipus és una branca de les matemàtiques originada a la dècada de 2010. Les principals innovacions respecte a altres teories de tipus més clàssiques són l'associació dels tipus amb ∞-grupoides, l'axioma d'univalència de Voevodsky i els tipus inductius d'ordre superior. Els tipus inductius d'ordre superior permeten definir objectes topològics com la circumferència o el tor d'una manera sintètica. En els primers capítols d'aquest treball es presenta una introducció a la teoria homotòpica de tipus, enfocada especialment a entendre els tipus inductius d'ordre superior.

A causa del poc temps transcorregut des de l'aparició de la teoria homotòpica de tipus, encara hi ha moltes preguntes obertes per respondre. En aquest treball s'enceta una línia de recerca motivada per una d'aquestes preguntes: com donar una definició adient d'orientabilitat que sigui aplicable a superfícies o, més generalment, a varietats. A partir de la definició ja existent del tor com a tipus inductiu d'ordre superior, hem estudiat una definició anàloga de l'ampolla de Klein. Ens hem centrat en el fet que, tant en topologia com amb tipus, el tor és un recobridor de dos fulls de l'ampolla de Klein. En el treball es detallen els conceptes bàsics relacionats amb recobridors en la teoria homotòpica de tipus i es demostren resultats que són rellevants per al cas concret del tor i l'ampolla de Klein.

Abstract

Homotopy type theory is a branch of mathematics that emerged in the decade of 2010. The major novelties with respect to previous type theories are the association of types with ∞-groupoids, Voevodsky's univalence axiom, and higher-order inductive types. Higher-order inductive types allow certain objects to be defined, such as a circle or a torus, in a synthetic way. The first chapters of this work offer an introduction to homotopy type theory, focusing especially on understanding higher-order inductive types.

Due to the short time elapsed since the advent of homotopy type theory, there are many open questions waiting to be answered. This work sets out a research direction motivated by one of these questions: how to find an appropriate definition of orientability which is meaningful for surfaces or, more generally, for manifolds. From the existing definition of a torus as a higher-order inductive type, we have studied an analogous definition of a Klein bottle, focusing on the fact that a torus is a two-sheeted covering of a Klein bottle. This work contains basic facts about coverings in homotopy type theory, as well as a few results that are relevant in the special case of the torus and the Klein bottle.

2010 Mathematics Subject Classification. 18G55, 03B70

Continguts

Introducció

La teoria de tipus va néixer entre el 1902 i el 1908, creada per Bertrand Russell com a solució a la paradoxa que va trobar en la versió de Gottlob Frege de la teoria de conjunts elemental. La teoria de conjunts va continuar avançant amb noves formalitzacions i poc després es va acceptar com a fonamentació de les matemàtiques.

Durant el segle XX, la teoria de tipus va continuar desenvolupant-se. L'any 1972, Per Martin-Löf va crear la teoria de tipus intuïcionista, que, juntament amb treballs de Thierry Coquand, va donar lloc als primers assistents de demostració que utilitzaven teoria de tipus. Actualment, la majoria dels assistents de demostració utilitzen llenguatges basats en alguna classe de teoria de tipus.

Un tipus és un concepte molt similar a un conjunt, ja que representa un contenidor de certs elements. Però, a diferència de la idea d'un conjunt, un element d'un tipus no té sentit fora d'aquest i no pot ser compartit per cap altre tipus. Per tant, cada element tindrà només un tipus associat.

Una altra diferència significativa és la manera d'integrar la lògica. La teoria de conjunts està definida al damunt d'una lògica de primer ordre, que sol utilitzar una lògica clàssica com a base. En canvi, la teoria de tipus substitueix tant la teoria de conjunts com la lògica de base. La lògica es codifica dintre de la teoria de tipus, amb interpretacions com la de Brouwer–Heyting–Kolmogorov. Conceptes com una implicació lògica queden definits a partir de tipus i elements.

Per arribar a la teoria homotòpica de tipus van fer falta moltes contribucions. El 1998, Martin Hofmann i Thomas Streicher van donar una demostració del fet que la teoria de tipus intuïcionista admetia un model en la categoria de grupoides. El 2005, Steve Awodey i el seu estudiant Michael Warren van publicar un model de la teoria de tipus intuïcionista utilitzant categories de Quillen. El nom "Homotopy type theory" va sorgir a partir d'una xerrada d'Awodey el 2007. El 2009, Vladimir Voevodsky va donar els detalls d'un model de la teoria de tipus amb complexos de Kan.

Entre 2012 i 2013, Awodey, Coquand i Voevodsky van organitzar un programa de recerca amb el títol "A Special Year on Univalent Foundations of Mathematics". Aquest esdeveniment va ser un catalitzador per al desenvolupament de tot el camp. Van decidir que calia un llibre sobre la teoria homotòpica de tipus. El llibre [Uni13] va ser redactat cooperativament i de manera oberta, en una plataforma web, i publicat amb una llicència Creative Commons molt permissiva. Actualment el llibre es pot descarregar gratuïtament des del web oficial.

La teoria homotòpica de tipus és una extensió de la teoria de tipus intuïcionista, que associa a cada tipus un ∞-grupoide. A més, afegeix l'axioma d'univalència de Voevodsky i els tipus inductius d'ordre superior.

Sorprenentment, molts objectes topològics es poden construir sintèticament com a tipus inductius d'ordre superior. Amb "sintèticament" ens referim a la definició amb axiomes que capturen els aspectes clau de l'estructura a definir, en contrast amb una definició "analítica", on normalment es defineix l'estructura a partir de les seves propietats dintre d'una altra estructura de la qual partim com a base. Per exemple, en topologia clàssica es defineix la circumferència analíticament utilitzant l'equació $x^2 + y^2 = 1$, que descriu una relació entre punts del pla mitjançant les seves coordenades.

El primer objectiu d'aquest treball és familiaritzar-nos amb els tipus inductius d'ordre superior i aconseguir utilitzar-los per a finalitats relacionades amb la teoria d'homotopia. Al buscar resultats sobre tipus inductius d'ordre superior, vam trobar que recentment s'havia definit el tor i s'havia demostrat que aquest era igual al producte de dues circumferències (un fet ben conegut a topologia). La demostració d'aquesta propietat va ser publicada per primer cop per Sojakova entre 2014 i 2015, amb l'esborrany [Soj14] i l'article [Soj15]. La seva demostració és molt tècnica i bastant extensa: l'esborrany té 27 pàgines i l'article en té 22. Aquesta extensió és deguda a la complexitat inevitable d'una igualtat 2-dimensional en la definició del tor.

El 2015 es va publicar l'article [LB15; Lic15b] de Licata i Brunerie, on també es demostrava l'equivalència entre un tor i un producte de dues circumferències, i s'hi va afegir una formalització implementada mitjançant l'assistent de demostració Agda [Lic15a]. L'article va aconseguir reduir l'extensió de la demostració de Sojakova a la meitat aproximadament, utilitzant tècniques més avançades.

En aquest treball estudiarem una part dels articles de Sojakova com a base per a la formalització del tor com a tipus inductiu d'ordre superior. Per tal d'anar més enllà utilitzant aquests nous conceptes, necessitàvem buscar algun altre problema basat en tipus inductius d'ordre superior aplicats a objectes topològics i que no trobéssim resolt. La nostra referència per a la cerca de resultats va ser la llista de problemes oberts en teoria homotòpica de tipus [nLa19] a l'enciclopèdia col·laborativa nCatLab. De totes les propostes de la llista, n'hi havia una que estava relacionada amb el tor:

> Similarly to the torus, consider the projective plane, Klein bottle, etc., as discussed in the book (Sec. 6.6). Show that the Klein bottle is not orientable. (This requires defining "orientable".) (I. 1)

Aquest problema ens obria una línia de recerca on buscar nous resultats interessants. S'hi defineix un objectiu a llarg termini: la definició del concepte d'orientabilitat a teoria homotòpica de tipus. Encara que aquest objectiu excedeix les possibilitats d'un treball final de grau, ens ha obert una nova direcció per treballar-hi.

Una primera observació és que es pot definir tant el pla projectiu com l'ampolla de Klein de manera similar a la definició sintètica del tor. Tanmateix, en teoria homotòpica de tipus, després de donar una definició com la del tor, s'acostuma a comprovar que aquesta definició compleix les propietats que esperaríem. Per aquest motiu, necessitàvem algun fet que ens permetés comprovar que una definició sintètica de l'ampolla de Klein és plausible.

Aquesta situació ens va conduir cap a un resultat ben conegut que relaciona el tor amb l'ampolla de Klein: el tor recobreix l'ampolla de Klein, concretament amb un recobriment de dos fulls. Vam trobar que els recobridors estaven formalitzats a la teoria homotòpica de tipus, concretament als articles [HH18; Hou13] de Favonia.

Motivats pel cas concret del tor i l'ampolla de Klein, vam decidir estudiar la relació entre els recobridors i els n-tipus en el seu sentit topològic: un espai topològic és un n-tipus si els seus grups d'homotopia en dimensions superiors a n s'anul·len. Aquest concepte té un anàleg directe en la teoria homotòpica de tipus, que expliquem en el treball. La circumferència i el tor són 1-tipus tant a topologia com en la teoria homotòpica de tipus.

Estructura del treball

Hem estructurat la memòria en tres capítols. En el primer, presentem la teoria de tipus intuïcionista seguida per la teoria homotòpica de tipus, de manera molt similar a com ho fa el llibre [Uni13].

En el segon capítol, presentem els tipus inductius i els tipus inductius d'ordre superior. Un cop ens haguem familiaritzat amb aquests conceptes, farem una passada pels continguts dels articles [Soj15; Soj14] de Sojakova, presentant la definició del tor i demostrant la primera part de l'equivalència entre el tor i el producte de dues circumferències.

Per acabar, al tercer capítol introduirem els conceptes de recobridor i n-tipus en la teoria homotòpica de tipus, i demostrarem uns resultats que ens relacionen aquests dos conceptes. Conclourem donant un recobridor de dos fulls de l'ampolla de Klein i conjecturant una equivalència entre aquest recobridor i el tor. Si aquesta equivalència s'acabés de demostrar, implicaria que l'ampolla de Klein és un 1-tipus. En teoria d'homotopia clàssica, el recobriment de l'ampolla de Klein pel tor és un cas particular dels recobriments orientables de dos fulls de les superfícies no orientables.

Capítol 1

Introducció a la teoria homotòpica de tipus

1.1 Teoria de tipus intuïcionista

En aquesta secció presentem els fonaments de la teoria de tipus intuïcionista, que és la que va definir Per Martin-Löf i és utilitzada com a base per a definir la teoria homotòpica de tipus.

Seguirem el desenvolupament del llibre [Uni13]. Per a una formalització computacional de tota aquesta teoria, es pot consultar el primer apèndix d'aquest mateix llibre.

1.1.1 Fonaments

La teoria de tipus és un sistema deductiu i, com a tal, està format per enunciats i regles. Les regles ens permeten construir nous enunciats a partir d'enunciats construïts anteriorment. Anomenarem metallenguatge al format per les regles i els enunciats.

En formalitzar matemàtiques en teoria de tipus, definirem tipus que ens permetran conceptualitzar des d'objectes com les funcions a raonaments com la lògica. Tot anirà dintre de la teoria, utilitzant el llenguatge intern. Aquest llenguatge intern es definirà a partir de definicions de tipus utilitzant enunciats i regles.

Considerarem dos enunciats diferents. El primer és el d'introducció d'elements d'un tipus. Si A és un tipus qualsevol, podem introduir un element a d'aquest tipus amb l'enunciat $a : A$. Com ja hem esmentat, els enunciats no són proposicions i no formen part de la lògica que utilitzarem per a modelar les matemàtiques. Per tant, a diferència de la relació matemàtica de pertinença, no podrem negar aquesta introducció. Aquest tipus d'enunciats són més similars a la introducció de variables en programació que a la pertinença entre conjunts. A més, un element a introduït per l'enunciat $a : A$ no pot ser part d'un altre tipus, és a dir, si B és un tipus qualsevol diferent de A, mai no podem afirmar $a : B$. Cada element pot tenir un i només un tipus.

El segon enunciat és la igualtat per definició. Aquesta igualtat ens comunica que dos elements són iguals per definició. Per exemple, considerant un tipus A qualsevol i dos elements x i y de A, seria vàlid afirmar que $a \equiv b$. Aquest enunciat també es pot denotar com $a \equiv b : A$, o en cas que estiguem definint a a partir d'una altra expressió, $a :\equiv b$. Aquesta igualtat no serà la que utilitzarem en la majoria de casos on escriuríem una igualtat a teoria de conjunts. Un exemple on utilitzaríem aquesta igualtat és en la definició de funcions i l'avaluació d'aquestes. Més endavant veurem que tenim una igualtat en el llenguatge proposicional que serà un tipus $\cdot =_A \cdot$, que farà el paper de la igualtat tradicional que coneixem.

Pel que fa a les regles, les definirem per a cada tipus. Un exemple de regla seria si $f : A \to B$ és una funció entre dos tipus qualssevol A i B, i tenim un element $a : A$, podem avaluar-la i obtindrem $f(a) : B$. Per tant, podem veure com a partir de dos enunciats d'introducció, el de la funció i el de l'element a, hem deduït un enunciat d'introducció per a l'element $f(a)$.

Per tant, la definició d'un nou tipus serà un recull de regles que el caracteritzin. En la majoria de casos, definirem tipus parametritzats, que són un conjunt de tipus que són molt similars i només varien segons el paràmetre utilitzat. Per exemple, definirem el tipus de les funcions entre dos tipus qualssevol A i B, el qual està parametritzat per aquests dos tipus A i B.

Abans de poder escriure teoria de tipus necessitem uns tipus bàsics. Primer donarem alguns tipus que considerem part dels fonaments i que utilitzarem per definir la resta de tipus del treball. La definició d'aquests fonaments ens apropa molt a la formalització d'un llenguatge i necessitem introduir alguns conceptes de llenguatges formals.

Anomenarem *variable* a una cadena de símbols que considerarem com única unitat indivisible amb un cert significat assignat. Direm que una variable és *lligada* si ha estat assignada a un significat en particular, i que és *lliure* en el cas contrari. Mitjançant les variables, definirem una *expressió* com una concatenació de variables i símbols. Donada una expressió Φ que conté una variable lliure x, podem construir l'expressió resultant de substituir totes les aparicions de x per una altra variable y. Denotarem al resultat d'aquesta operació com $\Phi[y/x]$.

Per exemple, a l'expressió "$x + 2$", que anomenarem Φ, la variable x és lliure. Aleshores podem substituir totes les aparicions de x a l'expressió Φ per una altra variable y, obtenint l'expressió $\Phi[y/x]$ que serà "$y + 2$". En canvi, la variable x és lligada, per exemple, en l'expressió "$\forall x, x = x + 0$".

Universos

Per començar a treballar amb la teoria de tipus, necessitem una manera d'introduir els tipus com a variables. Les expressions com "Sigui A un tipus qualsevol" es formalitzen mitjançant el concepte dels universos.

Un *univers* és un tipus que conté altres tipus. Per a evitar certes paradoxes relacionades amb l'univers contenint-se a si mateix, suposarem que existeix una jerarquia d'universos tal que

$$\mathrm{U}_0 : \mathrm{U}_1 : \mathrm{U}_2 : \cdots .$$

A més, aquesta jerarquia és acumulativa; això implica que tots els tipus de U_0 també existeixen com a elements de U_1.

A la pràctica, s'acostuma a utilitzar un únic símbol U per fer referència a qualsevol dels universos de la jerarquia. Aquest fet s'anomena *ambigüitat típica*, i s'utilitza perquè no afecta els resultats obtinguts i simplifica la notació. Tota la informació de la jerarquia d'universos es pot recuperar a partir del context. Llavors, utilitzant aquesta nova notació, l'expressió "Sigui A un tipus qualsevol" és equivalent a $A : \mathrm{U}$.

Funcions

En teoria de conjunts una funció es defineix com un conjunt de parells d'elements. Aquesta definició no ens parla directament de com s'han d'utilitzar les funcions, però ens permet deduir certes propietats que caracteritzen la manera usual de treballar amb elles. En canvi, a teoria de tipus les funcions es defineixen a partir de com s'utilitzen.

Definició 1.1. Siguin A i B dos tipus qualsevol. Definim el *tipus de les funcions amb domini A i codomini B*, que denotarem $A \to B$, com el que compleix les regles següents:

- **Introducció:** Donats una variable lliure x i una expressió Φ que pot contenir x, si per a tot element a de A tenim que $\Phi[a/x] : B$, llavors l'equació
$$f(x) :\equiv \Phi$$
ens introdueix una funció $f : A \to B$.
- **Eliminació:** Si $a : A$ i $f : A \to B$, llavors $f(a) : B$. Anomenarem a aquesta regla *aplicació d'una funció f a un element a*, i direm que $f(a)$ és la seva *imatge*.
- **Computació:** Donada una funció $f : A \to B$ amb equació $f(x) :\equiv \Phi$, per a tot $a : A$ es complirà
$$f(a) \equiv \Phi[a/x] : B.$$

A part de la notació que acabem de definir, existeixen altres notacions molt utilitzades per a definir les funcions. En els casos on no vulguem donar nom a una funció, utilitzarem la notació $(x \mapsto \Phi) : A \to B$. Definirem que aquesta notació és equivalent a una funció $f : A \to B$ definida amb una equació $f(x) :\equiv \Phi$. Aquesta equivalència s'anomena *principi d'unicitat*.

Durant aquest treball, en algunes ocasions també presentarem una funció de la manera següent:
$$\begin{array}{rccc} f: & A & \to & B \\ & x & \mapsto & \Phi \end{array}$$

Per exemple, podem definir una funció $f : \mathbb{N} \to \mathbb{N}$ mitjançant l'equació $f(x) :\equiv x + x$ o amb la notació anònima $(x \mapsto x + x) : \mathbb{N} \to \mathbb{N}$.

Existeixen funcions que utilitzarem freqüentment. Per a qualsevol tipus A, podem definir la *funció identitat de A*, que denotarem $\mathtt{id}_A : A \to A$, amb l'equació $\mathtt{id}_A(a) :\equiv a$ per a tot $a : A$.

La definició que hem donat només serveix per a definir funcions d'una variable. Però a partir d'aquestes, és possible definir funcions de diverses variables mitjançant la iteració de funcions.

Per exemple, una funció de dues variables amb els tipus A, B i codomini C s'escriu

$$f : A \to (B \to C)$$

i suposant $a : A$ i $b : B$, tenim $f(a) : B \to C$ i $f(a)(b) : C$. Per conveniència, definirem

$$f(a, b) :\equiv f(a)(b).$$

En expressions amb més d'una funció iterada, sempre suposarem associativitat cap a la dreta, és a dir

$$A \to B \to C :\equiv A \to (B \to C).$$

Finalment, definim un concepte que ens caracteritzarà les funcions amb un bon comportament "recursiu". Més endavant necessitarem aquesta definició.

Definició 1.2. Direm que una funció en n variables $f : A_1 \to \cdots \to A_n \to B$ és *estrictament positiva respecte a* B si els tipus $A_1, A_2, \ldots, A_n$ o bé són funcions iterades on B només apareix com a codomini, o bé no estan formats a partir de B.

Per exemple, de les quatre funcions següents

$$\begin{array}{ll} f_1 : A \to B \to C \to D & f_4 : A \to (D \to B) \to C \to D \\ f_2 : A \to D \to C \to D & f_5 : A \to (D \to D) \to C \to D \\ f_3 : A \to (B \to D) \to C \to D & \end{array}$$

només f_1, f_2 i f_3 són estrictament positives respecte a D.

Funcions dependents

Utilitzant els conceptes d'univers i de funció, considerem el cas que una funció tingui l'univers com a codomini. Anomenarem *família indexada per* A a una funció $P : A \to \mathrm{U}$. Observem que per a cada $a : A$, $P(a)$ serà un tipus diferent de U.

A partir de les famílies, podem definir les funcions dependents. Aquestes són semblants a les funcions ordinàries, però amb la peculiaritat que el tipus del codomini també varia en funció de l'argument donat. Aquesta construcció no té cap anàleg en teoria de conjunts.

Per donar un exemple, considerem un tipus qualsevol A, una família $P : A \to \mathrm{U}$ i una funció dependent f amb domini A i codomini P. Per a dos arguments diferents $x, y : A$ tindrem les imatges $f(x) : P(x)$ i $f(y) : P(y)$, on $P(x)$ i $P(y)$ són dos tipus diferents que depenen de l'argument.

Per a una definició més formal, ho presentarem de manera similar a les funcions ordinàries.

Definició 1.3. Siguin A un tipus qualsevol i $P : A \to \mathrm{U}$ una família indexada per A. Definim el *tipus de les funcions dependents amb domini* A *i codomini* P, que denotarem $\prod_{x:A} P(x)$, com el que compleix les regles següents:

- **Introducció:** Donats una variable lliure x i una expressió Φ que pot contenir x, si per a tot element a de A tenim que $\Phi[a/x] : B(a)$, llavors l'equació

  $$f(x) :\equiv \Phi$$

  ens introdueix una funció dependent $f : \prod_{x:A} P(x)$.

- **Eliminació:** Si $a : A$ i $f : \prod_{x:A} P(x)$, llavors $f(a) : B(a)$. Anomenarem a aquesta regla *aplicació d'una funció dependent f a un element a*, i direm que $f(a)$ és la seva *imatge*.
- **Computació:** Donada una funció $f : \prod_{x:A} P(x)$ amb equació $f(x) :\equiv \Phi$, per a tot $a : A$ es complirà
$$f(a) \equiv \Phi[a/x] : B(a).$$

Sigui $P : A \to \mathrm{U}$ una família constant, és a dir, amb $P(a) :\equiv B$ per a tot $a : A$. La funció dependent definida a partir d'aquesta serà equivalent a una funció ordinària $f : A \to B$.

Podem donar un exemple de funció dependent amb la versió dependent de la funció identitat $\mathtt{id} : \prod_{A:\mathrm{U}} A \to A$ definida per

$$\mathtt{id}(A, x) :\equiv x.$$

Igual que amb les funcions ordinàries, definim funcions dependents en diverses variables mitjançant funcions dependents iterades. Però cal tenir en compte que en aquest cas els tipus dels arguments poden dependre dels anteriors. Per exemple, considerant un tipus A qualsevol, $B : A \to \mathrm{U}$ una família i $C : \prod_{x:A}(B(x) \to \mathrm{U})$ una funció dependent que retorna una família, una funció dependent f en dues variables seria

$$f : \prod_{x:A} \prod_{y:B(x)} C(x, y)$$

i també com en el cas anterior, denotarem $f(x, y) :\equiv f(x)(y)$.

Finalment, denotarem les funcions dependents amb associativitat cap a la dreta, tant entre diverses funcions dependents, com si estem barrejant funcions ordinàries amb dependents. És a dir, posant alguns exemples:

$$\prod_{x:A} \left(\prod_{y:B(x)} C(x, y) \right) \equiv \prod_{x:A} \prod_{y:B(x)} C(x, y)$$
$$\prod_{x:A} (B(x) \to C(x)) \equiv \prod_{x:A} B(x) \to C(x)$$
$$A \to \left(\prod_{x:A} B(x) \right) \equiv A \to \prod_{x:A} B(x)$$

1.1.2 Construccions bàsiques

A partir dels fonaments que hem definit, construirem els tipus bàsics per a treballar amb teoria de tipus. El mètode que utilitzarem serà la manera estàndard de definir un tipus qualsevol.

Com hem vist en els casos de les funcions ordinàries i dependents, les regles que donem per a definir un tipus tenen una estructura. En general, per a definir un tipus qualsevol X serà necessari donar les següents:

1. **Regla de formació:** Defineix com formar una instància de X. Per exemple, en el cas de les funcions, la regla seria que donats A i B tipus qualssevol, podem formar un tipus $A \to B$ de funcions de A a B. En aquest treball considerarem implícita la regla de formació, ja que queda determinada per com anomenem el tipus.

2. **Regles d'introducció:** Descriuen com introduir elements de X. En el cas de les funcions, ens defineixen la introducció de funcions a partir d'una equació vàlida.
3. **Regles d'eliminació:** Donat un element del tipus X, aquestes regles ens permeten utilitzar-lo per obtenir altres enunciats, eliminant-lo en el procés. En el cas de les funcions, l'aplicació de funcions ens permet utilitzar una funció i un element del domini per obtenir un element del codomini.
4. **Regles de computació:** Descriuen l'aplicació de les regles d'eliminació als elements introduïts per les regles d'introducció utilitzant igualtats per definició.

Per als tipus que definim d'ara en endavant, no definirem les regles d'eliminació i computació directament, sinó que proporcionarem uns principis que ens permetran crear funcions ordinàries i dependents amb el tipus que vulguem definir com a domini. Anomenarem *principi de recursió* a la regla que ens permetrà crear funcions ordinàries, i *principi d'inducció* a la seva versió per a funcions dependents. Com veurem més endavant, aquests noms encaixen amb conceptes clàssics per a certs tipus, com els nombres naturals.

Donats aquests dos principis, les regles d'eliminació i computació podran ser deduïdes a partir d'aplicar el principi de recursió (o el d'inducció) seguit de la regla d'eliminació o computació de funcions (o de funcions dependents). Per tant, en aquest treball, definirem tipus donant la regla de formació implícita, les d'introducció i els principis de recursió i d'inducció.

Productes

Donats dos tipus A i B arbitraris, el tipus $A \times B$ s'anomena *producte cartesià* entre A i B. Per a construir elements d'aquest tipus, donem un element a de A i un b de B amb els quals introduïm l'element $(a, b) : A \times B$.

Ja sabem com formar el producte i com introduir els seus elements; per tant, és suficient donar els principis de recursió i d'inducció per acabar la seva descripció. Comencem pel de recursió. Aquest principi ens ha de donar una regla per a definir funcions amb domini $A \times B$. Sigui C un tipus qualsevol. Definim $f : A \times B \to C$ utilitzant una funció $g : A \to B \to C$, de manera que

$$f((a, b)) :\equiv g(a)(b) \equiv g(a, b).$$

Amb aquest principi, sabem que per a definir una funció amb domini un producte, només necessitem donar una funció g. Per exemple, utilitzant aquest principi podem definir les *projeccions* d'un producte. Siguin les funcions $g_1, g_2 : A \to B \to C$ tals que $g_1(a, b) :\equiv a$ i $g_2(a, b) :\equiv b$. Aplicant el principi de recursió, obtenim les funcions que denotarem $\mathtt{pr}_1 : A \times B \to A$ i $\mathtt{pr}_2 : A \times B \to B$ definides per

$$\mathtt{pr}_1((a, b)) :\equiv a,$$
$$\mathtt{pr}_2((a, b)) :\equiv b.$$

Per últim, vegem el principi d'inducció del producte. Sigui $P : A \times B \to \mathcal{U}$ una família qualsevol. Definim $f : \prod_{x:A \times B} P(x)$ a partir d'una funció $g : \prod_{x:A} \prod_{y:B} C((x, y))$ amb la qual

$$f((a, b)) :\equiv g(a)(b) \equiv g(a, b).$$

Productes dependents

Ara presentarem la generalització dependent del producte ordinari. Serà un producte en el qual el tipus del segon element dependrà del primer element. Donat un tipus qualsevol A i una família $B : A \to \mathrm{U}$, definirem $\sum_{x:A} B(x)$ com el *producte dependent* de A i B. Si la família B és constant, aquest tipus serà igual al producte $A \times B$.

La regla d'introducció d'aquest tipus ens diu que, donats un element a de A i un element b de $B(a)$, podem crear un element $(a, b) : \sum_{x:A} B(x)$.

Sigui C un tipus qualsevol. El principi de recursió d'aquest tipus ens diu que, donada una funció $g : \prod_{x:A} B(x) \to C$, existeix una funció $f : (\sum_{x:A} B(x)) \to C$ definida per l'equació

$$f((a,b)) :\equiv g(a,b).$$

Finalment, presentem el seu principi d'inducció. Sigui $C : (\sum_{x:A} B(x)) \to \mathrm{U}$ una família. Si elegim una funció $g : \prod_{x:A} \prod_{y:B(x)} C((x, y))$, llavors existeix

$$f : \prod_{(a,b):\sum_{x:A} B(x)} C((a,b))$$

definida per

$$f((a,b)) :\equiv g(a,b).$$

De manera similar al cas del producte no dependent, existeixen les *projeccions dependents*. Les definirem utilitzant els principis de recursió i d'inducció, ja que la segona projecció d'un producte dependent és una funció dependent. Anomenarem

$$\mathtt{pr}_1 : \left(\sum_{x:A} B(x)\right) \to A$$

$$\mathtt{pr}_2 : \prod_{w:\sum_{x:A} B(x)} B(\mathtt{pr}_1(w))$$

a les funciones de les projeccions definides per

$$\mathtt{pr}_1((a,b)) :\equiv a,$$
$$\mathtt{pr}_2((a,b)) :\equiv b.$$

Observem que hem donat el mateix nom a les projeccions dependents que a les normals, ja que en realitat les projeccions no dependents només són un cas particular de les dependents amb una família constant.

Tipus suma

Un altre dels tipus més utilitzats és el tipus suma, que representa el mateix concepte que el coproducte de teoria de categories. Donats dos tipus qualsevol A i B, anomenarem al tipus $A + B$ la *suma* d'aquests. Aquest tipus també correspon a la unió disjunta de conjunts.

Per aquest tipus tenim dues regles d'introducció. La primera, ens diu que a partir d'un element $a : A$ introduïm l'element $\mathrm{inl}(a) : A + B$. La segona, que a partir d'un element $b : B$ introduïm l'element $\mathrm{inr}(b) : A + B$.

Pel principi de recursió, considerem un tipus qualsevol C. Si elegim dues funcions $g_0 : A \to C$ i $g_1 : B \to C$, llavors existeix la funció $f : (A + B) \to C$, que podem definir per casos:

$$\begin{cases} f(\mathrm{inl}(a)) :\equiv g_0(a) \\ f(\mathrm{inr}(b)) :\equiv g_1(b). \end{cases}$$

Aquesta és la primera definició per casos del treball i, segons si l'element de $A + B$ és dels introduïts per una regla o l'altra, tindrà una imatge o l'altra.

Pel principi d'inducció, considerem una família qualsevol $C : (A + B) \to \mathrm{U}$. Si elegim dues funcions $g_0 : \prod_{x:A} C(x)$ i $g_1 : \prod_{x:B} C(x)$, llavors existeix la funció $f : \prod_{x:A+B} C(x)$ que podem definir per casos:

$$\begin{cases} f(\mathrm{inl}(a)) :\equiv g_0(a) \\ f(\mathrm{inr}(b)) :\equiv g_1(b). \end{cases}$$

Tipus unitat

El *tipus unitat* $\mathbf{1}$ representa el tipus amb un sol element, que denotarem per $\star : \mathbf{1}$. És similar a un conjunt que conté el conjunt buit a teoria de conjunts, o als objectes finals de teoria de categories.

Aquest tipus només té una regla d'introducció: la que ens permet instanciar l'element $\star : \mathbf{1}$. El seu principi de recursió ens diu que, donat un element c d'un tipus qualsevol C, podem definir una funció $f : \mathbf{1} \to C$ tal que $f(\star) :\equiv c$.

Per acabar, donats una família $C : \mathbf{1} \to \mathrm{U}$ i un element $c : C(\star)$, el principi d'inducció de la unitat ens diu que podem construir una funció dependent $f : \prod_{x:\mathbf{1}} C(x)$ tal que $f(\star) :\equiv c$.

Tipus buit

Un altre tipus a definir és l'equivalent al conjunt buit, el *tipus buit*, que denotarem $\mathbf{0}$. El tipus buit té moltes similituds amb els objectes inicials a teoria de categories i amb el conjunt buit a teoria de conjunts. El tipus $\mathbf{0}$ no té elements, i per tant no té cap regla d'introducció.

El principi de recursió ens dirà que, donat un tipus qualsevol C, sempre existirà una funció $f : \mathbf{0} \to C$ sense cap equació de definició, ja que no hi ha cap element de $\mathbf{0}$ amb el qual es pugui definir la imatge de f.

De manera molt similar, el principi d'inducció ens permet crear una funció dependent cap a una família qualsevol $C : \mathbf{0} \to \mathrm{U}$. Per tant, sempre existirà $f : \prod_{x:\mathbf{0}} C(x)$, sense cap equació per la manca d'elements.

Igualtats

Definirem la *igualtat proposicional* o *identificació* d'un tipus A entre dos elements x i y com el tipus $x =_A y$. El fet que la igualtat sigui un tipus resulta un fet poc natural al principi, però encaixarà amb la interpretació de proposicions com a tipus. Donant una petita introducció a l'apartat següent, els elements d'un tipus igualtat seran demostracions d'aquesta. Si fixem el tipus base, podem veure les identificacions a aquest tipus com la família

$$\cdot =_A \cdot : A \to A \to \mathrm{U}.$$

Ara que ja hem presentat el tipus de la igualtat, la pregunta natural a fer-nos és com introduïm elements. En aquest tipus, trobem una única regla d'introducció, que ens diu que per a tot tipus A i tot element $a : A$, existeix l'element

$$\mathrm{refl}_a : a =_A a.$$

Observem que, si tenim una igualtat per definició $x \equiv y : A$, també tenim una igualtat proposicional, ja que

$$\mathrm{refl}_x : (x =_A x) \equiv (x =_A y).$$

Ens queda saber com podem utilitzar els elements d'aquest tipus. Comencem amb el principi de recursió. Sigui $C : A \to A \to \mathrm{U}$ una família qualsevol i un element $a : A$. Si escollim un element $c : \prod_{x:A} C(x,x)$, existirà una funció $f : \prod_{x,y:A}(x =_A y) \to C(x,y)$ a partir de l'equació

$$f(a, a, \mathrm{refl}_a) :\equiv c(a).$$

Existeix un cas particular del principi de recursió, anomenat *indiscernibilitat d'iguals*. Aquest cas considera el principi de recursió aplicat a $C(x,y) :\equiv C(x) \to C(y)$. Aquesta propietat ens comunica que les igualtats es conserven.

El principi d'inducció per a les igualtats també té nom propi: l'anomenarem *inducció de camins*. Considerarem la família $C : \prod_{x,y:A}(x =_A y) \to C(x,y)$ i un element $a : A$. Donat un element $c(x) : \prod_{x:A} C(x,x,\mathrm{refl}_x)$, existeix la funció

$$f : \prod_{x,y:A} \prod_{p:x=_A y} C(x,y,p)$$

definida per l'equació

$$f(a, a, \mathrm{refl}_a) :\equiv c(a).$$

A l'aplicar aquest principi inductiu, podem demostrar propietats parametritzades per igualtats simplement considerant el cas reflexiu. Amb una demostració del cas reflexiu, creem una demostració pel cas genèric. Podríem relacionar-ho amb les demostracions clàssiques per inducció de nombres naturals, però en comptes de haver de demostrar el cas base i el $n+1$ a partir del n, només cal demostrar el cas base.

1.1.3 Proposicions

Com havíem esmentat en la introducció d'aquest apartat, un dels punts on més es diferencien la teoria de tipus i la de conjunts es en el seu tractament de la lògica. La teoria de conjunts es construeix al damunt d'un sistema formal lògic establert, i en canvi la teoria de tipus construeix la lògica dintre seu.

Aquest fet té implicacions molt sorprenents. Les proposicions són els mateixos tipus que utilitzem per a formalitzar els objectes matemàtics. Qualsevol tipus pot ser interpretat com una proposició, però en general només tindrà un sentit inherent en aquells que representin una proposició com a tal.

A part de les diferències de l'ordre de construcció, la teoria de tipus que construirem utilitzarà lògica intuïcionista en lloc de la clàssica. La lògica intuïcionista és un sistema lògic més feble que la lògica clàssica, on tenim els mateixos axiomes excepte la llei del tercer exclòs. En aquesta teoria, un tipus pot ser cert si tenim proves, fals si tenim proves que no pot ser cert, o cap dels dos casos, si no tenim proves de cap dels dos. És una lògica constructiva, més propera a la computació.

El canvi de lògica no proporciona limitacions matemàtiques, ja que sempre podem afegir la llei del tercer exclòs com a hipòtesi a les proposicions on la necessitem per a la demostració, de manera similar a com es tracta el axioma de l'elecció a teoria de conjunts. A més, aquest mètode de treball prioritza demostracions constructives, i fa fàcil de trobar totes les no constructives, fent explícit quines utilitzen la llei del tercer exclòs.

Llavors, en la interpretació de proposicions com a tipus, considerarem que un tipus és cert si està habitat, i que un tipus A és fals si el tipus $A \to \mathbf{0}$ està habitat.

Teoria de conjunts	**Teoria de tipus**
Cert	$\mathbf{1}$
Fals	$\mathbf{0}$
Negació de A	$A \to \mathbf{0}$
A i B	$A \times B$
A o B	$A + B$
A implica B	$A \to B$
A si, i només si B	$(A \to B) \times (B \to A)$
$\forall x \in A,\ B(x)$	$\prod_{x \in A} B(x)$
$\exists x \in A,\ B(x)$	$\sum_{x \in A} B(x)$

Taula 1.1: Interpretacions de les proposicions com a tipus

Utilitzant els tipus definits fins ara podem crear certes estructures complexes. Per exemple, podem construir el tipus dels semigrups:

$$Semigrup :\equiv \sum_{A:\mathcal{U}} \sum_{m:A \to A \to A} \prod_{x,y,z:A} m(x, m(y,z)) =_A m(m(x,y),z).$$

Com podem veure, s'utilitzen els tipus tant per a formalitzar el tipus base que serà el semigrup, la funció que serà l'operació inherent a aquest i les propietats que complirà. Si interpretem la part final del tipus com a proposició, aquesta ens diu que per a tot $x, y, z : A$ és complirà $m(x, m(y,z)) =_A m(m(x,y),z)$, és a dir, que l'operació m serà associativa per a elements de A.

Podem introduir un element d'aquest tipus donant el tipus base, l'operació que vulguem utilitzar i una demostració de la seva associativitat, en forma de funció dependent cap a una igualtat.

Per exemple, amb les eines introduïdes fins ara podem veure que el tipus unitat forma un semigrup amb el seu únic element. Utilitzant el principi de recursió de $\mathbf{1}$, definim la funció $m : \mathbf{1} \to \mathbf{1} \to \mathbf{1}$ tal que $m(\star, \star) :\equiv \star$. Llavors existirà una funció dependent $p : \prod_{x,y,z:\mathbf{1}} m(x, m(y,z)) = m(m(x,y),z)$ que, pel principi d'inducció de la unitat i per com hem definit m, podem definir amb l'equació

$$p(\star,\star,\star) :\equiv \mathsf{refl}_\star .$$

Per tant, podem veure que **1** forma un semigrup introduint l'element

$$(\mathbf{1},\, m,\, p) : \mathit{Semigrup}.$$

Per acabar, vegem un exemple de com demostraríem una proposició: el principi d'unicitat del tipus **1**. En aquest cas, el principi és una igualtat proposicional, i es pot demostrar directament a partir del principi d'inducció. La propietat ens diu que

$$\prod_{x:\mathbf{1}} x = \star,$$

que podríem llegir com que tot element de **1** és igual a $\star$.

Per a demostrar-ho, necessitem construir un element d'aquest tipus. Aplicant el principi d'inducció del tipus **1**, si escollim un element de tipus $\star = \star$ podrem definir la funció dependent que volem. Llavors, escollint $\mathsf{refl}_\star : \star = \star$, existirà una funció dependent $p : \prod_{x:\mathbf{1}} x = \star$ definida per

$$p(\star) :\equiv \mathsf{refl}_\star .$$

Finalment, per la interpretació de proposicions com a tipus, el fet de tenir un element equival al fet que la proposició sigui certa.

1.2 Teoria homotòpica de tipus

La teoria homotòpica de tipus és una extensió de la teoria de tipus intuïcionista. La principal diferència és la interpretació dels tipus com a espais de teoria abstracta d'homotopia i com a grupoides d'ordre superior, que anomenarem ∞-grupoides, de teoria de categories.

En la teoria d'homotopia, un espai topològic és un conjunt de punts X amb una topologia. Un camí p a X de x a y és una aplicació continua $p : [0,1] \to X$ tal que $p(0) = x$ i $p(1) = y$. Llavors, podem considerar una aplicació continua $\alpha : [0,1] \to \{\, [0,1] \to X \,\}$ tal que $\alpha(0) = p$ i $\alpha(1) = q$, on p i q són camins de x a y, i aquesta aplicació és un camí entre camins o homotopia.

El següent pas és definir les homotopies entre homotopies, anomenades 2-homotopies. Podem continuar aquest procés iteratiu de definició, arribant a definir inductivament les n-homotopies.

L'espai base X, els camins i les n-homotopies per a tot n formen una construcció que dona lloc a un ∞-grupoide, anomenat ∞-grupoide fonamental de X. Un ∞-grupoide és una categoria d'ordre infinit on tots els morfismes són invertibles. En aquest treball no donarem una definició precisa de les categories d'ordre infinit perquè necessitaríem introduir molts conceptes de teoria de categories que excedeixen els nostres objectius. En comptes de definir el concepte formalment, utilitzarem el ∞-grupoide fonamental com a exemple.

Tornant a teoria de tipus, considerem un tipus A qualsevol. Si $x, y : A$ són dos punts de A, el tipus $x =_A y$ conté identificacions entre aquests dos punts. Podem veure aquestes identificacions de manera similar a com veiem els camins en un espai topològic.

Si $p, q : x =_A y$ són identificacions entre x i y, podem considerar el tipus $p =_{x=_A y} q$, que conté identificacions entre aquestes dues identificacions. Anomenarem 2-igualtats a les identificacions entre identificacions.

Llavors, seguint el mateix procediment que en el cas d'espais topològics, podem construir una successió de tipus basats en el tipus A. Concretament, definirem els tipus de les n-igualtats per a tot n.

L'objectiu d'aquesta secció es veure que podem associar a cada tipus un ∞-grupoide, de manera totalment anàloga a com ho fem amb els espais topològics. Per tant, ens centrarem en comprovar que els tipus compleixen les mateixes propietats que els espais topològics amb el seu ∞-grupoide fonamental.

1.2.1 Els tipus com a ∞-grupoides

En aquest apartat veurem com associar un ∞-grupoide a cada tipus. Primer de tot necessitem definir en quins espais treballarem. En el cas topològic serà l'espai base, els camins i les n-homotopies per a tot n. En el de teoria de tipus serà el tipus base, les igualtats i les n-igualtats per a tot n. Podem veure una relació d'aquestes successions d'estructures a la taula següent:

Nivell	**Espais topològics**	**Tipus**
0	X	A
1	$\{ [0,1] \to X \}$	$\sum_{x,y:A} (x = y)$
2	$\{ [0,1] \to \{ [0,1] \to X \} \}$	$\sum_{x,y:A} \sum_{p_1,q_1 : x=y} (p_1 = q_1)$
$\vdots$	$\vdots$	$\vdots$
n	$\{ [0,1] \to \{ \cdots \{ [0,1] \to X \} \} \}$	$\sum_{x,y:A} \sum_{p_1,q_1 : x=y} \cdots \sum_{p_{n+1},q_{n+1} : p_n = q_n} (p_{n+1} = q_{n+1})$
$\vdots$	$\vdots$	$\vdots$

Taula 1.2: Comparació de l'estructura de ∞-grupoide en un espai topològic i en un tipus

Anomenarem C_n a l'enèsim conjunt de la successió d'homotopies en un espai topològic. Llavors, considerarem una relació d'equivalència $\sim_n$ a cada nivell n de la successió, definida per

$$\forall \alpha_n, \beta_n \in C_n,\ \alpha_n \sim_n \beta_n \iff \exists \gamma_{n+1} \in C_{n+1} \text{ tal que } \gamma_{n+1}(0) = \alpha_n \text{ i } \gamma_{n+1}(1) = \beta_n.$$

Aquesta relació d'equivalència ens indueix una partició dels conjunts C_n en classes d'equivalència, que denotarem $[\alpha_n]$, per a tot $\alpha_n \in C_n$. La successió de conjunts $C_n / \sim_n$

per a cada n és l'estructura que configura un ∞-grupoide. Observem que, a més, si existeix $\gamma_{n+1} \in C_{n+1}$ entre $\alpha_n, \beta_n \in C_n$, les homotopies α_n i β_n compliran $\alpha_n(0) = \beta_n(0)$ i $\alpha_n(1) = \beta_n(1)$. Anomenarem *llevat d'homotopia d'ordre superior* a les propietats entre elements d'un nivell n que es defineixen utilitzant la relació $\sim_n$ en comptes de la igualtat.

En el cas dels tipus, cada nivell és un producte dependent iterat amb una família d'igualtats $p_n = q_n$. És a dir, per cada parell d'elements p_n, q_n del nivell n, existirà un tipus $p_n = q_n$ habitat, o no, per igualtats entre p_n i q_n. Considerem dues n-igualtats $p_{n+1}, q_{n+1} : p_n = q_n$. Si existeix una igualtat $p_{n+2} : p_{n+1} = q_{n+1}$ al nivell superior, aleshores direm que p_{n+1} i q_{n+1} estan identificats. Com que estem considerant la mateixa igualtat com la relació entre elements, totes les igualtats seran *llevat d'igualtats d'ordre superior*.

Vegem quines són les propietats que caracteritzen les classes d'homotopia per a formar el ∞-grupoide fonamental. Per a cada nivell $n \geq 1$, considerem $\alpha_n \in C_n$ tal que $\alpha_n(0) = \gamma_{n-1}$ i $\alpha_n(1) = \delta_{n-1}$. Llavors:

- Existeix una operació de composició per a cada nivell n. Si $\beta_n \in C_n$ és tal que $\beta_n(0) = \delta_{n-1}$ i $\beta_n(1) = \epsilon_{n-1}$, existirà la composició entre α_n i β_n, que denotarem $\alpha_n \cdot \beta_n$, tal que $(\alpha_n \cdot \beta_n)(0) = \gamma_{n-1}$ i $(\alpha_n \cdot \beta_n)(1) = \epsilon_{n-1}$. A més, aquesta operació és associativa.

- La composició també està definida per a classes de la relació $\sim_n$ com la composició de representants, és a dir, $[\alpha_n] \cdot [\beta_n] = [\alpha_n \cdot \beta_n]$.

- Existeixen elements $\mathrm{refl}_{\gamma_{n-1}}, \mathrm{refl}_{\delta_{n-1}} \in C_n$ tals que

$$\begin{aligned} \mathrm{refl}_{\gamma_{n-1}}(0) = \mathrm{refl}_{\gamma_{n-1}}(1) = \gamma_{n-1} \quad &\text{i} \quad [\alpha_n] \cdot [\mathrm{refl}_{\delta_{n-1}}] = [\alpha_n], \\ \mathrm{refl}_{\delta_{n-1}}(0) = \mathrm{refl}_{\delta_{n-1}}(1) = \delta_{n-1} \quad &\text{i} \quad [\mathrm{refl}_{\gamma_{n-1}}] \cdot [\alpha_n] = [\alpha_n]. \end{aligned}$$

- Existeix un element invers ${\alpha_n}^{-1} \in C_n$, tal que ${\alpha_n}^{-1}(0) = \delta_{n-1}$, ${\alpha_n}^{-1}(1) = \gamma_{n-1}$ i

$$\begin{aligned} [\alpha_n] \cdot [{\alpha_n}^{-1}] &= [\mathrm{refl}_{\gamma_{n-1}}], \\ [{\alpha_n}^{-1}] \cdot [\alpha_n] &= [\mathrm{refl}_{\delta_{n-1}}]. \end{aligned}$$

Volem veure que els tipus compleixen totes aquestes propietats, i per tant, cada tipus dona lloc a un ∞-grupoide. Com que una igualtat $x = y$ entre elements $x, y : A$ és un tipus, només necessitem veure les propietats anteriors per a igualtats sobre un tipus qualsevol A, ja que d'aquesta manera ho haurem demostrat per a les n-igualtats, que són igualtats sobre els tipus de les $(n-1)$-igualtats. Comencem per definir la inversa i la composició d'igualtats:

Lema 1.4. *Sigui A un tipus qualsevol i $x, y : A$. Llavors existeix una funció*

$$\begin{aligned} (x = y) &\to (y = x) \\ p &\mapsto p^{-1} \end{aligned}$$

tal que ${\mathrm{refl}_x}^{-1} \equiv \mathrm{refl}_x$ *per a tot $x : A$.*

Anomenarem p^{-1} *l'invers de p.*

Demostració. Per a cada $x, y : A$ i $p : x = y$ volem definir $p^{-1} : y = x$. Per inducció de camins, és suficient considerar el cas $x \equiv y$ i $p \equiv \mathrm{refl}_x$.

En aquest cas, volem definir $\mathrm{refl}_x{}^{-1}$, que ha de ser del tipus $x = x$. Clarament podem fer-ho amb $\mathrm{refl}_x{}^{-1} :\equiv \mathrm{refl}_x$.

Per tant, per l'aplicació de la inducció de camins, tenim una definició de p^{-1} per a tot $p : x = y$. □

Lema 1.5. *Sigui A un tipus qualsevol i $x, y, z : A$. Llavors existeix una funció*

$$\begin{array}{ccccc} (x = y) & \to & (y = z) & \to & (x = z) \\ p & \mapsto & q & \mapsto & p \cdot q \end{array}$$

tal que $\mathrm{refl}_x \cdot \mathrm{refl}_x \equiv \mathrm{refl}_x$ *per a tot $x : A$.*

Anomenarem a aquesta funció *composició d'igualtats*.

Demostració. Per demostrar aquesta proposició, utilitzarem dues inducions de camins: una per a p i una per a q. De manera que suposem $x \equiv y \equiv z$ i $p \equiv q \equiv \mathrm{refl}_x$.

Llavors, com que volem definir $\mathrm{refl}_x \cdot \mathrm{refl}_x : x = x$, podem agafar $\mathrm{refl}_x \cdot \mathrm{refl}_x :\equiv \mathrm{refl}_x$. Per tant, hem demostrat l'existència de la funció i la propietat sobre aquesta. □

Amb aquestes dues definicions i existències, ja podem enunciar la resta de propietats. L'existència de l'element refl a cada igualtat la tenim gràcies a la regla d'introducció de les igualtats. Com hem esmentat abans, les igualtats de classes d'homotopia corresponen a igualtats d'ordre superior a teoria de tipus. Llavors, podem veure:

Lema 1.6. *Siguin $A : U$; $a, b, c, d : A$; $p : a = b$; $q : b = c$ i $r : c = d$. Tenim les següents propietats:*

(i) $p = p \cdot \mathrm{refl}_b$ *i* $p = \mathrm{refl}_a \cdot p$

(ii) $p^{-1} \cdot p = \mathrm{refl}_b$ *i* $p \cdot p^{-1} = \mathrm{refl}_a$

(iii) $(p^{-1})^{-1} = p$

(iv) $p \cdot (q \cdot r) = (p \cdot q) \cdot r$

Demostració.

(*i*) Aplicant inducció de camins a p, és suficient considerar el cas $a \equiv b$ i $p \equiv \mathrm{refl}_a$. Llavors,
$$\mathrm{refl}_a \equiv \mathrm{refl}_a \cdot \mathrm{refl}_a .$$

(*ii*) Apliquem inducció de camins a p, considerant $a \equiv b$ i $p \equiv \mathrm{refl}_a$, obtenint
$$\mathrm{refl}_a{}^{-1} \cdot \mathrm{refl}_a \equiv \mathrm{refl}_a \cdot \mathrm{refl}_a \equiv \mathrm{refl}_a .$$

(*iii*) Com en els casos anteriors, apliquem inducció de camins a p considerant $a \equiv b$ i $p \equiv \mathrm{refl}_a$. Llavors,
$$(\mathrm{refl}_a{}^{-1})^{-1} \equiv \mathrm{refl}_a{}^{-1} \equiv \mathrm{refl}_a .$$

(*iv*) Aplicarem inducció de camins a p, q i r, obtenint

$$\mathrm{refl}_a \cdot (\mathrm{refl}_a \cdot \mathrm{refl}_a) \equiv \mathrm{refl}_a \cdot \mathrm{refl}_a \equiv (\mathrm{refl}_a \cdot \mathrm{refl}_a) \cdot \mathrm{refl}_a\,. \qquad \square$$

Per tant, ja hem vist que efectivament podem associar a cada tipus un ∞-grupoide, ja que observem en els tipus la mateixa estructura i propietats que en el cas dels espais topològics. Ara veurem quines implicacions tindrà aquesta relació en la resta del llenguatge.

Finalment, veurem un altre tipus de composició: la composició horitzontal. Suposem que tenim dos elements obtinguts per composició d'igualtats $p_1 \cdot q_1$ i $p_2 \cdot q_2$, on p_1 i p_2 comparteixen tipus, així com q_1 i q_2. Aleshores, si tenim dues igualtats "horitzontals" a aquestes, és a dir, $\alpha : p_1 = p_2$ i $\beta : q_1 = q_2$, podem combinar-les en una sola igualtat de $p_1 \cdot q_1$ a $p_2 \cdot q_2$. Aquesta ens servirà més endavant per a definir transformacions entre composicions d'igualtats.

Lema 1.7. *Sigui A un tipus qualsevol; $x, y, z : A$; $p_1, p_2 : x = y$; $q_1, q_2 : y = z$; $\alpha : p_1 = p_2$ i $\beta : q_1 = q_2$. Llavors existeix l'operació*

$$\begin{array}{ccccc} (p_1 = p_2) & \to & (q_1 = q_2) & \to & (p_1 \cdot q_1 = p_2 \cdot q_2) \\ \alpha & \mapsto & \beta & \mapsto & \alpha * \beta \end{array}$$

Anomenarem a $\alpha * \beta$ la *composició horitzontal de α i β.*

Demostració. Per a definir la composició horitzontal necessitem definir dues composicions més. Considerem $\alpha \cdot_r q_1 : p_1 \cdot q_1 = p_2 \cdot q_1$. Demostrarem per inducció de camins sobre α que aquesta composició sempre existeix. Suposem $p_1 \equiv p_2$ i $\alpha \equiv \mathrm{refl}_{p_1}$. Llavors

$$\mathrm{refl}_{p_1} \cdot_r q_1 :\equiv \mathrm{refl}_{p_1 \cdot q_1} : p_1 \cdot q_1 = p_1 \cdot q_1.$$

Podem aplicar exactament el mateix raonament per definir $p_2 \cdot_r \beta : p_2 \cdot q_1 = p_2 \cdot q_2$. Finalment, podem definir $\alpha * \beta :\equiv (\alpha \cdot_r q_1) \cdot (p_2 \cdot_l \beta)$. $\square$

1.2.2 Les funcions com a functors

Donat que considerem ∞-grupoides associats a tipus, es natural pensar que els tipus de les funcions correspondran a functors entre ∞-grupoides.

Les funcions ens envien elements del tipus del domini a elements del tipus del codomini. Per tal que corresponguin a functors entre ∞-grupoides, hauran de respectar les n-igualtats per a tot n.

Amb respectar les igualtats ens referim al fet que, si existeix una igualtat en el tipus del domini, haurà d'existir "la imatge" d'aquesta en el codomini, i ser una igualtat entre les imatges dels seus extrems. Com en el cas dels tipus, és suficient estudiar igualtats sobre un tipus A qualsevol, i ens caracteritzaran les n-igualtats sobre $(n-1)$-igualtats. Per tant, començarem definint com aplicar funcions a igualtats:

Lema 1.8. *Sigui $f : A \to B$ una funció. Per a tot parell x, $y : A$ existeix una operació*

$$\mathtt{ap}_f : (x =_A y) \to (f(x) =_B f(y)),$$

i a més, per a cada $a : A$ tenim $\mathtt{ap}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)}$.

Anomenarem a $\mathtt{ap}_f$ *l'aplicació de f a igualtats*.

Demostració. Volem definir $\mathtt{ap}_f(p) : f(x) =_B f(x)$ per a tot $p : x =_A y$. Per inducció de camins, és suficient considerar el cas $x \equiv y$ i $p \equiv \mathrm{refl}_x$. Per tant, com que sabem que existeix $\mathrm{refl}_{f(x)} : f(x) =_B f(x)$, deduïm que $\mathtt{ap}_f(refl_x) :\equiv \mathrm{refl}_{f(x)}$. □

Finalment necessitem comprovar la resta de condicions de functorialitat:

Lema 1.9. *Siguin $f : A \to B$, $g : B \to C$, $p : x =_A y$ i $q : y =_A z$. Es compleix:*

(i) $\mathtt{ap}_f(p \cdot q) = \mathtt{ap}_f(p) \cdot \mathtt{ap}_f(q)$

(ii) $\mathtt{ap}_f(p^{-1}) = \mathtt{ap}_f(p)^{-1}$

(iii) $\mathtt{ap}_g(\mathtt{ap}_f(p)) = \mathtt{ap}_{g\circ f}(p)$

(iv) $\mathtt{ap}_{\mathtt{id}_A}(p) = p$

Demostració.

(*i*) Aplicant inducció de camins a p i a q, és suficient considerar el cas $x \equiv y \equiv z$ i $p \equiv q \equiv \mathrm{refl}_x$. Llavors,

$$\mathtt{ap}_f(\mathrm{refl}_x \cdot \mathrm{refl}_x) \equiv \mathtt{ap}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)} \equiv \mathrm{refl}_{f(x)} \cdot \mathrm{refl}_{f(x)} \equiv \mathtt{ap}_f(\mathrm{refl}_x) \cdot \mathtt{ap}_f(\mathrm{refl}_x).$$

(*ii*) Per inducció de camins amb $x \equiv y$ i $p \equiv \mathrm{refl}_x$, veiem que

$$\mathtt{ap}_f((\mathrm{refl}_x)^{-1}) \equiv \mathtt{ap}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)} \equiv (\mathrm{refl}_{f(x)})^{-1} \equiv (\mathtt{ap}_f(\mathrm{refl}_x))^{-1}.$$

(*iii*) Per inducció de camins amb $x \equiv y$ i $p \equiv \mathrm{refl}_x$, veiem que

$$\mathtt{ap}_g(\mathtt{ap}_f(\mathrm{refl}_x)) \equiv \mathtt{ap}_g(\mathrm{refl}_{f(x)}) \equiv \mathrm{refl}_{(g\circ f)(x)} \equiv \mathtt{ap}_{g\circ f}(\mathrm{refl}_x).$$

(*iv*) Com als dos casos anteriors, per inducció de camins amb $x \equiv y$ i $p \equiv \mathrm{refl}_x$, obtenim

$$\mathtt{ap}_{\mathtt{id}_A}(\mathrm{refl}_x) \equiv \mathrm{refl}_{\mathtt{id}(x)} \equiv \mathrm{refl}_x\,.$$ □

Per tant, hem vist que les funcions es poden associar a functors entre ∞-grupoides i respecten l'estructura que hem associat als tipus.

Més endavant, quan presentem estructures complexes, necessitarem aplicar funcions del tipus base a 2-igualtats. Vegem com fer-ho:

Lema 1.10. *Siguin $f : A \to B$; $x, y : A$; $p, q : x = y$ i $r : p = q$. Llavors existeix la igualtat*

$$\mathtt{ap}^2_f(r) : \mathtt{ap}_f(p) = \mathtt{ap}_f(q).$$

Anomenarem a aquesta funció *aplicació de f a 2-igualtats*.

Demostració. Podem definir aquesta funció amb l'equació $\mathtt{ap}^2_f(r) :\equiv \mathtt{ap}_{\mathtt{ap}_f}(r)$. Llavors, per les propietats de $\mathtt{ap}$, tindrem $\mathtt{ap}^2_f(\mathrm{refl}_p) \equiv \mathtt{ap}_{\mathtt{ap}_f}(\mathrm{refl}_p) \equiv \mathrm{refl}_{\mathtt{ap}_f(p)}$. □

1.2.3 Les famílies com a fibrats

De manera similar a com ens hem preguntat com aplicàvem funcions a igualtats, podríem fer la mateixa pregunta per a funcions dependents. En aquests cas, trobarem que una funció dependent $f : \prod_{x:A} E(x)$, on E és una família de A, correspon a seccions d'un fibrat definit per la família E en el ∞-grupoide associat.

El primer dubte amb les funcions dependents és com obtindrem una igualtat de sortida, si a priori les imatges dels extrems de la igualtat poden anar a parar a tipus diferents de la família. Per a relacionar les imatges d'una funció dependent necessitem un nou concepte: el transport.

Lema 1.11. *Siguin $E : A \to U$ i $x, y : A$. Existeix una funció*

$$\mathtt{trans}_E : (x =_A y) \to (E(x) \to E(y)).$$

Anomenarem a aquesta funció *transport de p a través de E.*

Demostració. Sigui $p : x =_A y$. Utilitzem inducció de camins, suposant $x \equiv y$ i $p \equiv \mathrm{refl}_x$. En aquest cas, $\mathtt{trans}_E(\mathrm{refl}_x) : E(x) \to E(x)$, i podem definir $\mathtt{trans}_E(\mathrm{refl}_x) :\equiv \mathtt{id}_{E(x)}$. □

Utilitzant el transport podem definir el concepte anàleg a l'aplicació de funcions ordinàries a igualtats.

Lema 1.12. *Suposem $x, y : A$, $E : A \to U$ i $f : \prod_{x:A} E(x)$. Llavors existeix la funció*

$$\mathtt{apd}_f : \prod_{p:x=_A y} \mathtt{trans}_E(p, f(x)) =_{E(y)} f(y).$$

Anomenarem a aquesta funció *aplicació dependent de f a igualtats.*

Demostració. Per inducció de camins, considerem $x \equiv y$ i $p \equiv \mathrm{refl}_x$. En aquest cas, pel lema del transport sabem que

$$\mathtt{trans}_E(\mathrm{refl}_x, f(x)) \equiv \mathtt{id}_{E(x)}(f(x)) \equiv f(x).$$

Per tant, $\mathtt{apd}_f(\mathrm{refl}_x) : f(x) =_{E(x)} f(x)$, i podem definir-ho com $\mathtt{apd}_f(\mathrm{refl}_x) :\equiv \mathrm{refl}_{f(x)}$. □

També podem preguntar-nos com elevar un igualtat al espai total, és a dir, al producte dependent generat per la família. En aquest cas, ens trobem un altre ús del transport.

Lema 1.13. *Sigui $E : A \to \mathrm{U}$; $x, y : A$; $p : x = y$ i $u : E(x)$. Llavors existeix la següent identificació a $\sum_{a:A} E(a)$:*

$$\mathtt{lift}_E(u, p) : (x, u) = (y, \mathtt{trans}_E(p, u)).$$

Anomenarem a aquesta identificació *elevació de la igualtat p a u.*

Demostració. Per inducció de camins, considerem $x \equiv y$ i $p \equiv \mathrm{refl}_x$. Llavors com que

$$\mathtt{trans}_E(\mathrm{refl}_x, u) \equiv u,$$

podem definir $\mathtt{lift}(u, \mathrm{refl}_x) :\equiv \mathrm{refl}_{(x,u)} : (x, u) = (x, u)$. □

Finalment, plantejarem els anàlegs a aquests conceptes aplicats a 2-igualtats, com hem fet per a les funcions, ja que els necessitarem més endavant. El concepte del `lift` dos-dimensional el definirem en una altra secció, ja que necessitem un teorema que encara no tenim.

Lema 1.14. *Siguin $E : A \to \mathrm{U}$; $x, y : A$ i $p, q : x = y$. Definim $\widetilde{E} : (x = y) \to \mathrm{U}$ per $\widetilde{E}(u) :\equiv \mathtt{apd}_f(u)$. Llavors existeix*

$$\mathtt{trans}^2_E : (p = q) \to \widetilde{E}(p) \to \widetilde{E}(q).$$

Anomenarem a aquesta funció *transport de 2-igualtats a través de E*.

Demostració. Per inducció de camins, suposem que $p \equiv q$ i $r :\equiv \mathrm{refl}_p$. Però llavors podem definir $\mathtt{trans}^2_E(\mathrm{refl}_p) :\equiv \mathtt{id}_{\widetilde{E}(p)}$. □

Lema 1.15. *Siguin $E : A \to \mathrm{U}$; $f : \prod_{x:A} E(x)$; $x, y : A$; $p, q : x = y$ i $r : p = q$. Llavors existeix la igualtat*

$$\mathtt{apd}^2_f(r) : \mathtt{trans}^2_E(r, \mathtt{apd}_f(p)) = \mathtt{apd}_f(q).$$

Anomenarem a aquesta funció *aplicació dependent de f a 2-igualtats*.

Demostració. Definim la funció amb l'equació $\mathtt{apd}^2_f(r) :\equiv \mathtt{apd}_{\mathtt{apd}_f}(r)$. A més, per les propietats de `apd`, tindrem

$$\mathtt{apd}^2_f(\mathrm{refl}_p) \equiv \mathtt{apd}_{\mathtt{apd}_f}(\mathrm{refl}_p) \equiv \mathrm{refl}_{\mathtt{apd}_f(p)}.$$

□

Lema 1.16. *Siguin $E : A \to \mathrm{U}$; $x, y : A$; $p, q : x = y$ i $r : p = q$. Per a tot $u : E(x)$, existeix*

$$\mathtt{transConnect}_E(r, u) : \mathtt{trans}_E(p, u) = \mathtt{trans}_E(q, u).$$

Demostració. Per inducció de camins, suposem que $p \equiv q$ i $r \equiv \mathrm{refl}_p$. Però llavors podem definir $\mathtt{trans}^2_E(\mathrm{refl}_p, u) :\equiv \mathrm{refl}_{\mathtt{trans}_E(p,u)}$. □

1.2.4 Homotopies i equivalències

Fins ara, hem presentat com les igualtats entre elements a la teoria homotòpica de tipus ens indueixen una estructura sobre cada tipus. Ara passem a preguntar-nos com definim les igualtats o equivalències entre funcions i tipus.

Per exemple, donades dues funcions $f, g : A \to B$, podríem considerar la igualtat punt a punt entre aquestes, és a dir, el tipus $\prod_{x:A}(f(x) = g(x))$. Que relació té aquesta igualtat amb la convencional $f = g$? Aquestes preguntes ens portaran cap a l'únic axioma utilitzat a la teoria homotòpica de tipus: *l'axioma d'univalència*. Comencem per posar noms a cada concepte i estudiar algunes propietats.

Definició 1.17. Siguin $E : A \to \mathrm{U}$ una família i $f, g : \prod_{x:A} E(x)$ dues funcions dependents. Una *homotopia* de f a g és una funció dependent del tipus

$$f \sim g :\equiv \prod_{x:A}(f(x) = g(x)).$$

Amb aquesta definició i les propietats vistes a la secció anterior podem demostrar la primera relació entre les homotopies i les igualtats de funcions.

Lema 1.18. *Sigui $H : f \sim g$ una homotopia entre les funcions $f, g : A \to B$ i $p : x = y$. Llavors tenim*

$$H(x) \cdot \mathtt{ap}_g(p) = \mathtt{ap}_f(p) \cdot H(y).$$

Anomenarem a aquesta propietat *naturalitat entre homotopies i funcions*.

Demostració. Per inducció a p, suposem $p \equiv \mathrm{refl}_x$ i $x \equiv y$. En aquest cas tenim

$$\begin{aligned}\Big(H(x) \cdot \mathtt{ap}_g(\mathrm{refl}_x) = \mathtt{ap}_f(\mathrm{refl}_x) \cdot H(x)\Big) &\equiv \Big(H(x) \cdot \mathrm{refl}_{g(x)} = \mathrm{refl}_{f(x)} \cdot H(x)\Big) \\ &\equiv (H(x) = H(x))\end{aligned}$$

que és cert ja que sempre existeix l'element $\mathrm{refl}_{H(x)} : H(x) = H(x)$. □

Aquesta propietat ens caracteritza a H com a transformació natural entre les funcions $f, g : A \to B$ considerades functors dels ∞-grupoides assignats a A i B. Per tant, les homotopies correspondran a transformacions naturals entre ∞-grupoides.

Finalment, existeix una funció que ens permet extreure una homotopia a partir d'una igualtat:

Lema 1.19. *Siguin A i B dos tipus qualssevol i $f, g : A \to B$ dues funcions. Existeix la funció*

$$\mathtt{hap} : (f = g) \to (f \sim g). \tag{1.1}$$

Demostració. Apliquem inducció de camins a p, suposant $f \equiv g$ i $p \equiv \mathrm{refl}_f$. En aquest cas tenim $(f \sim f) \equiv \prod_{x:A} f(x) = f(x)$, i per tant existeix

$$(x \mapsto \mathrm{refl}_f(x)) : f \sim f.$$

Finalment, per inducció de camins amb el cas base donat, obtenim $\mathtt{hap}$. □

Ara que ja hem vist les principals propietats de les homotopies, podem definir el concepte d'equivalència entre tipus a partir d'aquestes.

Definició 1.20. Donada una funció $f : A \to B$, una *quasi inversa de f* és un producte dependent

$$(g, \alpha, \beta) : \sum_{g:B\to A} (f \circ g \sim \mathtt{id}_B) \times (g \circ f \sim \mathtt{id}_A).$$

Definició 1.21. Donada una funció $f : A \to B$, direm que f és una *equivalència* si compleix

$$\mathtt{isEquiv}(f) : \left(\sum_{g:B\to A} (f \circ g \sim \mathtt{id}_B) \right) \times \left(\sum_{h:B\to A} (h \circ f \sim \mathtt{id}_A) \right).$$

Definició 1.22. Siguin A i B dos tipus qualsevol. Definim el *tipus de les equivalències* entre A i B com

$$(A \simeq B) : \sum_{f:A\to B} \mathtt{isEquiv}(f).$$

Anem a veure una equivalència que després necessitarem, entre el producte d'igualtats i igualtats de productes:

Teorema 1.23. *Siguin $E : A \to \mathrm{U}$ una família sobre un tipus A qualsevol, i $w_0, w_1 : \sum_{x:A} E(x)$ dos parells dependents. Llavors existeix una equivalència*

$$(w_0 = w_1) \simeq \sum_{p:\mathtt{pr}_1(w_0)=\mathtt{pr}_1(w_1)} \mathtt{trans}_E(p, \mathtt{pr}_2(w_0)) = \mathtt{pr}_2(w_1).$$

A més, anomenarem

$$\mathtt{dproj} : (w_0 = w_1) \to \left(\sum_{p:\mathtt{pr}_1(w_0)=\mathtt{pr}_1(w_1)} \mathtt{trans}_E(p, \mathtt{pr}_2(w_0)) = \mathtt{pr}_2(w_1) \right)$$

$$\mathtt{dpair} : \left(\sum_{p:\mathtt{pr}_1(w_0)=\mathtt{pr}_1(w_1)} \mathtt{trans}_E(p, \mathtt{pr}_2(w_0)) = \mathtt{pr}_2(w_1) \right) \to (w_0 = w_1)$$

a les funcions que formen aquesta equivalència.

Demostració. Es pot veure a [Uni13, Teorema 2.7.2., pàg. 101]. □

Corol·lari 1.24. *Siguin A, B dos tipus qualssevol; $a_0, a_1 : A$; $b_0, b_1 : B$; $p : a_0 = a_1$; $q : b_0 = b_1$ i $\alpha : (a_0, b_0) = (a_1, b_1)$. Llavors existeix una equivalència*

$$((a_0, b_0) = (a_1, b_1)) \simeq ((a_0 = a_1) \times (b_0 = b_1)).$$

A més, anomenarem

$$\mathtt{proj} : ((a_0, b_0) = (a_1, b_1)) \to ((a_0 = a_1) \times (b_0 = b_1))$$
$$\mathtt{pair} : ((a_0 = a_1) \times (b_0 = b_1)) \to ((a_0, b_0) = (a_1, b_1))$$

a les funcions que formen aquesta equivalència.

Demostració. Apliquem el teorema anterior amb la família constant $E : A \to \mathrm{U}$ tal que $E(a) :\equiv B$ per a tot $a : A$. □

Utilitzant aquest teorema, podem definir l'elevació de 2-igualtats, per a la qual necessitàvem la funció $\mathtt{dpair}$.

Lema 1.25. *Siguin $E : A \to \mathrm{U}$, $x, y : A$, $p, q : x = y$ i $r : p = q$. Per a tot $u : E(x)$, existeix la identificació següent:*

$$\mathtt{lift}^2_E(u, r) : \mathtt{lift}(u, p) \cdot \mathcal{A}(x, u, r) = \mathtt{lift}(u, q),$$

on $\mathcal{A}(a, b, s) :\equiv \mathtt{dpair}((\mathrm{refl}_a, \mathtt{transConnect}^2_E(s, b)))$.

Anomenarem a la funció definida *elevació de la* 2*-igualtat r a u*.

Demostració. Apliquem inducció de camins sobre r, és a dir, suposem $p \equiv q$ i $r \equiv \mathrm{refl}_p$. Obtenim

$$\mathcal{A}(x, u, \mathrm{refl}_p) \equiv \mathtt{dpair}((\mathrm{refl}_a, \mathtt{transConnect}^2_E(\mathrm{refl}_p, b))) \equiv \mathtt{dpair}((\mathrm{refl}_a, \mathrm{refl}_{\mathtt{trans}_E(p,u)}))$$

i per tant $\mathtt{lift}^2_E(u, \mathrm{refl}_p) :\equiv \mathrm{refl}_{\mathtt{lift}(u,p)}$. □

Arribats a aquest punt, tal com ens ha passat amb les homotopies i les igualtats aplicades a funcions, voldrem trobar quina relació hi ha entre les equivalències i les igualtats de tipus. Comencem per veure que existeix una funció que, donada una igualtat de tipus, ens retorna una equivalència:

Lema 1.26. *Siguin A i B dos tipus arbitraris. Existeix la funció*

$$\begin{aligned} \mathtt{idtoeqv} : (A =_{\mathrm{U}} B) &\to (A \simeq B) \\ p &\mapsto (\mathtt{trans}_{\mathtt{id}_{\mathrm{U}}}(p), P) \end{aligned} \tag{1.2}$$

on P és una demostració del fet que $\mathtt{trans}_{\mathtt{id}_{\mathrm{U}}}(p)$ és una equivalència entre A i B.

Demostració. Donada una igualtat $p : A =_{\mathrm{U}} B$, sabem que existeix la funció $f(p) :\equiv \mathtt{trans}_{\mathtt{id}_{\mathrm{U}}}(p)$, que és el transport de p a traves de la funció $\mathtt{id}_{\mathrm{U}} : \mathrm{U} \to \mathrm{U}$ amb equació $\mathtt{id}_{\mathrm{U}}(A) :\equiv A$ per a tot $A : \mathrm{U}$.

Volem trobar $P : \mathtt{isEquiv}(f(p))$. Per inducció de camins, es suficient considerar el cas $A \equiv B$ i $p \equiv \mathrm{refl}_A$. Llavors,

$$f(\mathrm{refl}_A) \equiv \mathtt{trans}_{\mathtt{id}_{\mathrm{U}}}(\mathrm{refl}_A) \equiv \mathtt{id}_{\mathtt{id}_{\mathrm{U}}(A)} \equiv \mathtt{id}_A$$

i en aquest cas $f(\mathrm{refl}_A)$ és una equivalència, ja que trivialment $\mathtt{id}_A$ és la seva pròpia quasi inversa. Finalment, obtenim la demostració $P : \mathtt{isEquiv}(f(p))$ com a resultat de la inducció de camins sobre el cas reflexiu anterior. □

Llavors, volem veure que aquesta funció és en realitat una equivalència. La teoria de tipus que hem definit fins ara és insuficient per a demostrar-ho. Per tant, ho imposarem mitjançant l'axioma d'univalència de Voevodsky.

Axioma 1.27 (Univalència)**.** *Siguin $A, B : \mathrm{U}$ tipus qualssevol. La funció (1.2) és una equivalència, és a dir,*

$$(A =_{\mathrm{U}} B) \simeq (A \simeq B).$$

Anomenarem `univalence` *a la quasi inversa de* `idtoeqv`*.*

Utilitzant l'axioma d'univalència, podem demostrar una relació més forta entre igualtats de funcions i homotopies. Concretament, podrem definir la inversa de la funció `hap` i veure que aquestes formen una equivalència.

Teorema 1.28 (Extensibilitat de funcions)**.** *Siguin A i B dos tipus arbitraris i $f, g : A \to B$ dues funcions. La funció (1.1) és una equivalència, amb quasi inversa*

$$\mathtt{funext} : (f \sim g) \to (f = g).$$

Demostració. Vegeu [Uni13, Teorema 4.9.5., pàg. 179]. □

1.2.5 Proposicions simples i conjunts

Fins ara hem vist que els tipus es comporten com a ∞-grupoides. Però podem trobar certes subclasses de tipus que no tenen cap informació rellevant a partir d'un cert nivell de la successió d'igualtats com a ∞-grupoides.

Obtindrem aquestes subclasses a partir d'exigir que un nivell de la successió de tipus sigui discret. Comencem demanant la condició més forta en quant a discretització: considerant tipus amb un sol element llevat d'igualtats. Aquesta propietat està relacionada amb els espais contràctils de topologia clàssica, on un espai topològic és contràctil quan podem deformar-lo contínuament en un punt.

Definició 1.29. Direm que un tipus A és *contràctil* si compleix

$$\mathtt{isContr}(A) :\equiv \sum_{a:A} \prod_{x:A} (a = x).$$

Als tipus contràctils també els anomenarem (-2)-*tipus*. Donat $P : \mathtt{isContr}(A)$, anomenarem a $\mathtt{pr}_1(P) \equiv a$ un *centre* de A.

La següent subclasse que podem definir és la de les proposicions simples. Aquests tipus tenen el mateix funcionament que les proposicions en teoria de conjunts. A diferència dels tipus contràctils, poden estar habitats o no. Però de manera similar als anteriors, si estan habitats, compleixen que els seus elements són iguals llevat d'igualtats d'ordre superior. Amb qualsevol tipus que sigui una proposició simple, saber quin element tenim exactament no ens aporta cap informació, ja que tots són equivalents.

Definició 1.30. Direm que un tipus A és una *proposició simple* si compleix

$$\mathtt{isProp}(A) :\equiv \prod_{x,y:A} (x = y).$$

També ho anomenarem (-1)-*tipus*.

La lògica generada per la interpretació de proposicions com a tipus té algunes propietats que no encaixen amb la que teníem a teoria de conjunts. Per a obtenir una lògica similar, podríem restringir-nos a interpretar com a proposicions només els tipus que siguin proposicions simples. Però llavors, podrem utilitzar els mateixos tipus com a connectives lògiques que els vistos a la interpretació anterior? La resposta és sí, llevat del producte dependent, que necessita una petita modificació per a ser una proposició simple. No veurem aquests resultats en aquest treball; per a qualsevol consulta es poden trobar a [Uni13].

Finalment, podem definir que un tipus sigui un conjunt. Utilitzarem la següent discretització en aquesta escala. En comptes de discretitzar elements, ho farem amb igualtats entre elements. Per tant, definirem un conjunt com un tipus on totes les igualtats entre dos elements són iguals, llevat d'igualtats d'ordre superior. Aquest concepte correspon a espais topològics discrets, on no trobem connexions no trivials entre punts.

Definició 1.31. Direm que un tipus A és un *conjunt* si compleix

$$\mathtt{isSet}(A) :\equiv \prod_{x,y:A} \prod_{p,q:x=y} (p = q).$$

També anomenarem 0-*tipus* als conjunts.

En l'últim capítol presentarem una generalització d'aquests conceptes, els *n-tipus*, juntament amb el seu anàleg en homotopia clàssica.

Capítol 2

Tipus inductius d'ordre superior

Tal com vam plantejar en els objectius d'aquest treball, volem ser capaços de definir objectes topològics clàssics com l'ampolla de Klein. Ara que ja hem descrit la teoria homotòpica de tipus, la següent qüestió a plantejar-nos és: com definim anàlegs d'objectes topològics?

En aquest capítol, presentem un dels mètodes més comuns en la definició de nous tipus. Després, veurem la seva generalització basada en la teoria homotòpica de tipus.

Per acabar, aprofitarem aquestes noves eines per definir alguns conceptes topològics com el tor o la circumferència, i veurem com demostrar propietats basades en aquests.

2.1 Tipus inductius

Com hem vist al capítol anterior, la definició de nous tipus pot ser feixuga, ja que hem d'especificar moltes regles. Però en alguns casos podem "generar" nous tipus a partir d'uns "constructors". En aquesta secció presentarem un mètode de generació de nous tipus que anomenarem *tipus inductius*.

Per a poder parlar de tipus inductius, necessitem formalitzar certs conceptes. Començarem per definir el concepte de "constructor". Si A és un tipus qualsevol, un *constructor* de A és un element de A o una funció estrictament positiva amb codomini A. Anomenarem *element constructor* i *funció constructora* respectivament als dos possibles tipus de constructors. Observem que, en el cas d'una funció constructora, aquesta sempre ens donarà elements de A a l'avaluar-la, per la condició que el codomini sigui A.

A més, necessitem formalitzar el concepte de "generar". Donat un tipus A qualsevol, direm que A està *generat lliurement* per una llista de constructors si cada element de A és igual per definició a un element constructor o a la imatge d'una funció constructora avaluada.

Utilitzant tots dos conceptes, definim *tipus inductiu* com un tipus generat lliurement per una llista de constructors. Per tant, per definir un nou tipus inductiu hem de donar una llista de constructors.

Observem que aquest mètode de definició de tipus ens exposa directament les regles de formació i introducció dels tipus que definim. A continuació veiem alguns exemples per familiaritzar-nos amb el concepte.

Comencem pels tipus inductius més trivials, que són els que només tenen elements constructors. Per exemple, considerem el tipus **1**, que com hem definit al capítol anterior, és el tipus d'un sol element $\star$. Aquest tipus està generat lliurement pel següent constructor:

$$\star : \mathbf{1}.$$

Un altre exemple senzill és el del tipus de dos elements **2**, generat pels constructors

$$0_{\mathbf{2}} : \mathbf{2},$$
$$1_{\mathbf{2}} : \mathbf{2}.$$

Observem que, tal com hem definit la generació lliure, els tipus creats utilitzant exclusivament elements constructors no podran tenir més elements que els que definim, com veiem en els dos casos anteriors.

Passem a presentar un exemple més interessant i que encara no havíem definit: els nombres naturals. La definició dels nombres naturals com a tipus inductius segueix un patró molt similar als axiomes de Peano. Definim el tipus dels *nombres naturals* $\mathbb{N}$ amb dos constructors:

$$0 : \mathbb{N},$$
$$\mathrm{succ} : \mathbb{N} \to \mathbb{N}.$$

Aquest és el primer exemple que veiem d'un tipus inductiu amb funcions constructores, i a més és un tipus inductiu recursiu, és a dir, alguna de les seves funcions accepta com a arguments elements del mateix tipus que definim. Com que $\mathbb{N}$ és un tipus inductiu, qualsevol element seu serà 0 o la imatge d'avaluar la funció succ en un element de $\mathbb{N}$. Per tant, com que només hi ha un cas base i una funció constructora recursiva, els elements de $\mathbb{N}$ seran generats formant una successió. Anomenarem els elements de la successió amb els seus noms habituals:

$$0,\ 1 :\equiv \mathrm{succ}(0),\ 2 :\equiv \mathrm{succ}(\mathrm{succ}(0)),\ 3 :\equiv \mathrm{succ}(\mathrm{succ}(\mathrm{succ}(0))),\ \ldots$$

Per veure com extreure els principis de recursió i inducció de tipus inductius, ens centrarem en els tipus **2** i $\mathbb{N}$, ja que dels exemples anteriors són els que no hem descrit en el primer capítol.

Comencem pel principi de recursió de **2**. Per definir una funció $f : \mathbf{2} \to C$ amb codomini un tipus qualsevol C necessitem escollir dos elements c_0 i c_1 de C. Llavors, la funció es defineix de manera que compleixi $f(0_{\mathbf{2}}) :\equiv c_0$ i $f(1_{\mathbf{2}}) :\equiv c_1$.

Pel que fa al principi d'inducció, donada una família $E : \mathbf{2} \to \mathrm{U}$, per a definir una funció dependent $f : \prod_{x:\mathbf{2}} E(x)$ necessitem escollir dos elements $e_0 : E(0_{\mathbf{2}})$ i $e_1 : E(1_{\mathbf{2}})$. Llavors, queda definida la funció f tal que $f(0_{\mathbf{2}}) :\equiv e_0$ i $f(1_{\mathbf{2}}) :\equiv e_1$.

Aplicant el principi anterior si E és una proposició, obtenim el clàssic raonament per casos, on, donada una proposició amb dos casos possibles, podem demostrar-la donant una demostració per a cadascun dels casos.

Ara que ja hem vist com deduir els principis d'eliminació en un tipus inductiu simple, vegem-ho en un cas amb funcions constructores com és el cas de $\mathbb{N}$. El principi de recursió per a $\mathbb{N}$ coincideix amb el concepte clàssic de successió de nombres naturals, que es defineixen a partir d'una recursió. Aquest fet no és a l'atzar: el nom de principi de recursió ve donat a partir d'aquest cas amb els nombres naturals.

Enunciem el principi de recursió. Donat un tipus qualsevol C, volem definir una funció $f : \mathbb{N} \to C$. Per fer-ho, necessitem escollir un element $c_0 : C$ i una funció $c_s : \mathbb{N} \to C \to C$. Llavors, existeix una funció $f : \mathbb{N} \to C$ tal que per a tot $n : \mathbb{N}$

$$\begin{aligned} f(0) &:\equiv c_0 \\ f(\mathrm{succ}(n)) &:\equiv c_s(n, f(n)). \end{aligned}$$

En el cas del principi d'inducció, considerarem una família qualsevol $C : \mathbb{N} \to \mathrm{U}$. Si escollim un element $c_0 : C(0)$ i una funció dependent $c_s : \prod_{n:\mathbb{N}} C(n) \to C(\mathrm{succ}(n))$, llavors existeix $f : \prod_{n:\mathbb{N}} C(n)$ tal que

$$\begin{aligned} f(0) &:\equiv c_0 \\ f(\mathrm{succ}(n)) &:\equiv c_s(n, f(n)). \end{aligned}$$

El principi d'inducció dels nombres naturals és equivalent a la demostració per inducció. L'element i la funció que hem d'escollir per a utilitzar el principi són el cas base i la demostració del cas $n + 1$ a partir del cas n.

En general, gràcies a la condició que les funcions constructores han de ser estrictament positives, existeix un algoritme que ens permet deduir els principis de recursió i d'inducció a partir dels constructors. Aquest algoritme no és simple: en el llibre base d'aquesta teoria [Uni13] no el donen explícitament perquè justifiquen que és massa tècnic. En canvi, fomenten l'aprenentatge intuïtiu a partir de molts exemples explicats detalladament. En aquest treball hem seguit el mateix procediment.

2.2 Tipus inductius d'ordre superior

Una de les limitacions dels tipus inductius és que només ens permeten generar elements del mateix tipus que definim. En el cas de la teoria homotòpica de tipus, podríem voler definir un tipus especificant no només els seus elements, sinó els elements de les identificacions que formen l'estructura de ∞-grupoide.

Aquesta generalització rep el nom de *tipus inductius d'ordre superior*, i és un dels pilars bàsics de la teoria homotòpica de tipus. La definició d'un tipus inductiu d'ordre superior serà igual a la d'un tipus inductiu ordinari, amb l'excepció que es permeten constructors que generin elements en els tipus de les igualtats al damunt del que construïm. A més, els constructors passen a ser una llista ordenada i poden utilitzar elements definits per altres constructors.

Com en el cas dels tipus inductius, posarem la restricció que les funcions constructores hauran de ser estrictament positives. A més, necessitarem afegir una nova condició. Per a cadascun dels costats de cada igualtat, aquesta només pot tenir un element constructor o igualtat constructora definida anteriorment, o una expressió formada per una combinació de construccions prèvies aplicades a operacions que formin una transformació natural respecte a les funcions constructores. Aquesta condició és molt complicada de justificar, i el llibre principal del camp [Uni13] diu que queda fora del seu abast.

A la pràctica, sabem que la concatenació de camins és preservada per totes les funcions de la teoria homotòpica de tipus. Per tant, podrem utilitzar-la. Aquesta és l'única operació que necessitarem aplicar als costats de les igualtats.

Per acabar d'assimilar el concepte, vegem un exemple amb la circumferència. Podem definir la circumferència com a tipus inductiu d'ordre superior de la manera següent:

Definició 2.1. La *circumferència* $\mathbb{S}^1$ és un tipus inductiu d'ordre superior generat pels constructors:

$$\begin{aligned} base &: \mathbb{S}^1, \\ loop &: base =_{\mathbb{S}^1} base. \end{aligned}$$

Interpretem aquesta definició. Si imaginem les igualtats com a camins topològics, estem definint la circumferència mitjançant un punt base que pertany a aquesta, i un camí que pertany a l'espai de llaços sobre aquest punt base.

Continuant amb la descripció de $\mathbb{S}^1$, enunciem el seu principi de recursió. Per a definir una funció f de la circumferència a un tipus C qualsevol, escollim un element $c : C$ i un $p : c =_C c$. Llavors, es defineix la funció $f : \mathbb{S}^1 \to C$ de manera que

$$\begin{gathered} f(base) \equiv c \\ \alpha : \mathtt{ap}_f(loop) = p. \end{gathered}$$

Tal com fèiem amb els tipus inductius, definim la imatge de l'element constructor *base* de $\mathbb{S}^1$ mitjançant una igualtat per definició.

En canvi, no podem aplicar f directament a l'element constructor *loop* de $base =_{\mathbb{S}^1} base$, ja que f té domini $\mathbb{S}^1$. Llavors, aplicarem f a la igualtat *loop* utilitzant l'operació $\mathtt{ap}_f$. Recordem que aquesta operació existeix sempre per a tota funció f i igualtat d'elements del domini d'aquesta.

A més, en comptes d'igualar per definició $\mathtt{ap}_f(loop)$ a p, s'utilitza una igualtat proposicional. Aquest fet és degut a la formalització utilitzada per a demostrar la consistència dels tipus inductius d'ordre superior.

Amb el principi de recursió, podem començar a analitzar les conseqüències la definició donada per la circumferència. Com el tipus $base =_{\mathbb{S}^1} base$ és una igualtat, sempre té un element refl_{base} i una operació de composició de camins. Per tant, és prudent comprovar que el llaç *loop* no és igual al camí trivial refl_{base}. Veiem que aquests dos camins són diferents:

Proposició 2.2. *Siguin base i loop els elements definits als constructors de* $\mathbb{S}^1$*. Llavors*

$$loop \neq \mathrm{refl}_{base}.$$

Demostració. Per a demostrar una desigualtat, suposarem la igualtat i arribarem a contradicció. Suposem $loop = \mathrm{refl}_{base}$. Sigui A un tipus qualsevol, $x : A$ i un camí $p : x = x$. Llavors existeix una funció $f : \mathbb{S}^1 \to A$ definida per $f(base) :\equiv x$ i $\alpha : \mathtt{ap}_f(loop) := p$. Per tant, tenim

$$p = \mathtt{ap}_f(loop) = \mathtt{ap}_f(\mathrm{refl}_{base}) = \mathrm{refl}_x.$$

A partir d'aquesta equació deduïm que per a tot tipus A totes les identificacions sobre aquest són trivials. Aquest fet implica que tots els tipus són conjunts, però podem veure, per exemple, que l'univers mai no ho és.

Per demostrar-ho, necessitem trobar un conjunt A amb un camí $p : A = A$ que no sigui igual a refl_A. Escollim $A :\equiv \mathbf{2}$ i definim una funció $f : A \to A$ amb equacions $f(0_\mathbf{2}) :\equiv 1_\mathbf{2}$ i $f(1_\mathbf{2}) :\equiv 0_\mathbf{2}$.

Llavors, per a tot $x : A$ obtenim $H(x) : f(f(x)) = x$ per definició de f. Per tant, f és una equivalència i la seva pròpia quasi inversa. Aplicant l'axioma d'univalència obtenim $p :\equiv \mathtt{univalence}(H) : A = A$.

Per demostrar que p no és igual a refl_A, suposem que són iguals. Llavors, per l'axioma d'univalència f serà igual a la funció $\mathtt{id}_A$, que implicaria $1_\mathbf{2} = 0_\mathbf{2}$, i per tant, hem arribat a una contradicció.

Finalment, hem trobat un conjunt $\mathbf{2}$ amb un camí $\mathtt{univalence}(H) : A = A$ que no és igual a refl_A. Per tant, l'univers mai no serà un conjunt. Aquest fet contradiu que tots els tipus siguin conjunts, i obtenim que $loop \neq \mathsf{refl}_{base}$, com volíem veure. $\square$

Per tant, sabem que $loop$ és un element no trivial i existeixen els següents elements possiblement no trivials del tipus $base =_{\mathbb{S}^1} base$:

$$\ldots,\ loop^{-1} \cdot loop^{-1},\ loop^{-1},\ \mathsf{refl}_{base},\ loop,\ loop \cdot loop,\ \ldots$$

Evidentment, sabem per teoria d'homotopia que el conjunt de les classes d'homotopia de llaços sobre un punt de la circumferència admet una estructura de grup isomorfa a $\mathbb{Z}$. Utilitzant la definició de la circumferència com a tipus inductiu d'ordre superior, és possible demostrar que el grup fonamental d'aquesta és $\mathbb{Z}$. En aquest treball no enunciarem aquest resultat perquè necessitaríem nous conceptes com la definició de grup fonamental, d'espai de llaços i de truncament de tipus.

Una de les principals preocupacions en definir un nou tipus inductiu és si hem caracteritzat el tipus que volíem exactament. Aquest dubte es veu amplificat amb la complexitat dels tipus inductius d'ordre superior. En el cas de la circumferència, la demostració que acabem de mencionar del fet que el seu grup fonamental sigui $\mathbb{Z}$ és una evidència molt forta de què la definició construeix el que entenem per circumferència en teoria clàssica. En la resta de definicions de tipus inductius d'ordre superior, també intentarem buscar caracteritzacions que ens confirmin les propietats que esperem de les construccions.

Per acabar la descripció de la circumferència, necessitem donar el seu principi d'inducció. Aquest principi no serà tan fàcil com en el cas de la recursió. Suposem que volem definir una funció dependent $f : \prod_{x:\mathbb{S}^1} E(x)$ a partir d'una família $E : \mathbb{S}^1 \to \mathcal{U}$. Clarament, per a l'element $base$, voldrem elegir un element $b : E(base)$, i imposarem l'equació

$$f(base) :\equiv b.$$

De manera similar amb les funcions, les complicacions arriben amb la imatge de $loop$. L'aplicació de funcions dependents a igualtats serà definida mitjançant l'operador $\mathtt{apd}$. Per tant, una definició natural de la imatge de la igualtat $loop$ a través d'una funció dependent f seria l'equació

$$\mathtt{apd}_f(loop) := l,$$

on l és la imatge que elegim per a $loop$. Llavors ens trobem amb la pregunta: de quin tipus hem d'elegir l? Clarament, si ha de ser igual proposicionalment a $\mathtt{apd}_f(loop)$, hauran de compartir tipus. Com hem vist en el capítol anterior, el tipus de $\mathtt{apd}_f(loop)$ és

$$\mathtt{apd}_f(loop)) : (\mathtt{trans}_E(loop, f(base)) = f(base)) \equiv (\mathtt{trans}_E(loop, b) = b),$$

d'on deduïm que podrem escollir $l : \mathtt{trans}_E(loop, b) = b$.

Un cop hem deduït tots els detalls del principi, podem enunciar-lo íntegrament. Donats una família $E : \mathbb{S}^1 \to \mathrm{U}$ i

$$\begin{aligned} &b : E(base), \\ &l : \mathtt{trans}_E(loop, b) = b, \end{aligned}$$

existirà una funció dependent $f : \prod_{x:\mathbb{S}^1} E(x)$ tal que

$$\begin{aligned} f(base) &:\equiv b, \\ \alpha : \mathtt{apd}_f(loop) &:= l. \end{aligned}$$

El cas general dels tipus inductius d'ordre superior és un tema de recerca actual. Per tant, donar un algoritme per a deduir els principis de recursió o d'inducció queda totalment fora de l'abast d'aquest treball. Com en el cas dels tipus inductius, intentarem transmetre intuïtivament el procediment amb els exemples que desenvoluparem.

2.2.1 La definició del tor

En aquesta secció definirem el tor com a tipus inductiu d'ordre superior. La definició és semblant a la de la circumferència i es defineix a partir d'un resultat clàssic del tor: els llaços estrictament "latitudinals" i els estrictament "longitudinals" commuten.

Aquesta definició i els seus principis de recursió i inducció són els que presenta Sojakova en els seus articles [Soj14; Soj15]. En aquestes referències no es donaven tots els detalls que donarem a continuació, ja que era un article destinat a un públic ja format en teoria homotòpica de tipus. Intentarem especificar tots els passos necessaris per arribar a entendre les deduccions dels principis.

Definició 2.3. Un *tor* $\mathbb{T}^2$ és un tipus inductiu d'ordre superior generat pels constructors

$$\begin{aligned} &b : \mathbb{T}^2, \\ &p : b = b, \\ &q : b = b, \\ &t : p \cdot q = q \cdot p. \end{aligned}$$

Per deduir el principi de recursió trobarem més dificultat que en el cas de la circumferència, a causa del fet que un dels constructors és una 2-igualtat que utilitza composició d'igualtats. A més, per primer cop notarem els efectes negatius del fet que les identificacions entre identificacions són proposicionals i no igualtats per definicions. Aquests fets ens obligaran a utilitzar unes transformacions entre la igualtat que elegim i la que assignem a la imatge.

Considerem

$$\texttt{apComp} : \prod_{u,v:b=b} (\texttt{ap}_f(u \cdot v) = \texttt{ap}_f(u) \cdot \texttt{ap}_f(v)),$$

que és una instància de la propietat demostrada al lema 1.9.

Comencem a deduir el principi considerant un tipus C qualsevol. Volem definir una funció $f : \mathbb{T}^2 \to C$. De manera similar a la circumferència, triarem un punt $b' : C$ i dues igualtats $p', q' : b' = b'$. Assignarem les imatges dels constructors amb les equacions

$$\begin{aligned} f(b) &:\equiv b', \\ \beta : \texttt{ap}_f(p) &= p', \\ \gamma : \texttt{ap}_f(q) &= q'. \end{aligned}$$

Llavors, només ens queda saber quin tipus d'igualtats podem escollir com a imatge de t, i com escrivim la seva equació. Sabem que l'operació amb la qual aplicarem f a t serà

$$\texttt{ap}^2_f(t) : \texttt{ap}_f(p \cdot q) = \texttt{ap}_f(q \cdot p). \tag{2.1}$$

Anomenarem t' a la igualtat que elegirem com a imatge de t. Encara no sabem de quin tipus serà t', però, com que serà una 2-igualtat de C, haurà de ser una igualtat entre camins construïts a partir de p' i q'. El tipus de (2.1) encara no compleix els requisits de t', però podem transformar-lo perquè ho faci. Llavors tenim les igualtats següents:

$$\begin{aligned} &\texttt{apComp}(p,q) : \texttt{ap}_f(p \cdot q) = \texttt{ap}_f(p) \cdot \texttt{ap}_f(q) \qquad &&\texttt{apComp}(q,p) : \texttt{ap}_f(q \cdot p) = \texttt{ap}_f(q) \cdot \texttt{ap}_f(p) \\ &\alpha * \beta : \texttt{ap}_f(p) \cdot \texttt{ap}_f(q) = p' \cdot q' &&\beta * \alpha : \texttt{ap}_f(q) \cdot \texttt{ap}_f(p) = q' \cdot p' \end{aligned}$$

Observem que amb aquestes equacions hem obtingut una manera de transformar elements de la igualtat de la imatge de t a elements construïts amb p' i q'. Per tant, donat un element $t' : p' \cdot q' = q' \cdot p'$, podem definir la imatge de t mitjançant l'equació següent:

$$\texttt{ap}^2_f(t) :\equiv \texttt{apComp}(p,q) \cdot (\alpha * \beta) \cdot t' \cdot (\beta * \alpha)^{-1} \cdot (\texttt{apComp}(q,p))^{-1}.$$

A més, podem simplificar l'equació definint $S(u_1, u_2, \delta_1, \delta_2) :\equiv \texttt{apComp}(u_1, u_2) \cdot (\delta_1 * \delta_2)$. Substituint, obtenim $\texttt{ap}^2_f(t) :\equiv S(p,q,\alpha,\beta) \cdot t' \cdot (S(q,p,\beta,\alpha))^{-1}$.

Finalment, podem enunciar el principi de recursió complet. Considerem un tipus qualsevol C. Si escollim

$$\begin{aligned} &b' : C, \\ &p' : b' = b', \\ &q' : b' = b', \\ &t' : p' \cdot q' = q' \cdot p', \end{aligned}$$

llavors existirà una funció $f : \mathbb{T}^2 \to C$ definida per

$$\begin{aligned} f(b) &\equiv b', \\ \alpha : \texttt{ap}_f(p) &= p', \\ \beta : \texttt{ap}_f(q) &= q', \\ \Theta : \texttt{ap}^2_f(t) &= \mathcal{S}(p,q,\alpha,\beta) \cdot t' \cdot (\mathcal{S}(q,p,\beta,\alpha))^{-1}, \end{aligned}$$

on $\mathcal{S}(u_1, u_2, \delta_1, \delta_2) :\equiv \mathtt{apComp}(u_1, u_2) \cdot (\delta_1 * \delta_2)$.

Ara que ja hem deduït el principi de recursió, presentarem les eines que utilitzarem per al d'inducció.

Lema 2.4. *Suposem $x, y, z : A$; $E : A \to U$; $p : x = y$ i $q : y = z$. Llavors és certa la propietat*

$$\mathtt{transComp}(p, q) : \mathtt{trans}_E(p \cdot q, f(x)) = \mathtt{trans}_E(q, \mathtt{trans}_E(p, f(x))).$$

Demostració. Per doble inducció de camins, considerem el cas $x \equiv y \equiv z$ i $p \equiv q \equiv \mathrm{refl}_x$. En aquest cas tenim

$$\mathtt{trans}_E(\mathrm{refl}_x \cdot \mathrm{refl}_x, f(x)) \equiv \mathtt{trans}_E(\mathrm{refl}_x, f(x)) \equiv f(x)$$
$$\mathtt{trans}_E(\mathrm{refl}_x, \mathtt{trans}_E(\mathrm{refl}_x, f(x))) \equiv \mathtt{trans}_E(\mathrm{refl}_x, f(x)) \equiv f(x)$$

i podem escollir $\mathtt{transComp}(\mathrm{refl}_x, \mathrm{refl}_x) :\equiv \mathrm{refl}_{f(x)}$. ☐

Lema 2.5. *Sigui $E : A \to \mathrm{U}$; $f : \prod_{x:A} E(x)$; $x, y, z : A$; $p : x = y$ i $q : y = z$. Llavors, existeix la igualtat*

$$\mathtt{apdComp}_f(p, q) : \mathtt{apd}_f(p \cdot q) = \mathtt{transComp}_f(p, q) \cdot \mathtt{ap}_{\mathtt{trans}_E(q)}(\mathtt{apd}_f(p)) \cdot \mathtt{apd}_f(q).$$

Demostració. Per doble inducció de camins, considerem $x \equiv y \equiv z$ i $p \equiv q \equiv \mathrm{refl}_x$. Llavors

$$\mathtt{transComp}_f(\mathrm{refl}_x, \mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)},$$
$$\mathtt{ap}_{\mathtt{trans}_E(\mathrm{refl}_x)}(\mathtt{apd}_f(\mathrm{refl}_x)) \equiv \mathtt{ap}_{\mathtt{id}_{E(x)}}(\mathrm{refl}_{f(x)}) \equiv \mathrm{refl}_{f(x)},$$

i substituint en la igualtat obtenim

$$\mathtt{apd}_f(\mathrm{refl}_x \cdot \mathrm{refl}_x) \equiv \mathtt{apd}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)}$$
$$\mathtt{transComp}_f(\mathrm{refl}_x, \mathrm{refl}_x) \cdot \mathtt{ap}_{\mathtt{trans}_E(\mathrm{refl}_x)}(\mathtt{apd}_f(\mathrm{refl}_x)) \cdot \mathtt{apd}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)}$$

Així podem definir $\mathtt{apdComp}_f(\mathrm{refl}_x, \mathrm{refl}_x) :\equiv \mathrm{refl}_{\mathrm{refl}_{f(x)}}$. ☐

Lema 2.6. *Considerem $E : A \to \mathrm{U}$; $f : \prod_{x:A} E(x)$; $x, y : A$; $p, q \in x = y$ i $r : p = q$. Llavors es compleix*

$$\mathtt{elimTrans2}_f(r, p) : \mathtt{trans}^2_E(r, \mathtt{apd}_f(p)) = (\mathtt{ap}_{\mathtt{trans}_E(\cdot, f(x))}(r))^{-1} \cdot \mathtt{apd}_f(p).$$

Demostració. Per inducció de camins, suposem que $p \equiv q$ i $r \equiv \mathrm{refl}_p$. Obtenim

$$\mathtt{trans}^2_E(\mathrm{refl}_p, \mathtt{apd}_f(p)) \equiv \mathtt{id}_{\mathtt{trans}_E(p, f(x)) = f(y)}(\mathtt{apd}_f(p)) \equiv \mathtt{apd}_f(p)$$
$$(\mathtt{ap}_{\mathtt{trans}_E(\cdot, f(x))}(\mathrm{refl}_p))^{-1} \cdot \mathtt{apd}_f(p) \equiv (\mathrm{refl}_{\mathtt{trans}_E(p, f(x))})^{-1} \cdot \mathtt{apd}_f(p)$$

Tornem a aplicar inducció de camins però a p i és suficient suposar $x \equiv y$ i $p \equiv \mathrm{refl}_x$. Però en aquest cas tenim

$$\mathtt{apd}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)}$$
$$(\mathrm{refl}_{\mathtt{trans}_E(\mathrm{refl}_x, f(x))})^{-1} \cdot \mathtt{apd}_f(\mathrm{refl}_x) \equiv \mathrm{refl}_{f(x)}{}^{-1} \cdot \mathrm{refl}_{f(x)} \equiv \mathrm{refl}_{f(x)}$$

Per tant, podem definir el cas base com $\mathtt{elimTrans2}_f(\mathrm{refl}_r, \mathrm{refl}_p) :\equiv \mathrm{refl}_{\mathrm{refl}_{f(x)}}$. ☐

Comencem a deduir el principi d'inducció. Sigui $E : \mathbb{T}^2 \to \mathcal{U}$ una família qualsevol. Volem definir una funció dependent $f : \prod_{x:A} E(x)$. Escollirem i assignarem les imatges de b, p i q igual que amb la circumferència. És a dir, escollim

$$\begin{aligned} &b' : E(b), \\ &p' : \mathtt{trans}_E(p, b') = b', \\ &q' : \mathtt{trans}_E(q, b') = b', \end{aligned}$$

i els assignem de la manera següent:

$$\begin{aligned} f(b) &\equiv b', \\ \alpha : \mathtt{apd}_f(p) &= p', \\ \beta : \mathtt{apd}_f(q) &= q'. \end{aligned}$$

Llavors, només ens queda saber com escollir i assignar la imatge de t, que anomenarem t'. Sabem que

$$\mathtt{apd}^2_f(t) : \mathtt{trans}^2_E(r, \mathtt{apd}_f(p \cdot q)) = \mathtt{apd}_f(q \cdot p).$$

Però tal com ens passava amb el cas del principi de recursió del tor, com que t està formada a partir de concatenacions, no podem escollir una t' del mateix tipus de $\mathtt{apd}^2_f(t)$ i que depengui de p' i q'. Llavors, trobarem una transformació a partir d'igualtats fins a arribar a una expressió formada per p' i q'.

Definim les expressions

$$\begin{aligned} \mathcal{A}_f(t) &:\equiv (\mathtt{ap}_{\mathtt{trans}_E(\cdot, b')}(t))^{-1} \\ \mathcal{B}_f(u_1, u_2, v_1, v_2) &:\equiv \mathtt{transComp}_f(u_1, u_2) \cdot \mathtt{ap}_{\mathtt{trans}_E(u_2)}(v_1) \cdot v_2 \\ \widetilde{\mathcal{C}}_f(u_1, u_2, \alpha_1, \alpha_2) &:\equiv \mathrm{refl}_{\mathtt{transComp}_f(u_1, u_2)} * \mathtt{ap}_{\mathtt{ap}_{\mathtt{trans}_E(u_2)}}(\alpha_1) * \alpha_2 \\ \mathcal{C}_f(u_1, u_2, \alpha_1, \alpha_2) &:\equiv \mathtt{apdComp}_f(u_1, u_2) \cdot \widetilde{\mathcal{C}}_f(u_1, u_2, \alpha_1, \alpha_2) \end{aligned}$$

i mitjançant les expressions i els lemes anteriors podem veure que

$$\begin{aligned} \mathtt{elimTrans2}_f(t, p \cdot q) &: \mathtt{trans}^2_E(t, \mathtt{apd}_f(p \cdot q)) = \mathcal{A}_f(t) \cdot \mathtt{apd}_f(p \cdot q) \\ \mathtt{apdComp}_f(p, q) &: \mathtt{apd}_f(p \cdot q) = \mathcal{B}_f(p, q, \mathtt{apd}_f(p), \mathtt{apd}_f(q)) \\ \mathtt{apdComp}_f(q, p) &: \mathtt{apd}_f(q \cdot p) = \mathcal{B}_f(q, p, \mathtt{apd}_f(q), \mathtt{apd}_f(p)) \\ \widetilde{\mathcal{C}}_f(p, q, \alpha, \beta) &: \mathcal{B}_f(p, q, \mathtt{apd}_f(p), \mathtt{apd}_f(q)) = \mathcal{B}_f(p, q, p', q') \\ \widetilde{\mathcal{C}}_f(q, p, \beta, \alpha) &: \mathcal{B}_f(q, p, \mathtt{apd}_f(q), \mathtt{apd}_f(p)) = \mathcal{B}_f(q, p, q', p'). \end{aligned}$$

Llavors, si escollim

$$t' : \mathcal{A}_f(t) \cdot \mathcal{B}_f(p, q, p', q') = \mathcal{B}_f(q, p, q', p'),$$

podem escriure l'equació de t com

$$\mathtt{apd}^2_f(t) :\equiv \mathtt{elimTrans2}_f(t, p \cdot q) \cdot (\mathrm{refl}_{\mathcal{A}_f(t)} * \mathcal{C}_f(p, q, \alpha, \beta)) \cdot t' \cdot (\mathcal{C}_f(q, p, \beta, \alpha))^{-1}.$$

Ara que tenim tots els detalls, podem enunciar el principi d'inducció del tor. Sigui $E : \mathbb{T}^2 \to \mathrm{U}$ una família qualsevol i $f : \prod_{x:\mathbb{T}^2} E(x)$ la funció dependent que volem definir. Donats

$$\begin{aligned}
b' &: E(b),\\
p' &: \mathtt{trans}_E(p, b') = b',\\
q' &: \mathtt{trans}_E(q, b') = b',\\
t' &: \mathcal{A}_f(t) \cdot \mathcal{B}_f(p, q, p', q') = \mathcal{B}_f(q, p, q', p'),
\end{aligned}$$

existeix $f : \prod_{x:\mathbb{T}^2} E(x)$ definida per

$$\begin{aligned}
f(b) &\equiv b',\\
\alpha : \mathtt{apd}_f(p) &= p',\\
\beta : \mathtt{apd}_f(q) &= q',\\
\Theta : \ \mathtt{apd}^2_f(t) &= \mathtt{elimTrans2}_f(t, p \cdot q) \cdot (\mathrm{refl}_{\mathcal{A}_f(t)} * \mathcal{C}_f(p, q, \beta, \gamma)) \cdot t' \cdot (\mathcal{C}_f(q, p, \gamma, \beta))^{-1},
\end{aligned}$$

on

$$\begin{aligned}
\mathcal{A}_f(t) &:\equiv (\mathtt{ap}_{\mathtt{trans}_E(\cdot, b')}(t))^{-1},\\
\mathcal{B}_f(u_1, u_2, v_1, v_2) &:\equiv \mathtt{transComp}_f(u_1, u_2) \cdot \mathtt{ap}_{\mathtt{trans}_E(u_2)}(v_1) \cdot v_2,\\
\mathcal{C}_f(u_1, u_2, \alpha_1, \alpha_2) &:\equiv \mathtt{apdComp}_f(u_1, u_2) \cdot \Big(\mathrm{refl}_{\mathtt{transComp}_f(u_1, u_2)} * \mathtt{ap}_{\mathtt{ap}_{\mathtt{trans}_E(u_2)}}(\alpha_1)\Big).
\end{aligned}$$

2.2.2 Equivalència entre el tor i el producte de dues circumferències

Com hem esmentat en els apartats anteriors, en el cas de la circumferència hi ha evidència que la definició donada és correcta, ja que compleix les propietats que esperaríem, com que el seu grup fonamental és $\mathbb{Z}$. En el cas del tor, necessitaríem veure que compleix alguna propietat característica per a justificar que la definició sintètica que hem donat és consistent amb la clàssica.

Donat que confiem en la nostra definició de $\mathbb{S}^1$, una opció és demostrar l'equivalència entre el tor i el producte de dues circumferències, que és un resultat molt conegut de topologia. Ara presentarem la primera part de la demostració, que és l'equivalència lògica entre aquests dos objectes. Una equivalència lògica són dues funcions entre dos objectes, una en cada sentit.

Proposició 2.7. *$\mathbb{T}^2$ i $\mathbb{S}^1 \times \mathbb{S}^1$ són lògicament equivalents. És a dir,*

$$(\mathbb{T}^2 \to \mathbb{S}^1 \times \mathbb{S}^1) \times (\mathbb{S}^1 \times \mathbb{S}^1 \to \mathbb{T}^2).$$

Demostració. Començarem definint la funció $G : \mathbb{T}^2 \to \mathbb{S}^1 \times \mathbb{S}^1$. Pel principi de recursió del tor, escollim

$$\begin{aligned}
(base, base) &: \mathbb{S}^1 \times \mathbb{S}^1,\\
\mathtt{pair}(\mathrm{refl}_{base},\ loop) &: (base, base) = (base, base),\\
\mathtt{pair}(loop,\ \mathrm{refl}_{base}) &: (base, base) = (base, base),
\end{aligned}$$

que seran les imatges de b, p i q respectivament, mitjançant les equacions

$$G(b) \equiv (base, base),$$
$$\alpha_G : \mathtt{ap}_G(p) = \mathtt{pair}(\mathrm{refl}_{base},\ loop),$$
$$\beta_G : \mathtt{ap}_G(q) = \mathtt{pair}(loop,\ \mathrm{refl}_{base}).$$

Només ens queda escollir la imatge de t, la qual sabem que tindrà un tipus de la forma

$$\Phi : \prod_{\alpha : base = base} (\mathtt{pair}(\mathrm{refl}_{base},\ \alpha) \cdot \mathtt{pair}(\alpha,\ \mathrm{refl}_{base})) = (\mathtt{pair}(\alpha,\ \mathrm{refl}_{base}) \cdot \mathtt{pair}(\mathrm{refl}_{base},\ \alpha))$$

i per tant l'element que volem trobar seria $\Phi(loop)$.

Però si apliquem inducció de camins a Φ, suposem $\alpha \equiv \mathrm{refl}_{base}$. En aquest cas,

$$(\mathtt{pair}(\mathrm{refl}_{base},\ \mathrm{refl}_{base}) \cdot \mathtt{pair}(\mathrm{refl}_{base},\ \mathrm{refl}_{base})) \equiv (\mathtt{pair}(\mathrm{refl}_{base},\ \mathrm{refl}_{base}))$$

i podem escollir $\Phi(\mathrm{refl}_{base}) :\equiv \mathrm{refl}_{\mathtt{pair}(\mathrm{refl}_{base},\ \mathrm{refl}_{base})}$. Per tant, existeix $\Phi(loop)$ i queda definit per inducció de camins.

Finalment, podem escriure l'última equació del principi de recursió:

$$\Theta : \mathtt{ap}^2_f(t) = \mathcal{S}(p, q, \alpha_G, \beta_G) \cdot \Phi(loop) \cdot (\mathcal{S}(q, p, \beta_G, \alpha_G))^{-1},$$

on $\mathcal{S}(u_1, u_2, \delta_1, \delta_2) :\equiv \mathtt{apComp}(u_1, u_2) \cdot (\delta_1 * \delta_2)$.

Ja hem definit la primera funció, i per tant ens queda definir $F : \mathbb{S}^1 \times \mathbb{S}^1 \to \mathbb{T}^2$. En comptes de definir directament F, definirem $F^{\to} : \mathbb{S}^1 \to \mathbb{S}^1 \to \mathbb{T}^2$, i extreurem F a partir d'aquesta.

Aplicant el principi de recursió de la circumferència, definim $f : \mathbb{S}^1 \to \mathbb{T}^2$ mitjançant les equacions

$$f(base) \equiv b,$$
$$\alpha_f : \mathtt{ap}_f(loop) = p.$$

Per a definir $F^{\to}$, necessitarem donar un camí entre funcions com a imatge de $loop$. Per fer-ho, utilitzarem les construccions que hem donat a l'apartat 1.2.4. En particular, definirem una homotopia $H : f \sim f$, i després aplicarem `funext` per a obtenir la igualtat que volem. Comencem per definir $H : f \sim f \equiv \prod_{x:\mathbb{S}^1}(f(x) = f(x))$.

Per inducció de la circumferència, assignarem $H(base) :\equiv q$, i ens queda donar la imatge de $loop$. Sigui $E(z) :\equiv (f(z) = f(z))$. Per a definir-ho necessitem un parell de propietats que podem definir fàcilment per inducció de camins:

- Per a tot $u : a = b$, $v : b = d$, $w : a = c$, $z : c = d$, tenim

$$\mathcal{I} : (u \cdot v = w \cdot z) \to (u^{-1} \cdot w \cdot z = v).$$

- Per a tot $s : x = y$ i $u : f(x) = f(x)$ existeix

$$\mathcal{T}(s, u) : \mathtt{trans}_E(s, u) = \mathtt{ap}_f(s)^{-1} \cdot u \cdot \mathtt{ap}_f(s).$$

Finalment estem preparats per definir

$$r :\equiv \mathcal{T}(loop, q) \cdot \mathcal{I}\left((\alpha_f * \mathrm{refl}_q) \cdot t \cdot (\mathrm{refl}_q * \alpha_f{}^{-1})\right) : (\mathtt{trans}_E(loop, q) = q),$$

i la igualtat $\delta_H : \mathtt{apd}_H(loop) = r$.

Llavors, definim $F^{\rightarrow} : \mathbb{S}^1 \rightarrow \mathbb{S}^1 \rightarrow \mathbb{T}^2$ pel principi de recursió de la circumferència, mitjançant les equacions

$$F^{\rightarrow}(base) \equiv f,$$
$$\alpha_{F^{\rightarrow}} : \mathtt{ap}_{F^{\rightarrow}}(loop) = \mathtt{funext}(H).$$

Finalment, definim $F : \mathbb{S}^1 \times \mathbb{S}^1 \rightarrow \mathbb{T}^2$ com

$$\begin{aligned} F((base, base)) &\equiv F^{\rightarrow}(base, base) \equiv b, \\ \alpha_F : \mathtt{ap}_F(\mathtt{pair}(\mathrm{refl}_{base}, \, loop)) &= \mathtt{ap}_f(loop) = p, \\ \beta_F : \mathtt{ap}_F(\mathtt{pair}(loop, \, \mathrm{refl}_{base})) &= \mathtt{hap}(\mathtt{ap}_{F^{\rightarrow}}(loop), base) = q, \\ \gamma_F : \mathtt{ap}_F(\mathtt{pair}(loop, \, loop)) &= \mathtt{apd}_{\mathtt{hap}(\mathtt{ap}_{F^{\rightarrow}}(loop))}(loop) = r, \end{aligned}$$

on les primeres igualtats es poden demostrar per inducció de camins i les segones de β_F i γ_F venen donades per $\mathtt{ap}_{\mathtt{hap}}(\alpha_{F^{\rightarrow}}) : \mathtt{hap}(\mathtt{ap}_{F^{\rightarrow}}(loop)) = H$, ja que $\mathtt{hap}$ és la quasi inversa de $\mathtt{funext}$. □

La segona part de la demostració és la més extensa de l'article: ocupa unes 12 pàgines. Per tal de no excedir-nos amb l'extensió del treball, ens limitarem a citar-la. En aquesta part, Sojakova demostra que les funcions que hem definit en la proposició anterior formen una equivalència.

Proposició 2.8. *$\mathbb{T}^2$ i $\mathbb{S}^1 \times \mathbb{S}^1$ són equivalents. És a dir,*

$$\mathbb{T}^2 \simeq \mathbb{S}^1 \times \mathbb{S}^1.$$

Demostració. Es pot consultar [Soj15, Section 5]. □

Per tant, acabem de veure una caracterització del tor que avala la nostra definició. A més, demostrant l'equivalència lògica hem vist com definir funcions des del tor utilitzant el principi d'inducció.

Capítol 3

Recobridors

En aquest capítol presentarem els espais recobridors i els n-tipus a la teoria homotòpica de tipus i demostrarem les propietats que ens relacionen aquests dos conceptes. Tot seguit, definirem sintèticament l'ampolla de Klein en teoria homotòpica de tipus i presentarem els nostres avenços en relació al problema (I. 1).

3.1 El concepte de recobridor

A topologia es defineix un espai recobridor d'un espai topològic X com un un fibrat $f : C \to X$ amb fibra discreta. En aquesta secció es presenta la definició anàloga en teoria homotòpica de tipus, seguint els articles [HH18; Hou13]. Considerem Set el tipus de tots els conjunts.

Definició 3.1. Un *espai recobridor* d'un tipus A és una família de conjunts indexada per A, és a dir, una família $P : A \to \mathrm{Set}$.

Definició 3.2. Un *espai recobridor puntejat* és un espai recobridor P d'un tipus A on existeix un punt distingit a de A i un altre b de la fibra $P(a)$.

En aquestes definicions, l'espai total de P, és a dir, $\sum_{x:A} P(x)$, correspondria al conjunt C de la definició clàssica en topologia. Per a tot $a : A$, cada tipus $P(a)$ correspondrà a una fibra, i per tant a $f^{-1}(a)$ en la definició clàssica. En aquesta interpretació, l'enunciat que les fibres siguin discretes equival a la propietat que cada fibra sigui un conjunt en el sentit de la teoria homotòpica de tipus.

Per tant, podríem il·lustrar un recobridor a teoria homotòpica de tipus com:

$$\begin{aligned} f : (\textstyle\sum_{x:A} P(x)) &\to A \\ w &\mapsto \mathtt{pr}_1(w) \end{aligned}$$

Per als resultats de la secció següent sobre elevacions de recobridors, necessitem definir el concepte de n-tipus. Per tal de definir els n-tipus, necessitem una successió similar als nombres naturals però que comenci per -2:

Definició 3.3. $\mathbb{N}'$ tal que

$$\text{-2} : \mathbb{N}',$$
$$\text{succ}' : \mathbb{N}' \to \mathbb{N}'.$$

Per simplificar la notació anomenarem $n + 1 :\equiv \text{succ}'(n)$, per a tot $n : \mathbb{N}'$. Veurem que $\mathbb{N}'$ és un tipus equivalent als naturals però amb punt base a -2. D'aquesta manera, podrem utilitzar tots els resultats vists pels naturals, a partir de les funcions de l'equivalència.

Lema 3.4. $\mathbb{N} \simeq \mathbb{N}'$

Demostració. Definim les funcions $f : \mathbb{N} \to \mathbb{N}'$ i $g : \mathbb{N}' \to \mathbb{N}$ amb les equacions

$$f(0) :\equiv \text{-2}, \text{ i per a tot } n : \mathbb{N}, \; f(\text{succ}(n)) :\equiv \text{succ}'(f(n))$$
$$g(\text{-2}) :\equiv 0, \text{ i per a tot } n' : \mathbb{N}', \; g(\text{succ}'(n')) :\equiv \text{succ}(g(n'))$$

Per demostrar l'equivalència, veurem $\prod_{n:\mathbb{N}}(g(f(n)) = n)$ i $\prod_{n':\mathbb{N}'}(f(g(n')) = n')$:

$$g(f(0)) \equiv g(\text{-2}) \equiv 0$$
$$g(f(\text{succ}(n))) \equiv g(\text{succ}'(f(n))) \equiv \text{succ}(g(f(n)))$$

$$f(g(\text{-2})) \equiv f(0) \equiv \text{-2}$$
$$f(g(\text{succ}'(n'))) \equiv f(\text{succ}(g(n'))) \equiv \text{succ}'(f(g(n')))$$

□

Utilitzant aquesta nova eina, podem definir els n-tipus, que generalitzen els conceptes de la subsecció 1.2.5. Informalment, un n-tipus és un tipus on per a tot $m > n$, totes les m-igualtats són trivials. És a dir, un tipus sense informació més enllà del nivell n.

Definició 3.5. Per a tot $n : \mathbb{N}'$, direm que un tipus A és un *n-tipus* si compleix la proposició recursiva

$$\texttt{is-n-type}(A) :\equiv \begin{cases} \texttt{isContr}(A) & \text{si } n \equiv \text{-2} \\ \prod_{x,y:A} \texttt{is-m-type}(x =_A y) & \text{si } n \equiv m + 1 \end{cases}$$

Finalment, introduirem dues propietats necessàries per al següent apartat:

Lema 3.6. *Siguin $A, B : \mathrm{U}$. Si $A \simeq B$ i A és un n-tipus, llavors B també ho és.*

Demostració. Vegeu [Uni13, Corol·lari 7.1.5., pàg. 273]. □

Lema 3.7. *Considerem un tipus A qualsevol. Per a tot $n : \mathbb{N}'$, si A és un n-tipus, llavors també és un $(n + 1)$-tipus.*

Demostració. Procedim per inducció sobre n. Per a $n \equiv \text{-2}$, volem veure que un tipus contràctil té els espais d'igualtats contràctils. Sigui $a_c : A$ el centre de contracció de A i siguin $x, y : A$ dos punts qualssevol. Volem veure que $x = y$ és contràctil.

Com que A és contràctil, existeixen $p_x : a_c = x$ i $p_y : a_c = y$ que compleixen

$$p_x \cdot {p_y}^{\text{-1}} : x = y.$$

Escollim $p_x \cdot {p_y}^{-1}$ com a centre de contracció de $x = y$, i comprovem si ho és. Per a tot $p : x = y$, volem veure que $p = p_x \cdot {p_y}^{-1}$. Per inducció de camins, suposem que $x \equiv y$ i $p \equiv \mathsf{refl}_x$. En aquest cas, obtenim trivialment que $p_x \cdot {p_x}^{-1} \equiv \mathsf{refl}_x$.

Per tant, acabem de veure el cas base de la inducció. Ara, suposem que ho tenim demostrat fins a n, i veiem-ho per $n+1$. Necessitem provar que $x = y$ és un $(n+1)$-tipus, suposant que A ho és.

Com que A és un $(n+1)$-tipus, per a tot $x, y : A$, $x = y$ serà un n-tipus. Aplicant la hipòtesis d'inducció, per a tot $p, q : x = y$, $p = q$ serà un n-tipus també. Però aquesta propietat és equivalent al fet que $x = y$ és un $(n+1)$-tipus, com volíem veure. □

Per acabar els fonaments d'aquest capítol, necessitarem el concepte de 0-connexió. Si A és un tipus qualsevol, direm que A és 0-*connex* si per a cada dos elements $x, y : A$ existeix un camí $p : x = y$. Aquesta propietat és l'equivalent a l'arc-connexió a teoria d'homotopia clàssica.

3.2 Elevacions en recobridors

Sigui $f : C \to A$ un recobriment d'espais topològics, on A és arc-connex. Llavors

- Per a tot $n \geq 0$, si A és un n-tipus, aleshores C també ho és.
- Per a tot $n \geq 1$, si C és un n-tipus, aleshores A també ho és.

Cal remarcar que la segona afirmació és falsa per a $n = 0$, com ho evidencia el recobriment $f : \mathbb{R} \to \mathbb{S}^1$ donat per l'aplicació exponencial. En aquesta secció, demostrarem els dos resultats anàlegs als dos anteriors a teoria homotòpica de tipus, ja que no els hem trobat en cap de les fonts consultades. Per al primer resultat, presentarem un teorema del llibre [Uni13] i després enunciarem el resultat com a corol·lari.

Teorema 3.8. *Sigui $n : \mathbb{N}'$ amb $n \geq -2$ i $B : A \to \mathcal{U}$ una família indexada per un tipus A qualsevol. Si A és un n-tipus i per a tot $B(a)$ és un n-tipus, llavors $\sum_{x:A} B(x)$ també ho és.*

Demostració. Demostrarem aquesta propietat per inducció. Si $n \equiv -2$, escollim el centre de contracció de $\sum_{x:A} B(x)$ com el parell (a_c, b_c), on a_c és el centre de A i b_c és el de $B(a_c)$. Donat qualsevol element $(a, b) : \sum_{x:A} B(x)$, podem construir una igualtat $\sum_{x:A} B(x)$ a partir de $p_a : a_c = a$ i $q_b : b_c = b$, pel teorema 1.23. Per tant, $\sum_{x:A} B(x)$ és contràctil, tal com volíem veure.

Considerem cert el cas n i veiem el cas $n+1$. Suposem que A i $B(a)$ per a tot $a : A$ són $(n+1)$-tipus. Llavors, donats $(a_0, b_0), (a_1, b_1) : \sum_{x:A} B(x)$, volem veure que $(a_0, b_0) = (a_1, b_1)$ és un n-tipus. Pel teorema 1.23, tenim que

$$((a_0, b_0) = (a_1, b_1)) \simeq \left(\sum_{p:a_0 = a_1} \mathtt{trans}_B(p, b_0) = b_1 \right)$$

i aplicant el lema 3.6, sabem que és suficient veure que $\sum_{p:a_0=a_1} \mathtt{trans}_B(p, b_0) = b_1$ és un n-tipus. Però això és cert, ja que com que A i $B(a)$ per a tot $a : A$ són $(n+1)$-tipus, $a_0 = a_1$ i $\mathtt{trans}_B(p, b_0) = b_1$ seran n-tipus, i podem aplicar la hipòtesi d'inducció. □

Corol·lari 3.9. *Considerem un tipus qualsevol A i un espai recobridor B de A. Per a tota $n : \mathbb{N}'$ amb $n \geq 0$, si A és un n-tipus, llavors $\sum_{x:A} B(x)$ també ho és.*

Demostració. Per a tot $a : A$, tenim que $B(a)$ és un 0-tipus per definició d'espai recobridor. Llavors, aplicant el lema 3.7 obtenim que $B(a)$ és un n-tipus, per a tot $n \geq 0$. Per tant, aplicant el teorema anterior obtenim que $\sum_{x:A} B(x)$ és un n-tipus. □

Per a demostrar la segona proposició, suposarem que A és 0-connex. Es podria demostrar aquest mateix resultat en el cas que A no sigui 0-connex, fent-ho per a cada component arc-connexa de A.

Proposició 3.10. *Sigui A un tipus 0-connex i $B : A \to \mathrm{Set}$ un espai recobridor puntejat de A. Per a tot $n : \mathbb{N}'$ tal que $n \geq 1$, si $\sum_{x:A} B(x)$ és un n-tipus, llavors A també ho és.*

Demostració. Demostrarem la propietat per inducció.

Cas base: Considerem $n \equiv 1$, suposant que $\sum_{x:A} B(x)$ és un 1-tipus, és a dir, existeix

$$\alpha : \prod_{w_0,w_1:\sum_{x:A} B(x)} \ \prod_{q_0,q_1:w_0=w_1} \ \prod_{s_0,s_1:q_0=q_1} (s_0 = s_1).$$

Considerem el punt distingit $a_0 : A$ i la seva imatge $b_0 : B(a_0)$. Volem veure que A és un 1-tipus, és a dir

$$\Phi : \prod_{a:A} \ \prod_{p_0,p_1:a_0=a} \ \prod_{r_0,r_1:p_0=p_1} (r_0 = r_1).$$

Siguin $\widetilde{a}_0 :\equiv (a_0, b_0)$, $\widetilde{a}_{p_0} :\equiv (a, \mathtt{trans}_B(p_0, b_0))$ i $\widetilde{a}_{p_1} :\equiv (a, \mathtt{trans}_B(p_1, b_0))$. Per a tot $a : A$, considerem les igualtats $p_0, p_1 : a_0 = a$, que podem elevar per aconseguir

$$\widetilde{p}_0 :\equiv \mathtt{lift}_B(b_0, p_0) : \widetilde{a}_0 = \widetilde{a}_{p_0}$$
$$\widetilde{p}_1 :\equiv \mathtt{lift}_B(b_0, p_1) : \widetilde{a}_0 = \widetilde{a}_{p_1}$$

Llavors, considerem $r_0, r_1 : p_0 = p_1$. Podem elevar aquestes homotopies

$$\widetilde{r}_0 :\equiv \mathtt{lift}^2_B(b_0, r_0) : \widetilde{p}_0 \cdot \mathcal{A}(a, b_0, r_0) = \widetilde{p}_1$$
$$\widetilde{r}_1 :\equiv \mathtt{lift}^2_B(b_0, r_1) : \widetilde{p}_0 \cdot \mathcal{A}(a, b_0, r_1) = \widetilde{p}_1$$

on $\mathcal{A}(x, b, r) :\equiv \mathtt{dpair}((\mathrm{refl}_x, \mathtt{transConnect}^2_B(r, b)))$.

Però com que

$$\mathtt{transConnect}^2_B(r_0, b_0), \mathtt{transConnect}^2_B(r_1, b_0) : \mathtt{trans}_B(p_0, b_0) = \mathtt{trans}_B(p_1, b_0)$$

són igualtats a $B(a)$, que és un 0-tipus ja que B és un recobridor, existirà una 2-igualtat

$$\beta : \mathtt{transConnect}^2_B(r_0, b_0) = \mathtt{transConnect}^2_B(r_1, b_0).$$

Llavors, definim $\mathcal{D} :\equiv \mathrm{refl}_{\mathtt{lift}_B(b_0,p_0)} * \mathtt{dpair}(\mathrm{refl}_{\mathrm{refl}_a}, \beta^{-1})$ i obtenim

$$(\mathcal{D} \cdot \widetilde{r}_0),\ \widetilde{r}_1 : \widetilde{p}_0 \cdot \mathcal{A}(a, b_0, r_1) = \widetilde{p}_1$$

que són dues 2-igualtats del mateix tipus a $\sum_{x:A} B(x)$.

Apliquem la hipòtesi inicial que $\sum_{x:A} B(x)$ és un 1-tipus per obtenir

$$\widetilde{\varphi} :\equiv \Phi(\widetilde{a}_0,\ \widetilde{a}_{p_1},\ (\widetilde{p}_0 \cdot \mathcal{A}(a, b_0, r_1)),\ \widetilde{p}_1,\ (\mathcal{D} \cdot \widetilde{r}_0),\ \widetilde{r}_1).$$

Finalment, pel teorema 1.23, existirà una descomposició que podem extreure amb

$$\varphi :\equiv \mathtt{pr}_2(\mathtt{dproj}(\widetilde{\varphi})) : r_0 = r_1,$$

i llavors φ és la igualtat entre r_0 i r_1 que necessitàvem per a demostrar que A és un 1-tipus.

Pas inductiu: Ara suposem que tenim demostrat el cas n, i considerem el cas $n+1$. Suposem que $\sum_{x:A} B(x)$ és un $(n+1)$-tipus, és a dir, existeix

$$\alpha : \prod_{w_0, w_1 : \sum_{x:A} B(x)} \mathtt{is-n-type}(w_0 = w_1).$$

Considerem el punt distingit $a_0 : A$ i la seva imatge $b_0 : B(a_0)$. Per a tot $a : A$, com que A és 0-connex, existirà $p : a_0 = a$. Aleshores, també existirà

$$\mathtt{lift}_B(b_0, p) : (a_0, b_0) = (a, \mathtt{trans}_B(p, b_0))$$

i pel teorema 1.23, tenim

$$((a_0, b_0) = (a, \mathtt{trans}_B(p, b_0))) \simeq \left(\sum_{q : a_0 = a} \mathtt{trans}_B(p, b_0) = \mathtt{trans}_B(p, b_0) \right).$$

Per tant, $(a_0, b_0) = (a, \mathtt{trans}_B(p, b_0))$ serà un n-tipus, ja que $\sum_{x:A} B(x)$ és un $(n+1)$-tipus. A més, pel lema 3.6, la darrera expressió obtinguda per l'equivalència anterior també serà un n-tipus.

Llavors, per la hipòtesi d'inducció, si veiem que $\widetilde{B} : (a_0 = a) \to \mathrm{U}$ definida per $\widetilde{B}(p) :\equiv \mathtt{trans}_B(p, b_0) = \mathtt{trans}_B(p, b_0)$ és un espai recobridor puntejat de $a_0 = a$, obtindrem que $a_0 = a$ és un n-tipus, i finalment que A és un $(n+1)$-tipus.

Veiem que per a tot $p : a_0 = a$, $\widetilde{B}(p)$ és un conjunt. Clarament, com que $B(a)$ és un conjunt, les igualtats a $B(a)$, com $\widetilde{B}(p)$, seran (-1)-tipus. Pel lema 3.7, $\widetilde{B}(p)$ també seran 0-tipus, i per tant $\widetilde{B}$ és un espai recobridor. Com que A és 0-connex, podem escollir $p_0 : a_0 = a$ i a la seva imatge $\mathrm{refl}_{\mathtt{trans}_B(p_0, b_0)} : \mathtt{trans}_B(p_0, b_0) = \mathtt{trans}_B(p_0, b_0)$ per a veure que $\widetilde{B}$ és un recobridor puntejat. □

3.3 L'ampolla de Klein

Utilitzant un procés totalment anàleg al del tor, definirem l'ampolla de Klein:

Definició 3.11. Una *ampolla de Klein* $\mathbb{K}$ és un tipus inductiu d'ordre superior generat pels constructors

$$\begin{aligned} &b_K : \mathbb{K}, \\ &p_K : b_K = b_K, \\ &q_K : b_K = b_K, \\ &t_K : p_K \cdot q_K = q_K \cdot {p_K}^{-1}. \end{aligned}$$

Observem que la definició de l'ampolla de Klein només es diferencia de la del tor en la seva 2-igualtat. En quant als principis, ens centrarem en donar el de recursió, ja que és l'únic que necessitem per a definir l'espai recobridor. No deduirem el principi pas a pas, ja que és molt semblant al del tor, amb unes igualtats afegides per als inversos d'identificacions. Considerem

$$\texttt{apInv} : \prod_{p:b_K=b_K} (\texttt{ap}_f(p^{-1}) = \texttt{ap}_f(p)^{-1}),$$

que és un cas particular de la segona propietat demostrada al lema 1.9.

Enunciem el principi de recursió. Considerem un tipus qualsevol C. Si escollim

$$\begin{aligned} &b' : C, \\ &p' : b' = b', \\ &q' : b' = b', \\ &t' : p' \cdot q' = q' \cdot p'^{-1}, \end{aligned}$$

llavors existirà una funció $f : \mathbb{K} \to C$ definida per

$$\begin{aligned} f(b_K) &\equiv b', \\ \alpha : \texttt{ap}_f(p_K) &= p', \\ \beta : \texttt{ap}_f(q_K) &= q', \\ \Phi : \texttt{ap}^2_f(t_K) &= \mathcal{L} \cdot t' \cdot (\mathcal{R})^{-1}, \end{aligned}$$

on

$$\begin{aligned} \mathcal{L} &:\equiv r(p_K, q_K) \cdot (\alpha * \beta), \\ \mathcal{R} &:\equiv r(q_K, {p_K}^{-1}) \cdot (\beta * (\texttt{apInv}(p_K) \cdot \texttt{ap}_{\cdot^{-1}}(\alpha))). \end{aligned}$$

3.4 El tor com a recobridor de l'ampolla de Klein

Per a poder plantejar el nostre recobridor de l'ampolla de Klein necessitarem un concepte més. Ens cal definir el tipus $\mathbb{Z}/2\mathbb{Z}$, però aquest es defineix com a quocient de conjunts a partir de $\mathbb{Z}$, que es defineix com a quocient de conjunts a partir de $\mathbb{N}$. El quocient de conjunts a la teoria homotòpica de tipus es defineix com un tipus inductiu d'ordre superior. Per no excedir-nos amb la llargària del treball, hem preferit no introduir aquests conceptes.

Per a aquest apartat, només necessitem saber que els tipus $\mathbb{Z}$ i $\mathbb{Z}/2\mathbb{Z}$ es comporten de manera molt similar a com ho fan en teoria de conjunts. A més, com que es defineixen per quocients, sempre seran 0-tipus, és a dir, conjunts. Aquest fet ve donat per una de les funcions constructores dels quocients.

Amb tota aquesta informació, ja estem preparats per a descriure el recobridor de l'ampolla de Klein. Considerem l'equivalència $E : \mathbb{Z}/2\mathbb{Z} \simeq \mathbb{Z}/2\mathbb{Z}$ definida per la funció $f : \mathbb{Z}/2\mathbb{Z} \to \mathbb{Z}/2\mathbb{Z}$ tal que $f(0_2) :\equiv 1_2$ i $f(1_2) :\equiv 0_2$, ja que aquesta és la seva pròpia inversa. Anomenarem

$$\begin{aligned} idt &:\equiv \mathrm{refl}_{\mathbb{Z}/2\mathbb{Z}} : (\mathbb{Z}/2\mathbb{Z} = \mathbb{Z}/2\mathbb{Z}), \\ flip &:\equiv \texttt{univalence}(E) : (\mathbb{Z}/2\mathbb{Z} = \mathbb{Z}/2\mathbb{Z}). \end{aligned}$$

Llavors, definirem la família $P : \mathbb{K} \to \mathrm{Set}$ pel principi de recursió de l'ampolla de Klein. Si escollim

$$\begin{aligned} \mathbb{Z}/2\mathbb{Z} &: \mathrm{U}, \\ idt &: \mathbb{Z}/2\mathbb{Z} = \mathbb{Z}/2\mathbb{Z}, \\ flip &: \mathbb{Z}/2\mathbb{Z} = \mathbb{Z}/2\mathbb{Z}, \\ \mathrm{refl}_{flip} &: (idt \cdot flip = flip \cdot idt^{-1}) \equiv (flip = flip), \end{aligned}$$

aleshores existirà $P : \mathbb{K} \to \mathrm{Set}$ tal que

$$\begin{aligned} P(b_K) &:\equiv \mathbb{Z}/2\mathbb{Z}, \\ \mathtt{ap}_f(p_K) &= idt, \\ \mathtt{ap}_f(q_K) &= flip, \\ \mathtt{ap}^2_f(t_K) &= \mathcal{L} \cdot \mathrm{refl}_{flip} \cdot (\mathcal{R})^{-1}, \end{aligned}$$

on

$$\begin{aligned} \mathcal{L} &:\equiv r(p_K, q_K) \cdot (\alpha * \beta), \\ \mathcal{R} &:\equiv r(q_K, {p_K}^{-1}) \cdot (\beta * (s(p_K) \cdot \mathtt{ap}_{\text{-}^{-1}}(\alpha))). \end{aligned}$$

Observem que, com que $\mathbb{Z}/2\mathbb{Z}$ és un conjunt, clarament P serà un recobridor. A més, afegint els punts distingits $b_K : \mathbb{K}$ i $0_2 : \mathbb{Z}/2\mathbb{Z}$, també serà un recobridor puntejat.

Conjecturem que l'espai total de P és equivalent al tor. Aquest fet s'hauria de poder demostrar d'una manera similar a la demostració de l'equivalència entre el tor i el producte de dues circumferències. Els detalls poden ser extensos i els deixem per a una eventual continuació d'aquest treball.

Gràcies als resultats obtinguts en les seccions anteriors, el fet que el tor recobreixi l'ampolla de Klein tindrà una conseqüència més. Per l'article [LS13], sabem que $\mathbb{S}^1 \times \mathbb{S}^1$ és un 1-tipus, i pel capítol anterior, sabem que el tor és equivalent a $\mathbb{S}^1 \times \mathbb{S}^1$. Aleshores, pel lema 3.6, sabem que $\mathbb{T}^2$ és un 1-tipus. Per tant, si demostrem que l'espai total de P és equivalent a $\mathbb{T}^2$, podrem aplicar la proposició 3.10 i obtindrem que l'ampolla de Klein també és un 1-tipus.

Conclusions

Gràcies a la realització d'aquest treball, ens hem pogut introduir en el món de la teoria homotòpica de tipus i hem aconseguit afrontar la recerca al voltant d'un problema interessant en aquest camp. Motivats per aquest problema, hem demostrat alguns resultats sobre recobridors que no consten en cap de les fonts bibliogràfiques consultades.

La teoria homotòpica de tipus obre moltes noves portes. La continuació natural d'aquest treball seria acabar la demostració del fet que el tor recobreix l'ampolla de Klein, demostrant que l'espai total del recobridor que hem definit és equivalent al tor. Com ja hem comentat, sembla una demostració del mateix caire que la que escriu Sojakova a l'article [Soj15], i que s'allarga unes 11 pàgines.

D'altra banda, hauria de ser possible aconseguir una variant més forta de la proposició 3.10. En la nova versió, no demanaríem que A fos 0-connex, sinó que demostraríem el resultat per a cada component arc-connexa. Per tal de concretar aquesta extensió, necessitaríem definir el concepte de components arc-connexes d'un tipus.

Seria interessant aprofitar els assistents de demostració com Agda o Coq i formalitzar els nous resultats obtinguts, de manera similar a com ho fan els experts de l'àrea com Sojakova, Licata o Brunerie. Tenim exemples que podríem seguir, com el cas de la formalització [Lic15a] de l'article [LB15].

També podríem explorar el recobriment de l'espai projectiu real de dimensió 2 per l'esfera en teoria homotòpica de tipus. En aquesta línia hem trobat l'article [BR17], que descriu els espais projectius reals de dimensió arbitrària.

Tornant al problema (I. 1), podríem avançar en la proposta d'una definició d'orientabilitat. En homotopia clàssica, tota varietat connexa M admet un recobridor de dos fulls orientable $\hat{M} \to M$ anomenat *recobridor d'orientació*. La varietat $\hat{M}$ és connexa si i només si M no és orientable. Quan M és orientable, $\hat{M}$ consisteix en dues còpies homeomorfes de la mateixa M. Aquest fet podria ser la clau per tal de donar una definició útil d'orientabilitat de varietats en la teoria homotòpica de tipus, basada en el concepte de recobridor.

Bibliografia

[BR17] Ulrik Buchholtz i Egbert Rijke. "The real projective spaces in homotopy type theory". A: *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (2017), pàg. 1 - 8.

[HH18] Kuen-Bang Hou (Favonia) i Robert Harper. "Covering Spaces in Homotopy Type Theory". A: *22nd International Conference on Types for Proofs and Programs (TYPES 2016)*. Ed. de Silvia Ghilezan, Herman Geuvers i Jelena Ivetić. Vol. 97. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, 11:1 - 11:16. ISBN: 978-3-95977-065-1. DOI: `10.4230/LIPIcs.TYPES.2016.11`. URL: `http://drops.dagstuhl.de/opus/volltexte/2018/9851`.

[Hou13] Kuen-Bang Hou (Favonia). *Covering Spaces*. 2013. URL: `https://homotopytypetheory.org/2013/04/27/covering-spaces/` (cons. 13-01-2020).

[LB15] D. R. Licata i G. Brunerie. "A Cubical Approach to Synthetic Homotopy Theory". A: *2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science*. Jul. de 2015, pàg. 92 - 103. DOI: `10.1109/LICS.2015.19`.

[Lic15a] Dan Licata. *Agda repository: The torus is the product of two circles*. 2015. URL: `https://github.com/dlicata335/hott-agda/blob/master/homotopy/TS1S1.agda` (cons. 15-01-2020).

[Lic15b] Dan Licata. *The torus is the product of two circles, cubically*. 2015. URL: `https://homotopytypetheory.org/2015/01/20/ts1s1-cubically/` (cons. 15-01-2020).

[LS13] Daniel R. Licata i Michael Shulman. "Calculating the Fundamental Group of the Circle in Homotopy Type Theory". A: *2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science* (2013), pàg. 223 - 232.

[nLa19] nLab authors. *Open problems in homotopy type theory*. 2019. URL: `https://ncatlab.org/homotopytypetheory/revision/open+problems/47` (cons. 15-01-2020).

[Soj14] Kristina Sojakova. "The torus T^2 is equivalent to the product $S^1 \times S^1$ of two circles". A: (2014). URL: `https://ncatlab.org/homotopytypetheory/files/torus.pdf`.

[Soj15] Kristina Sojakova. "The equivalence of the torus and the product of two circles in homotopy type theory". A: *CoRR* abs/1510.03918 (2015). arXiv: `1510.03918`. URL: `http://arxiv.org/abs/1510.03918`.

[Uni13] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study: `https://homotopytypetheory.org/book`, 2013.