Topological Quantum Gates in Homotopy Type Theory

David Jaz Myers*   Hisham Sati*,†   Urs Schreiber*

Abstract

Despite the plausible necessity of topological protection for realizing scalable quantum computers, the conceptual under-pinnings of topological quantum logic gates had arguably remained shaky, both regarding their physical realization as well as their information-theoretic nature.

Building on recent results on defect branes in string/M-theory [SS22-Def] and on their holographically dual anyonic defects in condensed matter theory [SS22-Ord], here we explain (as announced in [SS22-TQC]) how the specification of realistic topological quantum gates, operating by anyon defect braiding in topologically ordered quantum materials, has a surprisingly slick formulation in parameterized point-set topology, which is so fundamental that it lends itself to certification in modern homotopically typed programming languages, such as cubical `Agda`.

We propose that this remarkable confluence of concepts may jointly kickstart the development of topological quantum programming proper as well as of real-world application of homotopy type theory, both of which have arguably been falling behind their high expectations; in any case, it provides a powerful paradigm for simulating and verifying topological quantum computing architectures with high-level certification languages aware of the actual physical principles of realistic topological quantum hardware.

In companion articles [SS23-QM][SS23-EoS] (announced in [Sc22b]), we explain how further passage to "dependent linear" homotopy types naturally extends this scheme to a full-blown quantum programming/certification language in which our topological quantum gates may be compiled to verified quantum circuits, complete with quantum measurement gates and classical control.

In Memoriam of Yuri Manin
who introduced quantum computation
as well as Gauss-Manin connections
unknowing of their close relationship.

Contents

* Mathematics, Division of Science; and
Center for Quantum and Topological Systems,
NYUAD Research Institute,
New York University Abu Dhabi, UAE.

†The Courant Institute for Mathematical Sciences, NYU, NY

CENTER FOR
QUANTUM &
TOPOLOGICAL
SYSTEMS

1 Introduction

The need for topology in quantum computation. While the hopes connected with the idea of quantum computation (Lit. 2.1) are hard to overstate, experts are well-aware (Lit. 2.5) that practically useful quantum computation beyond the presently existing NISQ machines (Lit. 2.3) will require the development of profound stabilization mechanisms to protect quantum data against decoherence. This might be achievable at the software level by implementing enough redundancy within existing quantum hardware paradigms ("quantum error correction") but more likely it will (in addition) require error protection right at the hardware level, utilizing quantum materials whose quantum states are stabilized by fundamental physical principles broadly known as *topological* (Lit. 2.12), specifically by *topological order* appearing in *topological phases* of quantum materials (Lit. 2.7).

The open problem of topological quantum gates. Inevitable as the path of *topological quantum computing* (Lit. 2.4) may thus be in the long run, its theoretical underpinnings had arguably remained shaky (cf. [Va21][SS22-Ord, p. 2], Lit. 2.9), despite considerable interest and in contrast to the impression one may glean from a cursory perusal of the literature. This might have in part contributed to the apparent failure of the only attempt to date at implementing topologically protected qbits in the laboratory (Lit. 2.8). Generally, the original and still most promising idea of topological quantum logic gates operating on topologically ordered ground states (Lit. 2.7) by *adiabatic* (Lit. 2.6) *braiding* (Lit. 2.20) of *anyonic* (Lit. 2.17) defect worldlines had shifted out of the community's focus: Experimentalists have been focusing on topological states which, even if detected, would be intrinsically immobile and hence un-braidable (Lit. 2.8); while theorists have been exploring anyonic braiding in a generality remote from considerations of physical realizability (cf. Lit. 2.9 and p. 4).

A solution gleaned from high energy physics... But in a recent re-analysis of defects in topological quantum materials [SS22-Ord] (Lit. 2.7) – following analogous ("dual") considerations for stable *defect branes* in string/M-theory [SS22-Def] (Lit. 2.11) – we found a detailed realistic model for adiabatic anyon braiding, showing how the established classification of topological phases of matter (Lit. 2.7) by *topological K-theory* (Lit. 2.16) extends to describe topologically ordered states supporting braid group statistics (Lit. 2.17). This description of anyon braiding turns out to flow naturally from just fundamental constructions in *parameterized homotopy theory* (discussed §4), revealing a deeper purely homotopy-theoretic nature (Lit. 2.12) of topological quantum gates than has been appreciated before.

...lending itself to certified quantum programming. This formulation of topological quantum gates in parameterized homotopy theory is noteworthy also in view of a remarkable modern development in certified ("typed") programming language theory (Lit. 2.26): where strict adherence to the principle of assigning *types* to all data, in particular also to certificates of identification of pairs of other data, leads to these types behaving just as the *homotopy types* of parameterized homotopy theory. In effect, the fundamental certification language now called *homotopy type theory* ("HoTT", Lit. 2.27) serves at once as a general-purpose programming language as well as a proof language for constructions in parameterized homotopy theory.

Claim and broad Outline. In summary, this suggests that *homotopy-typed programming languages* (Lit. 2.27, exposition in §5.1) naturally serve for encoding (simulating) and formally verifying realistic topological quantum logic gates (Lit. 2.4), providing a natural theoretical basis for simulation and certification of realistic topological quantum computing platforms. We had briefly announced this result in [SS22-TQC]; here we introduce and explain in detail:

> **in §4** the relevant parameterized point-set topology as understood from traditional algebraic topology literature (Lit. 2.12);
> **in §5** the relevant dependent homotopy type theory, as operational in the programming language Agda (Lit. 2.28);
> **in §6** the construction of the homotopy data structure (Def. 6.7) which encodes topological quantum gates: Theorem 6.8.

More technical outline. Before concluding, here to outline our construction/claim in a little more technical detail:

Our starting point is the following two facts which are separately "well-known" to their respective experts, but whose striking conjunction does not seem to have been appreciated before:

(1) [SS22-Def, §5]: Plausible future topological quantum computation hardware realizes (only) those anyon braid quantum gates (Lit. 2.20) which act on quantum states by monodromy braid representations (Lit. 2.21) of Knizhnik-Zamolodchikov (KZ) connections (Lit. 2.24) on "$\widehat{\mathfrak{su}}_2^k$-conformal blocks" from conformal field theory. At level $k = 2$ these include the popular Majorana/Ising anyons, and for $k = 3$ the universal "Fibonacci anyons" (Lit. 2.19).

The point is that anyon braiding is often broadly hypothesized (cf. [Va21]) to be described by any braid group representation or any unitary braided fusion category. However, plausible physical realizations are expected to be much more specifically given by modular tensor categories of affine $\mathfrak{su}_2$- representations, these being quantum states in Chern-Simons theory hence correlators in (chiral) current algebra (i.e: WZW) conformal field theory (Lit. 2.24).

(2) [SS22-Def, §2]: A technical result known as the "*hypergeometric integral construction*" (Lit. 2.25) of KZ-solutions serves to show that just these $\widehat{\mathfrak{su}}_2^k$-monodromy braid representations have a natural construction in algebraic topology (Lit. 2.12), where they are given by the monodromy of canonical Gauss-Manin connections (Lit. 2.23) on fiberwise twisted cohomology groups (Lit. 2.15) of configurations spaces of points in the plane (Lit. 2.18).

This is striking, because the first item above, when taken at face value, invokes a fairly long and intricate sequence of constructions from conformal field theory and representation theory of affine Lie algebras, while the second item invokes only the most basic concepts of algebraic topology. It is via this translation from conformal quantum field theory to plain algebraic topology (discussed in §4) that topological quantum gates can be fully grasped by a homotopically typed programming language (as discussed in §5). More in detail, the structure of our construction and proof is indicated in the following flow diagram:
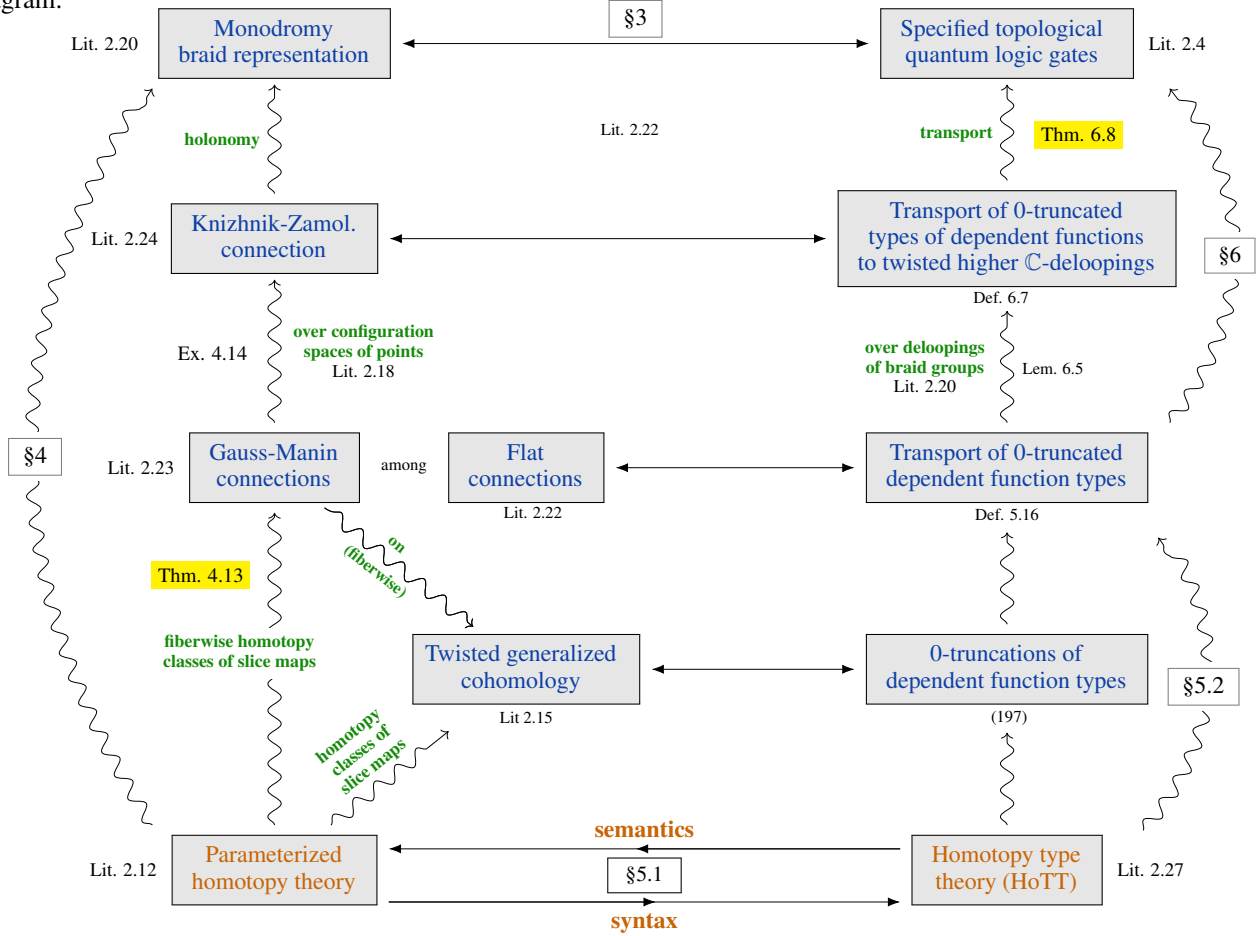
The novel results are Thm. 4.13 and Thm. 6.8 ; the rest is infrastructure (locally well-known to respective experts) connecting these to a novel global picture of anyonic quantum braid gates formalized in HoTT.

Conclusion. As a result, what we offer here is a previously missing understanding of topological quantum gates which is:
  (i) naturally rooted in the modern foundations of homotopical mathematics,
 (ii) fully aware of the physical principles underlying topological quantum materials, and
(iii) natively implementable in state-of-the-art programming certification languages such as `Agda`.

The first two points mean that the construction is amenable to pure mathematical analysis while at the same time reflecting the actual physics in question; and the third point makes this accessible to actual programming languages for verification and simulation.

Example application: Certified topological quantum compilation. Once topological quantum hardware becomes available, and generally when *simulating* topological quantum circuits, the key step (e.g. [ZW20]) in implementing any quantum algorithm is its *compilation* to a circuit (*Fig. QC*) consisting of those logic gates that the topological hardware actually offers.

While the *Solovay-Kitaev Theorem* [Ki97][So00] guarantees, under mild assumptions, that such quantum circuits exist for any prescribed accuracy (see [DN06][Br14]), it still requires work to find optimal circuits under given operational constraints (e.g. [BHZS05][HZBS07][HBS09][KBS14][JS21, §IV]). Our result provides a certification language (Lit. 2.26) that reflects the analytic detail of the topological quantum compilation, achieving the combination of:

(1) modern exact real computer analysis (cf. Lit. 2.29 & pp. 72) for certifying that one unitary operator approximates another;
(2) novel topological language constructs for certifying that/if such operators arise from physical anyon braiding (Thm. 6.8).

This may open the door to full formal verification of quantum compilation of realistic topological quantum gates based on their actual physical operation principles; a task arguably constitutive for quantum computation in the long run (cf. Lit. 2.5).

Outlook. Here we are just scratching the surface of the following topics on which type theory might now be brought to bear:

- The $\mathfrak{su}_2$-monodromy which we encode as a homotopy type in Def. 6.7, Thm. 6.8 is equivalently (one chiral half of) the quantum propagator of $\mathfrak{su}_2$-**Chern-Simons theory** (cf. [CR97][Ga00, §5]) on a cylinder with "Wilson line" insertions.

- As a type constructor, Def. 6.7 clearly works beyond the specific choice of twisting (57): that choice is needed only to identify its categorical semantics as the familiar KZ-monodromy on conformal blocks. But more general twistings may still have recognizable semantics. For instance, in [SS22-Def, Rem. 2.22] we provided evidence that for fractional levels $\kappa \mapsto \kappa/q$ we obtain the conformal blocks of **logarithmic CFTs** expected for some anyonic quantum states.

- The braids (Lit. 2.20) embodied by Def. 6.7 are a special case of embedded framed **cobordism**. Indeed, the way we found the construction presented here is (following [SS22-Conf][CSS21]) as a special case of (twisted) cohomology of *Cohomotopy moduli spaces*, the latter encoding embedded framed cobordism by Pontrjagin's theorem (cf. [SS21-MF][SS20-Tad]).

Last but not least, a couple of interesting extensions of homotopy type theory (Lit. 2.27) seamlessly lend themselves to handling these and further aspects of topological quantum programming — we aim to discuss these elsewhere:

- **Linear HoTT and Universal Quantum Programming.** A natural extension of homotopy types by (dependent) *linear* – namely: *quantum* – types exists ([Ri22], anticipated in [Sc14a]) providing a universal *quantum* programming language (see [Sc22b]) reflecting the linearity of quantum types (such as the no-cloning property) together with the quantum measurement process and classical control mechanisms.

  In fact, beyond ordinary quantum types, this knows about dependent *higher* homotopical linear types, namely about *parameterized spectra*, such as those consisting of the parameterized Eilenberg-MacLane types (Lit. 2.14) used here. This means that essentially the same homotopy data structure of Def. 6.7 but constructed as a linear homotopy type, integrates topological quantum gates into a full-blown quantum circuit language with quantum measurement gates and classical control; this is discussed in companion articles [SS23-QM][SS23-EoS].

- **Cohesive HoTT and Solid State Physics of Topological Quantum Gates.** The analysis of [SS22-Ord] concludes (Lit. 2.17) that realistic topological quantum gates implemented in topologically ordered quantum materials are described by the holonomy of Gauss-Manin (GM) connections (Lit. 2.23), not just on twisted ordinary cohomology (as discussed here), but on *T*wisted & *E*quivariant & *D*ifferential (*TED*) topological K-cohomology (Lit. 2.16), which refines the KZ-monodromy through a comparison map called the *secondary Chern character*. This predicts physical corrections to the traditional expectation of monodromy braiding gates operating by the holonomy of the Knizhnik-Zamolodchikov (KZ) connection: namely, that braid gate operations in the co-kernel of the secondary TED-Chern character are not actually realizable by adiabatic braiding of defects in real topological quantum materials, while new K-theoretic braiding operations appear in the kernel.

  The *differential* and *equivariant* structure required to resolve these corrections from solid state physics is naturally incorporated into homotopy theory by the consideration of *cohesive* (following the terminology of [La94][La07]) systems of modal operators ([SSS12, §3][Sc13][SS20-Orb][SS21-Bun]) in a way that again lends itself to implementation in type theory, then called *cohesive homotopy type theory* ([ScSh14][Sc15][We17][Sh18][My21][My22][MR23]).

These *linear* and *cohesive* refinements of homotopy type theory should naturally combine to a *cohesive linear homotopy type theory* (the corresponding semantics is discussed in [Sc13, §4.1.2][BNV16][GS19]), which we suggest to be the ultimate platform for certification of classically controlled quantum programming with realistic topological quantum gates: [1]

---
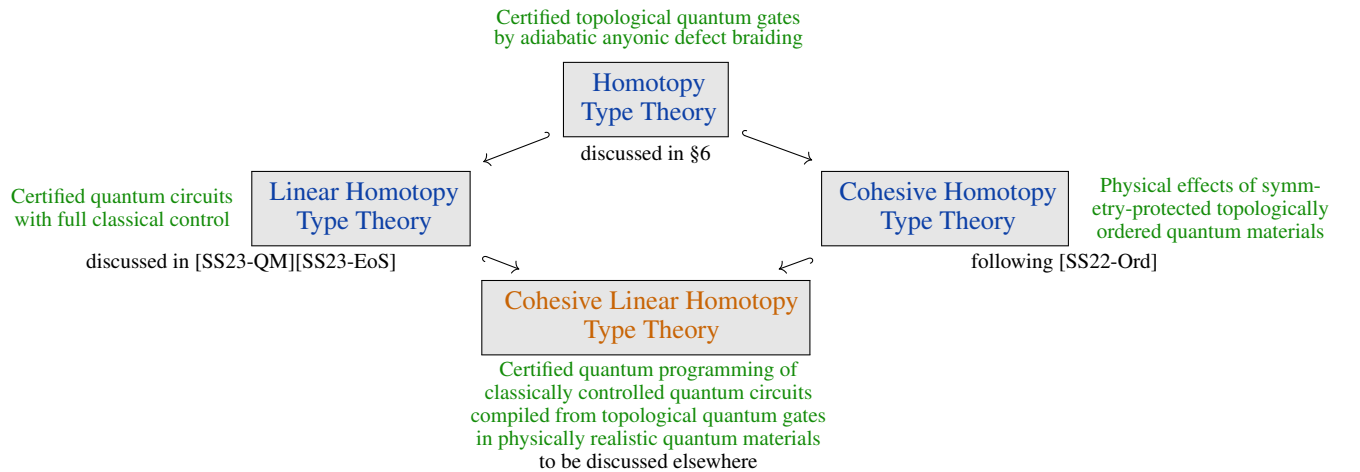
[1]Intriguingly, cohesive linear homotopy theory is also the semantic context in which to naturally make formal sense of the key ingredients of high energy physics, specifically of string/M-theory (cf. [Sc14b][SS20-Orb, p. 6]). This is in line with a deep relationship between strongly coupled quantum systems (such as anyonic topological order) and string/M-theory, cf. [SS22-Ord, Rem. 2.8][Sa23][Sc23].

2 Background and perspective

Since we are connecting areas of physics, mathematics, and computer science that have not been in close contact before, we provide here a kind of index for various technical terms, with brief explanations and pointers to the literature. The reader may or may not want to peruse this section linearly, and can certainly skip ahead to come back here only as the need arises.

Literature 2.1 (**Quantum computation**). The general idea of quantum computation was originally articulated by Yuri Manin [Ma80][Ma00], Paul Benioff [Be80] and Richard Feynman [Fe82], brought into shape by David Deutsch [De89], shown to be potentially of dramatic practical relevance by Peter Shor and others [Sh94][Si97]... *if* quantum error correction could be brought under control (Lit. 2.5), which was shown by Peter Shor and others [Sh95] to be at least a theoretical possibility.

Textbook accounts of the general principles of quantum computation and quantum information theory include: [NC10] [RP11][BCR18][BEZ20]. Impressions of the state of the field may be found in [Pr22] and in the discussion of NISQ machines (Lit. 2.3); elementary exposition leading up to the following discussion may be found in [Sc22a].

Here we are concerned with the *low hardware-level* formulation of computation, where elementary instructions ("logic gate names") are mapped to transformations on a state space (such as on the registers of a computing machine), and where programs consist of paths of such instructions ("logic circuits") executed as iterated such state transformations — we expand on this in §3. From this very basic perspective, the one key aspect of quantum computation to take note of is just that:

Where an operation on a classical state space is a *map of finite sets* (of states), the analogous quantum operation is an *invertible* (Lit. 2.2) *linear map* of finite-dimensional *complex vector spaces*. (In Thms. 4.13, 6.8 these vector spaces arise in the form of *complex cohomology groups*, Lit. 2.15.)

Classical computation		**Quantum computation**	
finite set of states $\{0,1\}^D$ $\xrightarrow{\text{computation}}$	finite set of states $\{0,1\}^D$	finite dim. vector space of states $\mathbb{C}^{2D}$ $\xrightarrow[\text{(complex linear map)}]{\text{computation}}$	finite dim. vector space of states $\mathbb{C}^{2D}$
instruction path $t_1 \rightsquigarrow t_2$		instruction path $t_1 \rightsquigarrow t_2$	

Table D. Where a classical *digital* computation (program) is a function (a map) between finite sets (of bits), a corresponding quantum computation is (in particular) a *linear map* between finite-*dimensional* complex vector spaces. Compare the discussion in §3.

This **complex linearity of quantum computation processes** is to a large extent the source of their richness, since:
  (1.) there is now a *continuum* (an un-countable multitude) of possible states, hence of data (see Lit. 2.29) where a classical digital computer sees only a finite set,
  (2.) the complex coefficients allow this multitude to interact usefully via constructive/destructive interference (in all-important contrast to the case of classical analog and/or probabilistic computing).

In particular, the *verification/certification* (Lit. 2.26) of a quantum computational process (such as a proof that a given set of quantum gates *compiles* to a prescribed linear map, to within specified accuracy, cf. p. 3) is in general a problem in *constructive analysis/exact complex computer arithmetic* (Lit. 2.29).

Other key aspects of quantum computation, such as unitarity of these linear maps (i.e., their preservation of Hilbert space structure and hence of the probabilistic interpretation of quantum physics) on the one hand, and non-unitary state collapse (under quantum measurement) on the other, play a tangential role in the present article but will be brought out fully in the companion articles [SS23-QM][SS23-EoS].

Literature 2.2 (**Reversible computing**). The conceptual development of quantum computation leads to a re-evaluation of basic principles of computing. For instance, at the fundamental microscopic level, all physical processes — and hence also all computational processes — are *reversible*. Executing non-reversible classical logic gates (such as AND : Bit$^2 \to$ Bit) means for a computer to discard information into — hence to interact with — an arbitrarily complex environment: this is known as *Landauer's principle* [Be03]. Trivial as this may superficially seem, it is exactly such environmental interactions that have to be shielded from a microscopic system in order to realize a coherent quantum computation process (Lit. 2.5) — which is then reversible as predicted by quantum mechanics according to unitary Schrödinger evolution. This way, quantum circuits (*excluding* quantum *measurement* gates, which we disregard here but turn to in the companion articles[SS23-QM][SS23-EoS]) are a form of reversible computation (e.g. [Ki97, §2.2][NC10, §1.4.1, 3.2.5][Am+20, §9]).

Curiously, exactly this fundamental reversibility of computation is natively reflected — under the formulation (in §3) of low-level computation as "path-lifting" — by homotopy-typed programming languages (Lit. 2.27), where it corresponds to the invertibility (78) of "parameter paths" in the guise of identification certificates (71) and thus to the denotational semantics of homotopy type theory in spaces/∞-groupoids reviewed in §5.1.

Literature 2.3 (**NISQ computers**). Currently existing quantum computers (such as those based on "superconducting qbits", see e.g. [CW08][HWFZ20]) serve as proof-of-principle of the idea of quantum computation (Lit. 2.1) but offer puny computational resources, as they are (very) *n*oisy and (at best) of *i*ntermediate *s*cale: "NISQ machines" [Pr18][LB20]. What is currently missing are noise-protection mechanisms that would allow to scale up the size and coherence time of quantum memory. The foremost such protection mechanism arguably (Lit. 2.5) is *topological* protection (Lit. 2.4).

Literature 2.4 (**Topological quantum computation**). The idea of topological quantum computation — specifically (we give an exposition in §3) of topological quantum logic gates operating by adiabatic (Lit. 2.6) braiding (Lit. 2.18) of worldlines of anyonic defects (Lit. 2.17) in topological quantum materials (Lit. 2.7) controlled by Chern-Simons/WZW theory (Lit. 2.24) — is attributed to [Ki03][Fr98][FKLW03][FLW02b] (see also [KL04]), further expanded on in [NSSFD08] (analogous discussion for discrete gauge groups, i.e. Dijkgraaf-Witten theory, is in [OP99][Mo03][Mo04]). Reviews include [BP08][RW18] and textbook accounts include [Wa10][Pa12][SL13][St20][Si21].

Interestingly, the key principle of topological quantum computation and also in more generality (as discussed in §3) was already clearly articulated in [ZR99] under the name "Holonomic Quantum Computation".

Beware that traditional literature tends to discuss the issue in the abstract, disregarding (cf. [Va21]) the question of physical realization of anyonic defect braiding in actual materials, even theoretically (Lit. 2.9). A (previously) popular experimental setup via "Majorana zero modes" now seems dubious (Lit. 2.8). On the other hand, the braiding of band node defects in the momentum space (Lit. 2.7) of topological semimetals may be promising (Lit. 2.10).

Literature 2.5 (**Need for topological quantum protection**). While the idea of topological quantum computation (Lit. 2.4) may at times be presented as just one of several interesting alternative approaches to quantum computation, there are good arguments that any practically useful, hence scalable, quantum computing architecture, beyond the currently available NISQ machines (Lit. 2.3), must be topologically protected by necessity:

> [Sau17]: "small [NISQ] machines are unlikely to uncover truly macroscopic quantum phenomena, which have no classical analogs. This will likely require a scalable approach to quantum computation [...] based on [...] topological quantum computation (TQC) [...] The central idea of TQC is to encode qubits into states of topological phases of matter. Qubits encoded in such states are expected to be topologically protected, or robust, against the 'prying eyes' of the environment, which are believed to be the bane of conventional quantum computation."

> [DS22a]: "The qubit systems we have today are a tremendous scientific achievement, but they take us no closer to having a quantum computer that can solve a problem that anybody cares about. [...] *What is missing is the breakthrough* [...] *bypassing quantum error correction by using far-more-stable qubits, in an approach called topological quantum computing.*"

Literature 2.6 (**Adiabatic nature of topological quantum gates**). The operation on degenerate quantum ground states by braiding (Lit. 2.18) of anyonic (Lit. 2.17) defect worldlines (Lit. 2.4) is an instance of unitary transformations given by the *Quantum Adiabatic Theorem* (e.g. [Ne80][ASY87][RO12][BDF20]). That topological quantum computation is, in this sense, an instance of "adiabatic quantum computation" [FGGS00][AvDKLLR07] is at times hard to discern from the literature. This is on the one hand because the term tends to be used by default for (non-topological) *quantum annealing* processes [GH20]; and on the other hand, because the experimental focus on detecting "Majorana zero modes" (Lit. 2.8) entirely stuck at endpoints of nanowires seems to have led to disregard for the eventual need for braiding, altogether.

References that do make explicit the adiabatic nature of topological quantum gate processes include (besides [ZR99]): [CFP02, p. 2][FKLW03, pp. 7][NSSFD08, p. 6] [CGD11][RO12, p. 1][Pa12, p. 50, 52][CLBFN15][MCMC19][St20, p. 321].

We amplify this because adiabatic transformations in quantum physics are an instance of *path lifting* (Lit. 2.30) — namely of classical *parameter paths* to (linear) maps between quantum state spaces depending on (namely: fibered over) these classical parameters (19). Concretely: Quantum adiabatic transformations are the parallel transport (Lit. 2.22) of a corresponding connection (a flat connection for "topological" processes, Lit. 2.22) on the quantum state bundle [Si83][WZ84] (review in [Be89][Va18][St20, §2]) whose holonomy/monodromy is known as *Berry phases* [Be84].

Curiously, this phenomenon of *adiabatic quantum transformations via path lifting* is natively captured by homotopically-typed languages (Lit. 2.27), see §3.

Literature 2.7 (**Topological quantum materials**). A *topological phase of matter* (textbook accounts include [Va18][St20, §II]) is a ground state of a crystalline material in which the vector bundle of occupied Bloch states (of electrons/positrons in the fixed atomic lattice) over the Brillouin torus (the lattice momentum space, cf. Lit. 2.10) is topologically non-trivial, in that it has a nontrivial class in (twisted, equivariant) topological K-theory (Lit. 2.16). A ground state is said to exhibit *topological order* [We91][St20, §6.2] if it is degenerate and its adiabatic (Lit. 2.6) dependency on the position of defects constitutes a braid group representation (Lit. 2.21). Review with comprehensive pointers to the literature is in [SS22-Ord].

Literature 2.8 (**Majorana zero modes?**). In recent years a lot of attention towards potentially realizing topological quantum computation (Lit. 2.4) has been focused on the proposal that topological quantum states might be realized in the form of "Majorana zero modes" [Ki01] localized at the endpoints of super/semi-conducting nanowires; see reviews in [Ma22][DS22b]. However, a series of prominent claims of experimental detection of such modes have now been retracted or called into question (see e.g. [DSP21]). Even if such modes could be detected, they would, by design, be immobile and hence impossible as a platform for topological braid quantum gates (Lit. 2.4) in the original sense based on adiabatic braiding movements (Lit. 2.6). While it has been argued that some kind of effective braiding of immobile "Majorana zero modes" could be emulated by other means, it seems implausible (certainly unproven) that this could be true in a sense which would still enjoy the topological protection property of actual adiabatic braiding (Lit. 2.6), thus defeating the point.

Therefore, it seems to us that even if Majorana zero modes localized in nanowires were real, they would be unlikely to support the original and established notion of topological quantum gates (Lit. 2.4) with which we are concerned here. A more promising experimental realization of actual topological quantum gates may be given by braiding of band nodes in momentum space (Lit. 2.10)

Literature 2.9 (**Mathematical specification of anyons**). While the basic idea of *anyons* (Lit. 2.17) may seem clear, it is not just their experimental realization that has been elusive (cf. Lit. 2.8), but arguably already their theoretical derivation and determination within mathematical physics had been sketchy (cf. [Va21]). Much of the mathematical/theoretical solid state physics literature takes it for granted that the answer to "*What is an anyon species?*" (cf. Lit. 2.19) is: "*Any unitary braid group representations!*", hence: *Any unitary R-matrix!* (e.g. [KL04], cf. Lit. 2.21); or more recently and such as to account for different anyon species (Lit. 2.19): "*Any unitary braided fusion category!*" (going back to [Ki06, §8, §E][NSSFD08, pp. 28][Wa10, §6.3] and repeated, usually without attribution, in numerous reviews, e.g. [RW18, §2.4.1][Ro22, §2.2]).

The basic idea, at least for defect anyons, is that their classical parameter space is the configuration space of points (Lit. 2.18) in an effectively 2-dimensional quantum material (or in its dual Brillouin torus, Lit. 2.10) so that the Berry phase transformation under adiabatic movement of the anyon positions (Lit. 2.6) on the material's topologically ordered ground state (Lit. 2.7) constitutes a braid representation (Lit. 2.21).

While this sounds plausible, the only detailed derivation (as far as we are aware) from first physics principles is our recent argument in [SS22-Ord, §3.3]; and the result there is something more specific and also slightly modified: Anyonic wavefunctions are given by certain K-theoretic (Lit. 2.16) corrections (cf. p. 4) to, specifically, the $\widehat{\mathfrak{su}_2}^k$-monodromy braiding rules ($\mathfrak{su}_2$-anyons, cf. [SS22-Ord, Rem. 3.12]). More concretely — and this is what matters for the present discussion — anyon braiding is given by the holonomy of the $\mathfrak{su}_2$-KZ connection (Lit. 2.24) arising as the Gauss-Manin connections (Lit. 2.23) on bundles of twisted generalized cohomology groups (Lit. 2.15) over configuration spaces of points (Lit. 2.18). While this cohomological derivation of anyons overlaps — up to some corrections and specifications — with the traditional postulate of braided fusion categories (see [SS22-Ord, Rem. 3.12]), it has the striking distinction that it directly lends itself (via the translation in §4) to certification in homotopically typed programming languages (as such discussed in §6).

Literature 2.10 (**Braiding of band nodes in momentum space**). A moment of reflection reveals that the notion of topological braid quantum gates (Lit. 2.4) relies only on properties of the abstract configuration space of the quantum material (its ground states, Lit. 2.7), not on the tacit assumption that this is identified with physical space ("position space"). In particular, all arguments usually made for would-be anyonic defects (Lit. 2.17) in position space immediately apply to the material's *momentum space* (the Brillouin torus, Lit. 2.7) if suitable topological defects are present there. While this possibility has received attention only very recently ([SS22-Ord, Rem. 3.9]), momentum-space defects in topological quantum materials are well-known, well-established, and well-studied: These are *band nodes* in topological semi-metals (i.e., Dirac/Weyl points). Moreover, their controlled movement and potential braiding (Lit. 2.20) by (adiabatic, Lit. 2.6) manipulation of external parameters has been established in a variety of (meta-)materials and seems a rather generic property. References are collected in [SS22-Ord, Rem. 3.9].

Literature 2.11 (**Holographic models for strongly correlated quantum systems**). Not just the physics of topologically ordered phases of matter (Lit. 2.4), but generally that of any *strongly interacting* quantum system falls outside (e.g. [Str13]) most of traditionally available "perturbative" analytic methods. The popular claim that quantum field theory is the most precise physical theory ever conceived tacitly refers to one special case where available *perturbative* methods happen to work well, namely in quantum electrodynamics (QED). But already quantum chromodynamics (QCD) at room temperature – which is meant to describe nothing less than (the nuclei of) ordinary matter – is so strongly coupled that there is currently no coherent theory for it (just a zoo of partial models and computer lattice simulations). Finding the *non-perturbative completion* of QCD which would explain in detail how "constituent quarks" are "confined" within hadronic bound states such as atomic nuclei, is an open "Millennium Problem" (for good accounts see [RS20][Ro21][DRS22][Ro22]).

One promising approach to this problem is to focus on the dynamics of the "flux tubes" which are thought to connect, and thereby strongly bind, any quark to another. Subtle quantum effects make these flux tubes behave [Po98][Po99][Po02] like "strings" propagating in a higher dimensional spacetime with only their endpoints (the quarks) constrained to the actual

spacetime hypersurface, which now appears as a $(3+1)$-dimensional hypersuface (called a "brane") embedded in a higher dimensional "bulk" spacetime. This formulation has come to be known as "holographic QCD" (reviewed in [Er15][RZ16, §4]) a variant of the more famous but less realistic *AdS/CFT correspondence* (reviewed in [AGM00]). The analogous description of strongly coupled condensed matter systems as intersecting brane models in string theory is known as *holographic quantum matter* (or variant names) [ZLSS15][HLS18]. Here the role of the nuclear force is played by the "Berry connection" (Lit. 2.6) and that of quarks by fermionic quasi-particles; see the dictionary in [SS22-Def, Table 1].

While the full non-perturbative completion of this *string theoretic*-picture of strongly-coupled quantum theory also still remains to be formulated (working title: "M-theory" [Du99]), in this case, there is a tight web of hints and consistency checks available (due to insights that came to be known as "the second superstring revolution" [Schw96]). It is by exploring such hints (in [SS22-Def], following [SS22-Conf]) that the model for topological quantum gates discussed here was discovered in [SS22-Ord]. While none of this string-theoretic background is needed (nor assumed) for the present article, it may help to put the constructions into their broader perspective.

Literature 2.12 (**Topology and Homotopy theory**). Strictly speaking, *topology* is the study of general spaces — "topological spaces", such as Euclidean space of any dimension, but also mapping spaces (e.g. [AGP02, §1]), configuration spaces (Lit. 2.18), parameter spaces, etc. — up to continuous deformations (homeomorphisms); while *homotopy theory* is the study of such spaces up to the coarser notion of (weak) homotopy equivalences (which is about continuous deformations of *continuous maps* between spaces): *homotopy types*. However, the term *topological* is often used for what more precisely would be called *homotopical*, so that the terminological distinction is blurred (cf. [Mill19, p. vii]). This is notably the case for "topological quantum computation" (Lit. 2.4) which is really concerned with quantum gates parameterized by *homotopy* classes of continuous (hence: topological) computation paths, where the invariance under deformations (namely: under homotopy) reflects the robustness of the computation against noise, cf. *Fig. H* in §3.

This way, the qualifiers "topological" in "topological quantum computation" and "homotopy" in "homotopy type theory" are actually synonymous!

On the other hand, one distinguishes further between:
- **point-set topology** – which is topology/homotopy theory of the familiar notion of spaces consisting of sets of points equipped with topological cohesion;
- **algebraic topology** – which studies topological spaces through the homological algebra of their *(co)homology groups* (Lit. 2.15); and it is the tendency of these to only depend on the underlying homotopy types which makes this *de facto* be a subject of homotopy theory);
- **abstract homotopy theory** – which is concerned with models for homotopy types beyond topological spaces – notably simplicial sets, cf. around (107) below – and with variants, such as local, equivariant, stable, etc. homotopy types.
- **parameterized** versions of all of these, where all spaces and hence all constructions on these are allowed to vary over some parameter space.

Textbook introductions to basic point-set topology include [Ja84][Mu13] and introductions to the algebraic topology and homotopy theory based on these topological foundations include [Sp66][Ja84][Ro88][Br93][Ha02][AGP02][tD08][Str11] [Ar11]. (For early history of the subject see [Hi88].) The *parameterized* point-set topology/homotopy theory which we need in §4 is laid out in [MS06], following [Bo70][BB78]. Abstract homotopy theory (going back to [Br65][Qu67][Br73][Ad74]) is reviewed for instance in [KP97][Ri14][Ri20], and concise summaries of facts needed in our context are given in [FSS20-Cha, §A][SS21-Bun, §3.1]).

For standard technical reasons, we assume all topological spaces in the following to be (a) compactly generated (cf. [SS21-Bun, (1.2)]) and (b) of the homotopy type of a CW-complex.

Literature 2.13 (**Homotopy groups**). The basic homotopical invariants of topological spaces X (Lit. 2.11) are their *homotopy groups* $\pi_n(X)$ (e.g., [AGP02, §3]). Foremost among these is the *fundamental group* $\pi_1(X)$ of a connected space, which is the group of deformation classes of closed loops in the space, based at any fixed point (e.g. [AGP02, §2.5]). Similarly, the *higher homotopy groups* $\pi_n(X)$ of a connected space are deformation classes of based maps from the n-sphere S^n into X – these are all abelian for $n \geq 2$.

Finally, for not necessarily connected spaces X we have their set of path-connected components, denoted $\pi_0(X)$. For instance, the connected components of a *mapping space* $\mathrm{Map}(Y, X)$ are the *homotopy classes* of maps $Y \to X$.

Literature 2.14 (**Eilenberg-MacLane spaces**). For any abelian group A and $n \in \mathbb{N}_{\geq 1}$ there exists a connected topological space unique up to homotopy equivalence (Lit 2.12) – called an *Eilenberg-MacLane (EM) space* (e.g. [AGP02, §6]) and equivalently denoted $K(A, n)$ or $B^n A$ – whose homotopy groups (Lit. 2.13) are concentrated on A in degree n:

$$\pi_{n'}\left(B^n A\right) \simeq \begin{cases} A & \text{if } n' = n, \\ * & \text{otherwise}. \end{cases}$$

If $n = 1$ then BG exists also for non-abelian groups G, known as the classifying space for G-principal bundles.

For abelian groups A, their nth EM-space is naturally equivalent to the based loop space of the $(n+1)$st

$$B^n A \simeq \Omega B^{n+1} A$$

which makes them constitute a spectrum HA, the Eilenberg-MacLane spectrum for A.

Finally, given a group G (possibly non-abelian) acting by homomorphisms on the abelian group A, we have a corresponding parameterization of EM-spaces over the classifying space of G:

$$
\begin{array}{ccc}
B^n A & \longrightarrow & B^n A /\!\!/ G \\
& & \downarrow \\
& & BG
\end{array}
$$

Literature 2.15 (**Cohomology**). The *ordinary cohomology groups* $H^n(X;A)$ of a topological space (Lit. 2.12) are traditionally introduced in terms of cochains, but the equivalent reformulation central for our purpose is (e.g. [AGP02, §7.1, Cor. 12.1.20] cf. [FSS20-Cha, Ex. 2.2]) as homotopy classes of maps into an Eilenberg-MacLane space (Lit. 2.13):

$$H^n(X;A) = \pi_0\big(\mathrm{Map}(X, B^n A)\big).$$

If $n = 1$ and G a possibly non-abelian group, then

$$H^1(X;G) = \pi_0\big(\mathrm{Map}(X, BG)\big)$$

is called the *first non-abelian cohomology* of X. This is the evident special case of *generalized non-abelian cohomology* (cf. [FSS20-Cha, §2.1]) where for *any* space $\mathscr{A}$ we take the cohomology of X with coefficients in $\mathscr{A}$ to be

$$\mathscr{A}(X) :\equiv \pi_0\big(\mathrm{Map}(X, \mathscr{A})\big). \tag{1}$$

For example, if $\mathscr{A} = S^n$ then this yields the cohomology theory known as *Cohomotopy*.

Often this is considered for the special case when $\mathscr{A}$ is part of a sequence $\{E_n\}_{n \in \mathbb{N}}$ of pointed spaces where each is equipped with a weak homotopy equivalence to the based loop space of the next one (a *spectrum* of spaces). In this case

$$E^n(X) = \pi_0\big(\mathrm{Map}(X, E_n)\big)$$

is the *Whitehead generalized cohomology* of X with coefficients in E.

For example, if $E_0 = \mathrm{Fred}^{\mathbb{K}}$ is the space of *Fredholm operators* on a separably infinite-dimensional $\mathbb{K}$-Hilbert space, then

$$\mathrm{KU}^{-n}(X) = \pi_0\left(\mathrm{Map}\big(X, \Omega^n \mathrm{Fred}^{\mathbb{C}}\big)\right), \qquad \mathrm{KO}^{-n}(X) = \pi_0\left(\mathrm{Map}\big(X, \Omega^n \mathrm{Fred}^{\mathbb{R}}\big)\right)$$

is the *topological K-cohomology* of X (cf. Lit. 2.16)

More generally, if the coefficient space $\mathscr{A}$ is acted on by a group G (which we shall assume to be discrete for ease of exposition), then one may form the $\mathscr{A}$-fiber bundle over the classifying space BG which is *associated* to the universal G-principal bundle EG (also known as the *Borel construction* or *homotopy quotient* of $\mathscr{A}$ by G). Accordingly generalizing the definition (1) of cohomology to the "slice over BG" generalized cohomology to *twisted cohomology* (see [FSS20-Cha, §2.2])

If $\mathscr{A} = B^n\mathbb{C}$ is an n-fold delooping of (the discrete abelian group underlying) the ring of complex numbers and $G \equiv (\mathbb{C}^\times)^\flat$ denotes the (discrete!) group of units with its canonical multiplication action on $\mathbb{C}$, then the local coefficient bundle E on the right is, for $n = 0$, just a flat line bundle $\mathscr{L}$ (Lit. 2.22) or rather its underlying horizontal covering space (Lit. 2.30).

For general n it is the local coefficient bundle for twisted ordinary complex cohomology with

$$H^n(X; \mathscr{L}) \;=\; H^{n+\tau}(X; \mathbb{C}) \;\simeq\; \pi_0\Big(\mathrm{Map}\big(X, B^n\mathbb{C} /\!\!/ \mathbb{C}^\times\big)_{B\mathbb{C}^\times}\Big). \tag{2}$$

This is the reason why flat vector bundles are often referred to simply as "local systems" (namely: of coefficients for twisted ordinary cohomology), see Lit. 2.22. Discussion of the corresponding twisted cohomology goes back to [De70, §2, 6], a textbook account is [Vo03II, §5.1.1].

In the form of "cohomology of a local system" $\mathscr{L}$ on the left of (2) twisted cohomology is used abundantly in the discussion of the hypergeometric integral construction (Lit. 2.25).

Literature 2.16 (**K-Theory classification of topological phases**). At (very) low temperatures, the electrons in a "weakly correlated" crystalline material will incrementally fill up the lowest available 1-electron ground states in the Coulomb background of the atomic nuclei which constitute the crystal lattice (e.g. [Va18, §2]). As the wave vectors/momenta of the electrons range through the *Brillouin torus* of available lattice momenta (e.g. [FM12, p. 52]), these *valence states* form a complex vector bundle (18) over the Brillouin torus. If this vector bundle is "topologically non-trivial", one says that the crystalline material is in a *topological phase of matter* (Lit. 2.7).

In more detail, the full relativistic computation of the electron states shows that their ground state bundle forms a class in the *topological K-cohomology* (Lit. 2.15) of the Brillouin torus, constituted by the electron valence bundle and a possible admixture of positronic ground states ([SS22-Ord, Fact 2.3], following a famous original suggestion due to [Ki09]). Moreover, if the dynamics of the electrons respects crystalline symmetries, then so does their valence bundle which is then classified by the corresponding *equivariant K-theory* of the Brillouin torus [FM12]. More generally, if the electron dynamics respects also some internal symmetries and/or is subject to Berry phases, their ground states are classified in TED K-theory [SS22-Ord, §2.3, 3.1].

However, if the electrons in the material are *strongly correlated* (strongly interacting, cf. Lit. 2.11) then their ground states must be represented more properly by joint n-electron states — for larger n the larger the interaction. By the Pauli exclusion principle (which prevents n-electron states to be non-trivial if any pair of their momenta coincides) these form a vector bundle over the *configuration space of n distinct points* (Lit. 2.18) in the Brillouin torus. Accordingly, it is generally the TED K-cohomology of these n-configuration spaces which classifies such *topologically ordered* (strongly correlated) topological phases [SS22-Ord, §3.2].

Literature 2.17 (**Anyons**). It is tradition (a comprehevsive list of literature on this and the following aspects is provided in [SS22-Ord, §3.3]) to motivate the concept of *anyons* as hypothetical particle species which are conceptually in between the firmly established *bosons* – whose joint quantum state picks up no transformation under their pair exchange –, and the firmly established *fermions* – whose joint quantum state picks up a sign $-1 = \exp(i\pi) \in \mathrm{U}(1)$ under pair exchange: For anyons one imagines *any* given phase $\exp(\phi i) \in \mathrm{U}(1)$ under pair exchange, whence the name. Under mild assumptions this makes non-trivial sense (only) for quantum particles of codimension 2, hence notably for pointlike (quasi-)particle excitations in effectively 2-dimensional materials (e.g. in atomic mono-layer crystals similar to graphene). In a superficially evident generalization of this notion, one speaks of *non-abelian* anyons if the exchange phases are more general unitary operators in some unitary group $\mathrm{U}(n)$ acting on a higher dimensional Hilbert space of their joint quantum states.

For a recent impression of the review literature on anyon physics, see for instance [MMN23].

However, despite the popularity of this motivation, it misleads about the crucial fact that such anyonic phase factors are supposed to depend on dynamical braid paths (Lit. 2.20) traversed by anyons around each other (thus constituting a braid representation, Lit. 2.21), while the boson/fermion phases are kinematical properties of their wavefunctions enforced even if no effective motion takes place. Indeed, close inspection of more concrete anyon models discussed in the theoretical physics literature reveals first of all two crucially different incarnations of anyons (cf. [SS22-Ord, Table 5]), neither of which quite fits the naive popular motivation:

On the one hand, abelian anyons are thought to be modeled by fermion-like *quasi-particles* which couple via a so-called "fictitious gauge field" such that their *quantum propagation* picks up *Aharonov-Bohm phases* with respect to each other. On the other hand, non-abelian anyons are thought to be modeled as *defects* in topologically ordered quantum materials whose *adiabatic movement* (Lit. 2.6) by classical parameter evolution makes them pick up *Berry phases*. At the same time, a microscopic understanding of how such non-abelian anyons would appear in realistic materials had arguably remained elusive (cf. Lit. 2.8).

We believe that the first satisfactory theoretical model for anyons — one which (a) unifies these two notions as well as (b) properly embeds their theory into that of the ambient topological phases of matter (Lit. 2.7) which are expected to host them — is the one laid out in [SS22-Ord, §3.3], where the reader may also find extensive referencing of the traditional literature on the matter. The upshot of the anyon model in [SS22-Ord, §3.3] is that:

(1) anyonic quantum ground states are classes in TED K-theory (Lit. 2.16) of configuration spaces of points (Lit. 2.18) in the punctured Brillouin torus of the given quantum material — approximated by their TED *Chern-character* which is a class in *ordinary* TED cohomology;

(2) the anyonic phase factors are the holonomies of the *Gauss-Manin connection* (Lit. 2.23) on the resulting bundles of TED (K-)cohomology groups over the configuration spaces of punctures in the Brillouin torus.

The main result of [SS22-Def] is that this prescription — in the approximation of ordinary TED cohomology as opposed to the full TED K-theory — actually reproduces much of the expected description of anyon braiding, and hence derives it from first principles. The lift to full TED K-theory then predicts subtle "torsion" corrections to this traditional picture (cf. p. 4).

Literature 2.18 (**Configuration spaces of (defect-)points**). The parameters whose motion controls topological quantum computation gate operation (Lit. 2.4) by adiabatic (Lit. 2.6) braiding (Lit. 2.20) are configurations of *defects* (Lit. 2.17) in a quantum material (Lit. 2.7), often tacitly assumed to be spatial positions but possibly instead being critical momenta given by points in "momentum space" (Lit. 2.10). For such defects to potentially be of anyonic nature they need to have co-dimension = 2 in the ambient quantum material, which for the case of point-like defects means that the quantum material must be an effectively 2-dimensional crystal consisting of a single or a very small number of atomic layers (such as now familiar from *graphene*). This is the case we are focusing on here.

Key is the assumption that these defects remain separated from each other, hence that the defect points have *distinct positions* (possibly meaning: distinct positions in momentum space, hence distinct wave vectors!). In physical reality, it may be possible to force defects to merge, but for the materials of interest this should correspond to an effectively discontinuous change of the system's ground state, which we disregard by the same rationale by which we disregard, say, the process of destructing the crystal structure altogether: while physically possible, this takes us out of the regime of quantum materials whose description is of interest here.

In conclusion then, the relevant parameters of defect positions for topological quantum gates move in a space of configurations of distinct points in a surface. For actual 2-dimensional crystals, this surface is a (Brillouin-)*torus*, but for simplicity here we will disregard the global topology of tori, which means that we consider configurations of points in the plane $\mathbb{R}^2$, often regarded as the complex plane $\mathbb{C}$ in this context.

Figure D – Configuration of (defect-)points in the plane.
In laboratory realization the plane represents an atomic mono-layer (at most a small multi-layer) of a crystalline quantum material (Lit. 2.7), or its Fourier-dual space of wave-vectors (of excitations in the crystal); and the points represent anyonic defects (Lit. 2.17) in this structure (such as band nodes of a 2d semi-metal, Lit. 2.10).
Mathematically, the whole configuration is itself one point in the *configuration space of points* (3) in the plane (cf. Lit. 2.19). Jointly moving the defect-positions means to move along a curve in this configuration space (Lit. 2.20), hence to move the points jointly along a "braid" (Lit. 2.20).
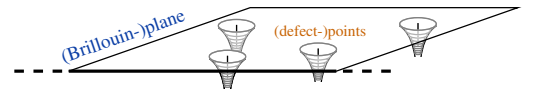
Such parameter-*spaces of configurations of points* in the plane have long been studied in pure mathematics [FaN62]; a textbook account is [FH01], gentle introduction includes [Will20][Wils18], more details may be found in [Co09], and brief review in our context is in [SS22-Conf]. Much of their mathematical interest lies in the fact (5) that these spaces are Eilenberg-MacLane spaces (Lit. 2.14) of braid groups [FaN62][FoN62] (Lit. 2.20).

Concretely, the *configuration space* of $N+1$ *ordered* points (carrying labels $1, 2, ..., N+1$, cf. Lit. 2.19) in the plane $\mathbb{R}^2$ is the complement in the product space $(\mathbb{R}^2)^{N+1}$, i.e. in the space of possibly coincident positions of $N+1$ points, away from the subspace where any pair of them coincides:

$$\underset{\substack{\text{configuration space}\\\text{of ordered points}}}{\underset{\{1,\cdots,N+1\}}{\mathrm{Conf}}\left(\mathbb{R}^2\right)} \;:\equiv\; \underset{\substack{\text{space of unconstrained}\\\text{positions of ordered points}}}{\left(\underbrace{(\mathbb{R}^2)\times\cdots\times(\mathbb{R}^2)}_{N+1\,\text{factors}}\right)} \;\overset{\text{minus}}{\setminus}\; \underset{\substack{\text{subspace where any pair of points coincide}}}{\left\{z_1,\cdots,z_{N+1}\in\mathbb{R}^2 \,\middle|\, \underset{I\neq J}{\exists}\; z_I = z_J\right\}} \tag{3}$$

The symmetric group $\mathrm{Sym}(N+1)$ evidently acts on such an ordered configuration space by permuting the order of the labels;

11

and the quotient of this action is the *un-ordered configuration space*

$$\underset{\substack{\text{configuration space}\\ \text{of un-ordered points}}}{\text{Conf}_{N+1}(\mathbb{R}^2)} \; :\equiv \; \underset{\substack{\{1,\cdots,N+1\}\\ \text{configuration space}\\ \text{of ordered points}}}{\text{Conf}}(\mathbb{R}^2) \; \underset{\text{modulo permutations}}{\Big/} \; \text{Sym}(N+1) \,. \tag{4}$$

These configuration spaces (3) (4) canonically inherit the structure of topological spaces (Lit. 2.12 – in fact of smooth manifolds) and the statement is ([FaN62, p. 118][FoN62, §2], review in [Wils18, §2.2][Will20, pp. 9]) that the underlying homotopy type (Lit. 2.12) is the delooping (the first Eilenberg-MacLane space, Lit. 2.14) of the (pure) braid group (Lit. 2.20):

$$\underset{\{1,\cdots,N+1\}}{\text{Conf}}(\mathbb{R}^2) \; \underset{\text{whe}}{\simeq} \; B\text{PBr}(N+1)\,, \quad \text{i.e.:} \quad \pi_k\Big(\underset{\{1,\cdots,N+1\}}{\text{Conf}}(\mathbb{R}^2)\Big) = \begin{cases} \text{PBr}(N+1) & \text{if } k=1 \\ * & \text{otherwise} \end{cases} \tag{5}$$

$$\text{Conf}_{N+1}(\mathbb{R}^2) \; \underset{\text{whe}}{\simeq} \; B\text{Br}(N+1)\,.$$

Here the non-trivial statement is that the higher homotopy groups $\pi_{k\geq 2}$ are all trivial: That the configuration space is connected ($\pi_0 = *$) is fairly evident; and the (pure) baid group may be understood as being *defined* (7) to be the remaining nontrivial fundamental group π_1.

Somewhat less widely studied (away from the subject of hypergeometric KZ-solutions, Lit. 2.25) but key for our analysis is the fact that these *ordered* configuration spaces are naturally fibered over each other, locally trivially, by maps that forget the last point in a configuration (due to [FaN62, Thm. 3], reviewed as [Bir75, Thm. 1.2], exposition in [Wils18, §2.1]):

$$
\begin{array}{ccc}
\underset{\{1,\cdots,N+2\}}{\text{Conf}}(\mathbb{R}^2) & & \big(z_1,\cdots,z_N,z_{N+1},z_{N+2}\big) \\
\downarrow{\scriptstyle\text{pr}_{N+1}^{N+2}} & & \updownarrow \\
\underset{\{1,\cdots,N+1\}}{\text{Conf}}(\mathbb{R}^1) & & \big(z_1,\cdots,z_N,z_{N+1}\big) \\
\downarrow{\scriptstyle\text{pr}_N^{N+1}} & & \updownarrow \\
\underset{\{1,\cdots,N\}}{\text{Conf}}(\mathbb{R}^1) & & \big(z_1,\cdots,z_N,\big)\,.
\end{array} \tag{6}
$$

On homotopy types this gives the fibration of delooped braid groups (205) which drives the anyon braiding in the final construction (224).

Literature 2.19 (**Anyon species**). The mathematical literature on configuration spaces (Lit 2.18) tends to consider by default the un-ordered configuration space (4) and the corresponding general braid group (Lit. 2.20). But physically realistic defects (*Figure D*) tend to appear in different "species", hence carrying different labels, and are hence described by the *ordered* configuration space (3) and by the corresponding *pure* braid group (7).

In terms of braid representations (Lit. 2.21) this distinction is brought out by the shift of focus in the relevant physics literature to *braided tensor/fusion categories* (see references in [SS22-Ord, Rem. 3.12]), which is one way of encoding that each strand of a braid carries a label (here: of an object in the category).

Specifically for the realistic *monodromy braid representations* arising as the holonomy/transport of KZ-connections on $\mathfrak{su}_2$-conformal blocks (Lit. 2.24) the possible anyon species are indexed by *weights* (in CFT: "highest weights", see [SS22-Def, pp. 8]), namely by natural numbers $w \in \{0,1,\cdots,k\}$ ranging from the trivial case 0 (corresponding to: no defect) up to a fixed "level" $k = \kappa - 2 \in \mathbb{N}$ which controls the resulting braiding phases, see (53) below.

For example (see e.g. [JS21][GHL22] and further references listed in [SS22-Def, §5.2]):
- the popular notion of *Majorana anyons* (alternatively: *Ising anyons*) correspond to $k=2$ ($\kappa = 4$) and their species of weights 0,1,2 are traditionally denoted "1", "σ" and "ψ", respectively.
- the first class of anyons whose braid gates are universal for quantum computation are the "Fibonacci anyons" corresponding to $k=3$ ($\kappa = 5$)[2].
- Beyond that, are "parafermions" and a whole hierarchy of anyon $\mathfrak{su}_2$ classes at higher levels $k \geq 3$ which do not (yet) carry their own names, but which would all be universal for topological quantum computation (see [KMM23]).

---

[2]In comparing to the physics literature, specifically on Fibonacci anyons, beware that for anyons at odd level it is customary to list only the species of even weight.

Literature 2.20 (**Braiding**). Recall (see, e.g., [Mi72]) that by a *group* one means a *group of operations*, on some object, which are associatively composable and invertible (cf. pp. 54). By a *braid group* (due to [Ar25], monographs include [FoN62][Bir75], exposition in [Will20]) one means the group of joint continuous movements of a fixed number $N+1$ of non-coincident points in the plane, from any fixed configuration back to that fixed configuration. The "worldlines" traced out by such points in space-time under such an operation look like a braid with $N+1$ strands, whence the name.

As with actual braids, here it is understood that two such operations are identified if they differ only by continuous deformations of the "strands" without breaking or intersecting these, hence by an *isotopy* in the ambient $\mathbb{R}^3$.
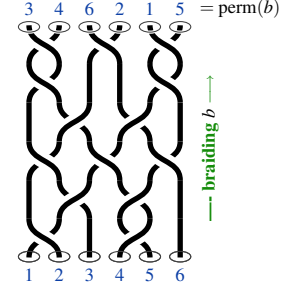
A quick way of saying this with precision (we consider a more explicit description in a moment) is to observe that a braid group is thus the *fundamental group* π_1 (Lit. 2.13) of a configuration space of points in the plane (Lit. 2.18). Here it makes a key difference whether one considers the points in a configuration as *ordered* (labeled by numbers $1, \cdots, N+1$) in which case one speaks of the *pure braid group*, or as indistinguishable (albeit in any case with distinct positions!) in which case one speaks of the *braid group* proper: After traveling along a general braid b the order of the given points may come out permuted by a permutation perm(b), and the braid is called *pure* precisely if this permutation is trivial:

$$
\begin{array}{ccccc}
\substack{\textbf{pure}\\ \textbf{braid group}} & & \textbf{braid group} & & \substack{\textbf{permutation group}\\ \text{("symmmetric group")}} \\
\mathrm{PBr}(N+1) & \xleftarrow{\;\mathrm{fib}_e(\mathrm{perm})\;} \mathrm{Br}(N+1) & \xrightarrow{\;\mathrm{perm}\;} & \mathrm{Sym}(N+1) \\
\Vert & \Vert & & \wr \\
\pi_1\left(\underset{\{1,\cdots,N+1\}}{\mathrm{Conf}}(\mathbb{R}^2)\right) \xrightarrow[\substack{\textbf{forget}\\ \textbf{ordering}}]{} & \pi_1\left(\mathrm{Conf}_{N+1}(\mathbb{R}^2)\right) \xrightarrow[\substack{\textbf{forget}\\ \textbf{positions}}]{} & \pi_1\left(\mathrm{Conf}_{N+1}(\mathbb{R}^\infty)\right) \\
\substack{\textbf{configuration space}\\ \textbf{of ordered points}\\ \textbf{in the plane}} & \substack{\textbf{configuration space}\\ \textbf{of un-ordered points}\\ \textbf{in the plane}} & \substack{\textbf{configuration space}\\ \textbf{of un-ordered points}\\ \textbf{in higher dim Eucl. space}}
\end{array}
\tag{7}
$$

Since these configuration spaces have no other non-trivial homotopy groups (5), the vertical identifications mean equivalently that the homotopy type of these configuration spaces constitute *deloopings* or *classifying spaes* or *Eilenberg-MacLane spaces* in degree 1 (Lit. 2.14) for the braid groups; in particular:

$$
\underset{\{1,\cdots,N+1\}}{\mathrm{Conf}}(\mathbb{R}^1) \underset{\mathrm{whe}}{\simeq} B\,\mathrm{PBr}(N+1) \underset{\mathrm{whe}}{\simeq} K\big(\mathrm{PBr}(N+1),1\big).
\tag{8}
$$

A more explicit way to describe the braid group $\mathrm{Br}(N+1)$ is to observe, first, that any braid may, clearly, be obtained as a composition of those elementary braids which do nothing but pass a pair of neighbouring points past each other:

$$\tag{9}$$

While any braid may be obtained as a composition of just these generators, not every pair of such compositions yields distinct braids. For example, if a pair of such elementary braids acts on disjoint strands, then the order in which they are applied does not matter up to the pertinent continuous deformation of braids:

$$\tag{10}$$

On the other hand, when consecutive triples of elementary braids do act on the same strands, then an evident continuous deformation relates them as follows:

$$(11)$$

A classical theorem due to Artin [Ar25, §3] (cf. [FoN62, §7]) says that these are the *only* relations between these generators, in that the braid group is *presented* by these *generators and relations*, in the general sense of group presentations (e.g. [MKS66][Jo90], cf. pp. 65): [3]

$$\underset{\text{braid group}}{\mathrm{Br}(N+1)} \; \simeq \; \underset{\text{Artin generators (9)}}{\mathrm{FreeGrp}\big(\{e, b_1, \cdots,, b_n\}\big)} \Big/ \left(\begin{array}{l} \overset{\text{Artin braid relations (10) (11)}}{b_i \cdot b_j \;=\; b_j \cdot b_i \;\text{ if }\; i+1 < j} \\ b_i \cdot b_{i+1} \cdot b_i \;=\; b_{i+1} \cdot b_i \cdot b_{i+1} \end{array} \right) \tag{12}$$

Analogously for the pure braid group (7), it is fairly evident that any pure braid can be obtained by composing "weaves" in which one strand lassoes exactly one other strand:

$$\underset{\substack{(i,j)\text{th} \\ \text{generating} \\ \text{pure braid} \\ b_{ij}}}{} :\equiv \; \; \tag{13}$$

For example, consecutive application of such generators for fixed i and decreasing j yields pure braids of the following form:

$$(14)$$

As before in (12), these pure braid generators (13) constitute a finite presentation, now of the pure braid group (we show the optimized set of pure braid relations due to [Le10, Thm. 1.1, Rem. 3.1]):

$$\underset{\substack{\text{pure} \\ \text{braid group}}}{\mathrm{PBr}(N+1)} \; \simeq \; \underset{\text{pure braid generators (13)}}{\mathrm{FreeGrp}\big(\{e\} \sqcup \{b_{ij}\}_{1\leq i<j\leq N+1}\big)} \Big/ \left(\begin{array}{ll} \overset{\text{pure braid relations [Le10, Thm. 1.1]}}{b_{ij} \cdot b_{rs} \;=\; b_{rs} \cdot b_{ij}} & \text{if } r < s < i < j \text{ or } i < r < s < j \\ b_{ji} \cdot b_{ir} \cdot b_{rj} \;=\; b_{ir} \cdot b_{rj} \cdot b_{ji} \;=\; b_{rj} \cdot b_{ji} \cdot b_{ir} & \text{if } r < i < j \\ b_{rs} \cdot \big(b_{jr} \cdot b_{ji} \cdot b_{js}\big) \;=\; \big(b_{jr} \cdot b_{ji} \cdot b_{js}\big) \cdot b_{rs} & \text{if } r < i < s < j \end{array} \right) \tag{15}$$

As before in (10), the first of these relations (15) simply say that the order of applying pure braid generators is irrelevant if these act on disjoint intervals of strands:

---

[3] In (12) we include the neutral element in the set of generators just in order to stick with the convention used in (179) below, where it is most natural to regard the free group-construction as an operation on pointed sets.

$$(16)$$

The further relations in the presentation (15) of the pure braid group concern cases where pure braid generators do "overlap", specifically with the products (14) of other generators:

$$(17)$$

Notice that all these pure braid relations are *commutator* relations [Le10, Rem. 3.10], saying that one pure braid generator commutes with a product of pure braid generators, such as those in (14). This implies that group homomorphisms out of a pure braid group into an *abelian* group are given by assigning any of the abelian group elements to the pure Artin generators (13) (used in Lem. 6.5 below).

Literature 2.21 (**Unitary braid representations**). A unitary *braid re-presentation* (review in [Ab15]) is a linear representation of a (pure) braid group (Lit. 2.20) by unitary operators (on any Hilbert space), hence is a group homomorphism of the form

$$\rho : \mathrm{Br}(N+1) \xrightarrow{\ \ \text{homomorphism}\ \ } \mathrm{U}(k), \qquad \text{or just} \qquad \rho : \mathrm{PBr}(N+1) \xrightarrow{\ \ \text{homomorphism}\ \ } \mathrm{U}(k).$$

In the context of topological quantum computation (Lit. 2.4) one imagines that $\rho(b)$ is the unitary transformation on the ground state of an effectively 2-dimensional topological quantum material (Lit. 2.7) which is induced by adiabatially (Lit. 2.6) moving $N+1$ anyonic (Lit. 2.17) defect points (Lit. 2.18) along the given braid b (Lit. 2.20).

Some authors (e.g. [KL04]) consider representations of the non-pure braid group (7) in the special case that $k = (N+1)d$, with each strand contributing a d-dimensional tensor factor to the full Hilbert space (so that the first Artin braid relation (10) is satisfied), in which case the representation condition for the remaining second Artin braid relation (11) is famous as the *Yang-Baxter equation* (e.g. [YG91][Ab15, §5]), much studied in pure algebra.

Similarly, if the strands (anyons) carry labels (different anyon species, Lit. 2.19) so that one is looking at representations of the *pure* braid group then many authors consider the corresponding algebraic structure of braided fusion categories, going back to [Ki06, §8, §E][NSSFD08, pp. 28][Wa10, §6.3] and repeated in numerous reviews, e.g. [RW18, §2.4.1][Ro22, §2.2].

However, braiding transformations thought to be physically realizable in quantum materials arise more specifically as *monodromy representations*, being the holonomy/transport of the *Knizhnik-Zamolodchikov connection on bundles of conformal blocks* (e.g. [TH01][Ab15, §4][GHL22]), see Lit. 2.24 and see *Figure A*.

Literature 2.22 (**Flat vector bundles, Local systems, Holonomy and Monodromy**). For better or worse, the following terms in pure mathematics are more or less equivalent to each other, their choice of usage depending more on the subject context and author's background than on the underlying phenomenon as such, which is a reflection of the importance of the latter:
- *flat vector bundles*
- *local systems (of coefficients)*
- *holonomy (of flat parallel transport)*
- *monodromy*

To start with, a (topological, Lit. 2.12) *vector bundle* is a *locally trivial fibration of vector spaces*, namely is a continuous map of topological spaces $p_{\mathscr{H}} : \mathscr{H} \to X$ such that over an open cover $\{U_i \subset X\}_{i \in I}$, $\underset{i \in I}{\sqcup} U_i =: U \twoheadrightarrow X$ it looks like a trivial fibration $\mathrm{pr}_U : U \times \mathscr{H}_0 \to U$ of some "typical fiber" vector space $\mathscr{H}_0$, and such that the induced *transition functions* over the cover intersections $U \times_X U \underset{\mathrm{pr}_2}{\overset{\mathrm{pr}_1}{\rightrightarrows}} U$ are continuous and fiberwise linear.

$$\begin{array}{ccc}
U \times \mathscr{H}_0 & \longrightarrow & \mathscr{H} \\
\downarrow{\scriptstyle \mathrm{pr}_U} & {\scriptstyle (\mathrm{pb})} & \downarrow{\scriptstyle p_{\mathscr{H}}} \\
U & \longrightarrow\!\!\!\!\twoheadrightarrow & P
\end{array} \qquad (18)$$

In our application to quantum physics in §3 the base space P is a space of classical *parameters* and the *fiber* $\mathscr{H}_p$ of $p_{\mathscr{H}}$ over p is (or rather: underlies) a *Hilbert space of quantum states* of a quantum system with these external parameters.

Now, a *flat connection* "on" this vector bundle (namely: connecting its fibers to each other) is a rule for how to *lift paths* (Lit. 2.30) from the base space P to the total space $\mathscr{H}$, for every choice of lift of their starting point, such that (1.) these lifts depend only on the homotopy-class of the path (for fixed endpoints) and (2.) respect the concatenation of paths and (3.) lift constant paths to constant paths. Or rather: Such lifting is called the *parallel transport* of a flat connection, that connection itself being understood as the prescription for lifting "infinitesimal paths" (tangent vectors to paths) which one may make sense of when the topological vector bundle is equipped with differentiable structure.

This parallel transport along *closed* paths (representing elements in the fundamental group of the base P, Lit. 2.13) is called the *monodromy* of the flat connection (constituting a linear representation of the fundamental group on the fibers over the base point); and this monodromy equivalently characterizes the flat connection over each connected component of P ([De70, §I.1], cf. e.g. [Di04, Prop. 2.5.1]).

Hence a flat connection on a vector bundle $\mathscr{H} \twoheadrightarrow P$ locally stratifies $\mathscr{H}$ into *horizontal subspaces*, making it a covering space $\mathscr{H}^\delta \twoheadrightarrow P$ (Lit. 2.30). Finally (but here we do not further need this): the *sheaf of local sections* of this covering space $\mathscr{H}^\delta$ is then a *locally constant sheaf of vector spaces* and as such known as a *local system* (terminology going back to [St43], textbook accounts include [Vo03I, §9.2.1][Di04, §2.5]). This terminology is short for *local system of coefficients*, referring to the use of such flat vector bundles as coefficients for *twisted cohomology* (Lit. 2.15).

Literature 2.23 (**Gauss-Manin connections**). In the parameterized perspective on algebraic topology (Lit. 2.12), one is naturally led to consider fiberwise (co)homology groups (Lit. 2.15) over a parameter space. Under suitable conditions, the resulting bundles of (co)homology groups over that base space carry a canonical flat connection (Lit. 2.22), meaning that there is a canonical way of identifying (co)homology elements along small paths of parameter values.

This notion of *Gauss-Manin connections* in a context of algebraic geometry is due to [Ma58] [4]; see also [Gro66][Ka68] [KO68][Gri70]. In §4 we are concerned with the case of twisted cohomology (Lit. 2.15) on topological fiber bundles, following [EFK98, §7.5][Vo03I, Def. 9.13][Ko02, §1.5, 2.1], where the *hyergeometric integral construction* (Lit. 2.25) shows that a special case of Gauss-Manin connections are *Knizhnik-Zamolodchikov connections*.

Literature 2.24 (**Knizhnik-Zamolodchikov connections on conformal blocks**). The Hilbert spaces of topologically ordered ground states of effectively 2-dimensional topological quantum materials (Lit. 2.7) in dependence on the position of anyonic defects (Lit. 2.18) are thought (Lit. 2.4) to be the spaces of states of a Chern-Simons field theory, or rather their "chiral half", called the spaces of "conformal blocks" of a WZW (aka affine current algebra) conformal field theory (see [FMS97, §C]). A derivation of this (previously unproven, cf. [Va21]) assumption from first principles is argued in [SS22-Ord, §3].

As the parameters (the defect configurations) vary, these spaces of conformal blocks form a bundle over the configuration space of points (Lit. 2.18) which carries a canonical flat connection (Lit. 2.22) known as the *Knizhnik-Zamolodchikov connection*, [FMS97, §15.3.2][Ko02, §1.5, 2.1][Ab15, §4], cf. Ex. 4.14.

The monodromy (Lit. 2.22) of this KZ-connection constitutes a unitary representation of the fundamental group of the configuration space of points, hence a braid group representation (Lit. 2.21), thought to reflect the operation of topological quantum gates (Lit. 2.4) by anyon braiding (Lit. 2.9). Here we consider all this via its *hypergeometric integral* representation (Lit. 2.25).

Literature 2.25 (**Hypergeometric integral construction of conformal blocks** [SS22-Def, Prop. 2.15, 2.17]). What is known as the *hypergeometric integral construction* in conformal field theory is, in the end, an equivalence between
  (1) the bundle of conformal blocks with its KZ-connection (Lit 2.24)
  (2) the bundle of suitably twisted cohomology groups (Lit. 2.15) of configuration spaces (Lit. 2.18) of $N + \bullet$ points equipped with their Gauss-Manin connection (Lit. 2.23, Ex. 4.14).

The idea originally emerged from the work of several authors in conformal field theory, an early reference is [DJMM90]; more accessible exposition is given in [EFK98, §4, §7]. The proof that for suitable parameters (53) the resulting KZ-connection is exactly that on $\mathfrak{su}_2$-conformal blocks is due to [FSV94].[5]

That this proof is rather technical (but we tried to bring out the key result concisely in [SS22-Def, §2]) witnesses the complexity inherited from the traditional notion of $\widehat{\mathfrak{su}_2}^k$-conformal blocks and only serves to highlight how remarkable it is that the hypergeometric integral construction reveals them as being equivalent to a purely cohomological construction. It is this remarkable fact which allows us, in the next step, to further identify bundles of conformal blocks with a remarakbly simple homotopy data structure (in Theorem 6.8 below).

Literature 2.26 (**Software verification, data typing and their mathematical semantics**). A profound confluence of computer science and pure mathematics occurs [ML82] with the consideration of *certified* software (e.g. [Ch07]), where the formal verification of the correct implementation of any program (such as by prescribed bounds on numerical rounding errors, cf. Lit. 2.29) happens to coincide with rigorous (and constructive) *proof* of a corresponding mathematical theorem – and vice versa. (Technically, this works due to the *Curry-Howard correspondence*, see around (92) below.) Such verification/proof languages (like `Agda`, Lit. 2.28) are (dependently) *typed* in that strictly every piece of data they handle has assigned a precise *type* which provides the strict specification that data has to meet in order to qualify as input or output of that type ([ML82][Th91][St93][Lu94][Gu95][Co11][Ha16]).

The abstract theory of such data typing is known as (dependent-)*type theory* and the modern flavor relevant here is often called *Martin-Löf type theory* in honor of [ML71][ML75][ML84], for more elaboration and introduction see also [Ho97][UFP13].

Once this typing principle is adhered to, the distinction vanishes between writing a program and verifying its correctness. Moreover, such a properly typed functional program may equivalently be understood as a *mathematical* object, namely as a mathematical function (60) from the "space" of data of its input type to that of its output type — called its *denotational semantics* (a seminal idea due to [Sc70][ScSt71], for exposition see [SK95, §9]) and more specifically, for dependent type theories: its *categorical semantics* in locally Cartesian closed categories (67) [See84][Ho97, §3] (locally Cartesian closed *model* categories for homotopy types) reviewed e.g. in [Ja98][Sh12], or more generally categories with families [Dy96], comprehension categories [Ja93], or natural models [Aw13]. When the univalence axiom (105) is included then this categorical semantics takes place (106) in categories of "higher geometric spaces" (aka: ∞-stacks) called *model toposes* [Re10] (∞-*toposes* [Lu09], see around (107) below), a statement that was first conjectured in [Aw12] and fully proven in [Sh19], exposition is in [Ri22].

---

[4]Yuri Manin hence pioneered both the notion of *quantum computation* (Lit. 2.1) as well as the notion of *Gauss-Manin connections* (Lit 2.23 – Gauss' name appears here just as a general tribute to a historical mathematician). Our result shows that these two seemingly unrelated ideas are in fact closely related.

[5]Beware that these authors equivalently speak in terms of the complexification $\mathfrak{sl}(2, \mathbb{C})$ of the real Lie algebra $\mathfrak{su}_2$.

This tight interrelation between the theories of verified computation of types and of mathematical spaces has been adver-tised as the *Computational Trilogy* (59), references are provided in [SS22-TQC, p. 4]. It is via this "Rosetta stone" of the Computational Trilogy, generalized to homotopy typing (Lit. 2.27), that we translate the algebraic topology of topological quantum gates (in §4) into their homotopy type theory (in §5).

Literature 2.27 (**Homotopically typed programming**). An operation on data so fundamental and commonplace that it is easily taken for granted is the *identification* of one piece of data with another. If the two pieces of data have a simple type — say, they are both numbers — than to identify them means to verify that they are equal. But for data of a more complicated type — say, representing a mathematical structure such as a vector space — then to identify them means to give an *isomorphism* or *equivalence* between them. But taking the idea of program verification by data typing (Lit. 2.26) seriously leads to consideration also of *certificates of identification* between pieces of data of any given type which thus must themselves be data of "identification type" [ML75, §1.7], see around (71) below. In the case of simple types, these certificates may just witness the fact that two pieces of data are equal; but for types of structures, a certificate of identification would be the data of an isomorphism (Lit. 5.1).

Trivial as this may superficially seem, something profound emerges with such "thoroughly typed" programming languages (the technical term is: *intensional type theories*, see [St93, p. 4, 13][Ho95, p. 16], but compare the comment below (105)), in that now given a pair of such identification certificates the same logic applies to themselves and leads to the consideration of identifications-of-identifications (first amplified in [HS98]), and so on to higher identifications, *ad infinitum*.

Remarkably, the "denotational semantics" (Lit. 2.26) of types equipped with such towers of identification types, hence the corresponding pure mathematics, is ([AW09][Aw12], exposition in [Sh12][Ri22]) just that of abstract homotopy theory (Lit. 2.12) where identification types are interpreted (73) as path spaces and higher-order identifications correspond to higher-order homotopies (78). One also expresses this state of affairs, somewhat vaguely, by saying that HoTT has *semantics* in homotopy theory, and conversely that HoTT is a *syntax* for homotopy theory – we review this dictionary in §5.1 below.

Ever since this has been understood, the traditional ("intuitionistic Martin-Löf"-)type theory of [ML75][NPS90] has essentially come to be known as *homotopy type theory* (HoTT) when it satisfies the *univalence* or type extensionality principle: an identification between two types is equivalently an equivalence between them[6] (105). The univalence principle enforces that identification of types coincides with operational equivalence (exposition in [Ac11]).

The standard textbook account for "informal" (human-readable) HoTT is [UFP13], exposition may be found in [BLL13], gentle introduction in [Ri18][Ri23] (the former more extensive); and see §5.1 below. These approaches must add the univa-lence principle to an existing type theory as an axiom, and for this reason cannot be used as programming language. Cubical type theory [CCHM15] adds new primitive rules to type theory in order to give a computational content to the univalence principle. Available software that *runs* homotopically typed programs includes Cubical Agda (Lit. 2.28), which implements cubical type theory and therefore has computational univalence, and Coq[7] (which takes univalence as an axiom(105)).

Literature 2.28 (**The programming/certification language** Agda). The homotopically typed (Lit. 2.27) language Agda[8] is due to [No09], its "cubical" enhancement with computational univalence (105) is due to [VMA19]. For introductions to dependent type theory (Lit. 2.26) in Agda see [St16] and for introduction to homotopy type theory (Lit. 2.27) in Agda see [Es19]. Existing libraries of Agda-code relevant for our discussion include: [1lab][UniMath].

Literature 2.29 (**Exact real computer arithmetic**). One key application of software certification (Lit. 2.26) is to algorithms dealing with *exact real computer arithmetic* ([Vui88][YD95][PE97]), i.e. with operations involving or requiring arbitrary high numerical precision. The basic strategy is to represent a real number by an algorithm which for any prescribed bound on precision produces a rational number that is guaranteed to approximate the intended real number to within that specified bound (cf. Rem. 6.2 below). While traditional *floating-point arithmetic* on computers is notoriously prone to rounding errors and hard to verify, exact real computer arithmetic provides algorithms that provably satisfy prescribed accuracy bounds. The underlying mathematics of exact real computer arithmetic is that of *constructive analysis* [Bish67][BB85][Br99] (which has been fully developed already decades ago, then mainly motivated on philosophical grounds, as an alternative to classical analysis), and naturally implemented in typed programming languages (e.g. [O'C07][GNSW07]) such as in Agda (Lit. 2.28): [Mu22][Lu15], cf. pp. 72.

While the application of real analysis to the simulation of classical systems in physics and engineering needs no further emphasis, the simulation of quantum computational systems involves real analysis yet more fundamentally, in that the vector spaces (Hilbert spaces) of quantum states involve arbitrary-precision data even where the corresponding classical state spaces form a finite set (cf. *Table D*). Specifically, in quantum computation (Lit. 2.1) the analog of a classical computation on a

---

[6] The univalence principle is widely attributed to [Vo10], but the idea (under a different name) is actually due to [HS98, §5.4], there however formulated with respect to a subtly incorrect type of equivalences (as later shown in [UFP13, Thm. 4.1.3]). The new contribution of [Vo10, p. 8, 10] was a good definition of the types (103) of ("weak") equivalences between types.

[7] Coq landing page: coq.inria.fr

[8] Agda landing page: wiki.portal.chalmers.se/agda

finite set of *bits* is a (unitary) *linear operator* on a finite-dimensional complex vector space (such as spaces of *conformal blocks* in the case of anyons, Lit. 2.17), whose certification, as such, already involves arbitrary precision arithmetic — see the discussion below in §6.

In fact, the core issue of quantum computation, namely, the *compilation* of a set of prescribed *quantum logic gates* to a *quantum circuit* which evaluates to a prescribed unitary operator is one that must be understood in the sense of exact real arithmetic (constructive analysis) as to be verified for any finite bound on precision (cf. p. 3).

Literature 2.30 (**Path lifting and fiber transport**). The archetypical phenomenon which leads over from "point-set topology" to genuine homotopy theory (Lit. 2.12) is the classification of covering spaces by the "transport" operation induced on their fibers via "path lifting" (e.g. [tD08, §3][Mø11]).

The *fundamental theorem of covering spaces* (e.g. [tD08, Thm. 3.3.2][Mø11, 7.8]) shows that covering spaces are equivalently classified by this transport action of the fundamental group $\pi_1(P)$ (Lit. 2.13) of their base spaces P given by such lifting of closed paths in the base space. A major class of examples of this situation arises from vector bundles $\mathscr{X} \twoheadrightarrow P$ equipped with flat connections ("local systems", Lit. 2.22): The flat connection locally stratifies $\mathscr{H}$ into "horizontal subspaces" making it a covering space $\mathscr{H}^\delta \twoheadrightarrow P$. The corresponding path lifting is the "parallel transport" induced by the flat connection.

(More generally, one calls a map $p : X \to B$ of topological spaces a *Serre fibration* if for every Euclidean family of continuous paths in the "base space" B and for every lift of the corresponding family of starting points through p to X, there is also a compatible lift of the entire family of paths to X. If one thinks of these lifted paths as lines along which their starting points "flow" to their endpoints, then such lifts again serve to "functorially transport" the fibers of p along paths in B (e.g. [tD08, §5.6]) in a homotopy-correct way.

Noticing here that every (discrete) group G arises as the fundamental group of *some* topological space, and specifically of its Eilenberg-MacLane spaces $BG = K(G,1)$ (Lit. 2.14), this means in our context that all systems of *reversible* logic gates (Lit. 2.2) acting on the same space of (memory-)states is equivalently embodied as the transport operation induced by path lifting in some covering space! This perspective we expand on in §3.

Curiously, such path lifting/fiber transport is a *native* language construct (74) in homotopy-typed programming languages (Lit. 2.27) such as Agda (Lit. 2.28). This remarkable fact drives our discussion in §5 and §6 below, as highlighted now in §3.
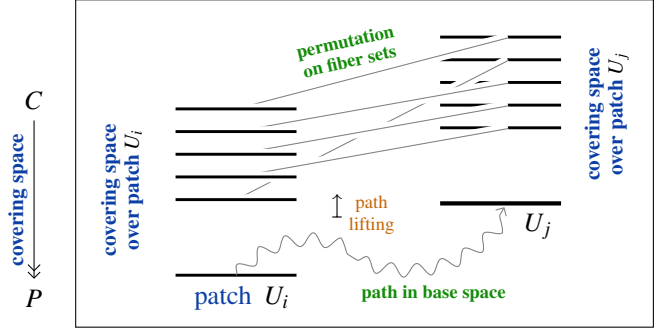
3   Topological quantum gates...

While the broad idea of topological quantum logic gates is now 25 years old and classical (Lit. 2.4), its conceptual fine-print has arguably remained somewhat elusive all along, both regarding their physical realization (cf. Lit. 2.8) as well as their information-theoretic nature (Lit. 2.9). In this section, we provide a quick modernized review and explanation of the basic idea of topological quantum gates, informed by their cohomological realization ([SS22-Ord], Lit. 2.16) and such as to bring out their secret nature [SS22-TQC] as a fundamental construction in parameterized point-set topology (which is developed below in §4) and in homotopy type theory (developed below in §5):

The driving theme of our discussion is the observation/claim that:

Fundamental (quantum) computing processes are lifts of classical parameter paths, i.e. the programs, to (linear) maps of state spaces, i.e. the (quantum) gates.

(19)

This may sound simple, but we claim it is profound (similar statements are in [ZR99][NDGD06][DN08][LW17]): Namely, it means that natural certification languages for low hardware-level (quantum) computation ought to natively know about *path lifting* (Lit. 2.30). This is unheard-of in traditional programming languages — but it is the hallmark (74) of homotopically-typed languages! (Lit. 2.27) Moreover, under such identification of low-level computation with path-lifting, homotopically-typed languages natively reflect the crucial reversibility (76) of fundamental quantum computational processes (Lit. 2.2).

The idea of quantum circuits. For example, a *quantum circuit* (Lit. 2.1) is *de facto* the compilation of a sequence of basic linear operators U_p (the "gates") following a sequence of computing instructions p. Choosing different classical paths of instructions yields different quantum algorithms in a controlled way, and this dependency of the quantum transport on classical instruction paths is what it means for quantum computations to be programmable (the *circuit model* of quantum computation [De89], e.g. [NC10, §II.4]).
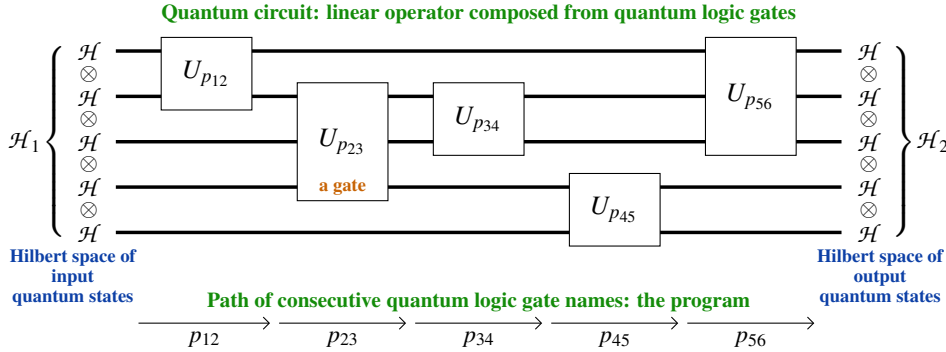
Figure QC. A quantum logic circuit acting on a compound of quantum systems (each shown as a solid line), compiled by composing a sequence of quantum logic gates (shown as boxes).
Such quantum circuits are naturally understood as compound functions on *linear* types, we expand on this in the companion articles [SS23-QM][SS23-EoS]. Here we focus on the internal operation of single (topological quantum) gates.

However, in real laboratory implementations of quantum computers, the idealization of discretized instruction steps is necessarily approximated by actual physical processes which, however abrupt they may appear, are fundamentally continuous. This means that all *real* quantum computational processes depend on *continuous paths* in a classical parameter space.

For example, for many quantum computing architectures, such as for the original *spin resonance* principle (see, e.g., [Co00][Eq22]) as well as for the contemporary *superconducting qbits* (e.g., [CW08][HWFZ20]) it is the case that to operate a quantum gate means to send an electromagnetic pulse of a finite duration through the system (see, e.g., [GIB12] and [MFL22], respectively): Therefore, the hardware-level instructions of such quantum computers are continuous parameter paths in the configuration space of the ambient electromagnetic field, quite of the following schematic form:
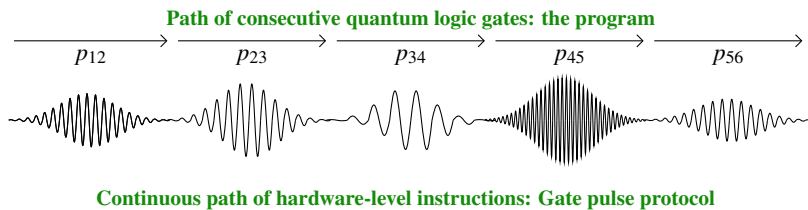
Figure P. The execution of a quantum algorithm on common types of NISQ machines (Lit. 2.3), like the currently popular *superconducting qbit* architectures, corresponds to subjecting the system to a *gate pulse protocol* of radio/microwave modulation (schematically shown on the left) hence to driving it along a *continuous path* in the configuration space of external parameters (here: the electromagnetic cavity field amplitude).

This is to amplify that *all* quantum-computation is fundamentally an evolution of quantum states controlled by continuous classical parameter paths. Now the case of *topological* quantum computation corresponds to the special case where these continuous parameter paths are (1.) adiabatic and their effect on the system states is (2.) homotopy-invariant:

The idea of adiabatic quantum transport. A traditional computing process of the form indicated in *Figure P* exchanges energy between the quantum system and its control environment. In fact, common NISQ architectures (Lit. 2.3) are designed to encode qbit states as energy eigenstates of anharmonic quantum oscillators, so that passing between their energy levels (the notorious *quantum jumps*) is what it means to execute computations on these systems in the first place. At the same time, these energetic interaction channels with their environment is what makes these NISQ machines suffer from noise and decoherence.

In contrast, a topological quantum process (Lit. 2.4) is, first of all, to take place entirely on the (topologically ordered) *ground states* of a topological quantum material (Lit. 2.7), hence on their lowest energy states, without absorbing any energy from the environment: The notorious *energy gap* which measures the fidelity of topological phases of matter (Lit. 2.7), separating their topological ground states from their ordinary excited states, is the room within which the control environment may shed energy without disturbing the coherent quantum phase.

This state of affairs is neatly captured by one of the classical theorems of mathematical quantum mechanics: The *Quantum Adiabatic Theorem* (Lit. 2.6) says (as nicely brought out for quantum computation already in [ZR99]) that in the asymptotic limit of sufficiently gentle (= "adiabatic") movement of external classical parameters, the induced quantum system's evolution asymptotically preserves gapped energy eigen-states, hence in particular preserves gapped ground states, and hence acts on the Hilbert space $\mathcal{H}$ of gapped ground states by unitary operators U_p that vary continuously with the parameter path p.

Figure T. Schematically shown on the left is the "adiabatic" (Lit. 2.6) transport of quantum states along linear maps depending on continuous paths in a classical parameter space.
The diagram on the right indicates our description of such situations by (linear) homotopy type families depending on a base homotopy type, as explained in §5 below (see (74) and (106) below, noticing that we relegate discussion of *linearity* of quantum types to [SS23-QM][SS23-EoS]).
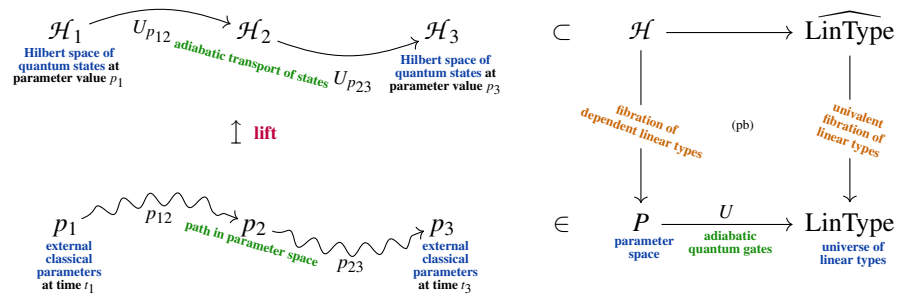

The idea of quantum annealing. For example, a widely-known implementation of the above *Quantum Adiabatic Theorem* in quantum computation is the paradigm of "quantum annealing" [KN98][FGGS00] (review in [RSDC22]). Here one considers a single-parameter path linearly interpolating between two given Hamiltonians, H_0, H_1 on a fixed Hilbert space $\mathcal{H} :\equiv \mathcal{H}_1 = \mathcal{H}_2$:

If one can arrange for both Hamiltonians to have unique gapped ground states and for the ground state of the first Hamiltonian to be preparable, while that of the second Hamiltonian is unknown but identifiable with the answer to a given computational problem, then the corresponding adiabatic quantum transport effectively computes that answer.

Due to its restriction to finding a unique ground state, quantum annealing as such is not a universally programmable form of computation: It is guaranteed to discover the ground state of H_1 and does nothing else. However, in this restrictivity annealing does foreshadow a key aspect of topological quantum computation in degenerate form (cf. Lit. 2.6): Since the ground state of H_1 is unique, the annealing process does not actually depend on the exact parameter path chosen to arrive there, it is *robust against perturbations of the computational path* (cf. [CFP02, p. 2]).

The idea of topological quantum computation. Generally, the profound practical problem with implementing the theoretically straightforward idea of programmable quantum processes (Lit. 2.1) is that real quantum machines are not in idealized isolation but are coupled to their environment, which necessarily acts like a "thermal bath": Inevitable noise in the environment causes perturbations that tend to de-cohere the machine's quantum state and thus tend to destroy its intended quantum computation (cf. Lit. 2.5).

Concretely, parameter paths realizable in real laboratories are noisy (cf. Lit. 2.3), hence are drawn randomly from an ensemble of small perturbations of the intended path. The result of transporting a pure quantum state along such a noisy ensemble is in general a decohered mixture of pure states which may no longer support the quantum interference effects on which quantum algorithms crucially rely; unless, that is, one could somehow guarantee that the quantum transport depending on these paths is actually *in*dependent of their small perturbations, really depending only on the global properties of these paths.

This is the idea of topological quantum computation (Lit. 2.4): to ensure that the quantum adiabatic process depends only on the *topological homotopy classes* (Lit. 2.12, 2.13) of the parameter paths (relative to their endpoints).
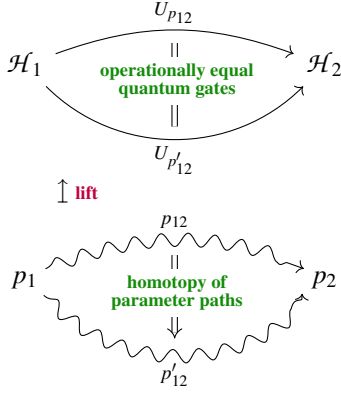
Figure H. An adiabatic quantum transport (Fig. T) is *topological* (or rather: *homotopical*, cf. Lit. 2.12) if it depends on the parameter path between fixed endpoints only up to small continuous deformations, namely up to homotopy (indicated on the left).

When this is the case, then quantum transport depends *robustly* on "global" properties of parameter paths, such as their winding number (cf. Lit. 2.13) around "holes" in parameter space (schematically indicated on the right) and hence constitutes a form of *topological quantum computation* (Lit. 2.4).
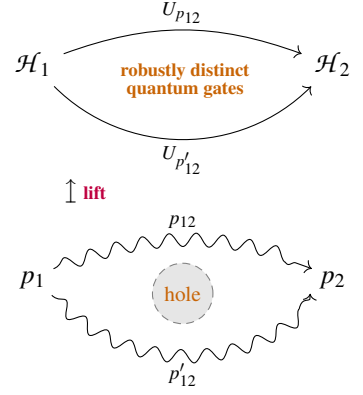
The topological quantum computer scientist is thus led to search for topological quantum materials which are dependent on classical parameter spaces that have a rich structure of "holes" in them, namely with a rich *fundamental group* (Lit. 2.13).

The idea of anyon braid grates. There could be several possible choices for such topological quantum systems (cf. [ZR99]), but the original proposal by Kitaev (Lit. 2.1) may be the most promising and has come to often be treated as synonymous with topological quantum computation as such. Here one imagines that a quantum material's gapped and topologically ordered ground state (Lit. 2.7) depends topologically on tuples $(z_1, \cdots, z_n)$ of pairwise distinct positions of *defect points* ("anyons", Lit. 2.17) which are effectively constrained to move inside a surface Σ (such as for a crystalline material consisting of a few monolayers of atoms).

For example, much attention has been focused on the idea that such defects might be realized by quantum vortices in the surfaces of quantum fluids, such as certain Bose-Einstein condensates (e.g. [MPSS19]). The defect parameters could also be more abstract, such as being the critical "nodal" values (not of positions but) of *momenta* of electrons in topological semimetals (Lit. 2.10). Such nodal momentum values typically vary with fairly easily controllable external parameters such as external strain exerted on the material's crystal structure.

In any case, in such a situation the classical parameter space P is effectively the *configuration space of points* (Lit. 2.18) in the surface Σ

$$\underset{\{1, \cdots, N\}}{\mathrm{Conf}} (\Sigma) = \left\{ (z_1, \cdots, z_N) \in \Sigma \,\Big|\, \underset{I \neq J}{\forall} \, z_I \neq z_J \right\}.$$

A *path* in such a space is an *n*-tuple of "worldlines" of defects which may move around each other but never coincide (at any given instant of time), thus forming the appearance of a "braid" of *n* strands in 3d space (Lit. 2.20).

This implies that even if Σ is taken to be topologically trivial (e.g. Σ could be the disk through the equator of a Bose-Einstein condensate) there are still plenty of distinct homotopy classes of paths in $\underset{\{1, \cdots, N\}}{\mathrm{Conf}} (\Sigma)$, corresponding to all those braids which cannot be untied. If a quantum material can be found whose degenerate ground states are transported topologically but non-trivially along such braidings of defect points, then this would realize topological quantum computation by *anyon braid gates* in the original sense of Freedman and Kitaev (Lit. 2.4).

Figure A. If the classical parameter space of a dependent quantum system (*Figure T*) is a configuration space (Lit. 2.18) of (anyonic) defect points (Lit. 2.17) in a plane, then a parameter path is a *braid* (Lit. 2.20).

If the (degenerate) ground state of a topological quantum system depends topologically on the defect poisitions (Lit. 2.7), then their adiabatic transport along such braid paths realizes quantum gates that form a linear representation of the braid group.

In the technologically viable situation of $\mathfrak{su}_2$-anyons, this is the *monodromy representation* of the canonical flat KZ-connection on the *bundle of conformal block* over configuration (Lit. 2.24).
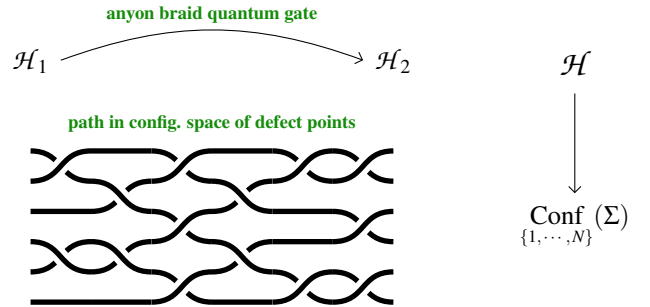
The idea of certified braid gate types. But none of this intricate internal structure of topological braid quantum gates is visible to existing quantum programming languages; and any traditional implementation of this information (via conformal quantum field theory methods) would be formidable to construct and then inefficient to use. Yet, detailed verification (Lit. 2.26) of the operation and compilation of these braid gates will arguably be crucial for practical scalable quantum computation (Lit. 2.5), and will serve for topological quantum simulation already now.

Our claim here is that this problem finds an elegant solution by regarding it through the novel lens of homotopically-typed programming languages (Lit. 2.27), where the construction of types certifying braid quantum gate operation magically turns out to be essentially a native language construct (Thm. 6.8 below). This we explain now.

4 ...via parameterized point-set topology

Here we show that the construction of Gauss-Manin connections (Lit. 2.23) on bundles of fiberwise cohomology groups (Lit. 2.15) has an elementary description in the context of parameterized homotopy theory (Lit. 2.12): this is the content of Thm. 4.9 for the untwisted case and Thm. 4.13 in the broader twisted case.

This re-formulation is so elementary (in the technical sense) that it becomes effectively a tautology when formulated in the corresponding formal language of homotopy type theory (which we turn to in §5): The covering space which exhibits the flat Gauss-Manin connection is simply the fiberwise 0-truncation of the *fiberwise* mapping space (the internal hom-object in the slice) into the corresponding classifying space. This applies at once in the generality of twisted generalized and/or non-abelian cohomology theories, such as twisted K-theory and twisted Cohomotopy. Applied to twisted complex cohomology groups of configuration spaces of points in the plane, it yields the Knizhnik-Zamolodchikov (KZ) connection on bundles of $\widehat{\mathfrak{su}_2}^k$-conformal blocks (Lit. 2.24), and thus the monodromy braid representation characteristic of $\mathfrak{su}_2$-anyons (Lit. 2.20).

Here we use (just the most basic aspects of) parameterized "point-set" homotopy theory (as laid out in [MS06], going back to [Bo70]) in order to show that Gauss-Manin connections in (twisted) generalized cohomology groups on fibers of bundles are exhibited by the fiberwise 0th Postnikov stage (127) of the fiberwise mapping space (fiberwise space of sections) into the given classifying space (classifying fibration).

With the relevant notions and results from parameterized point-set homotopy theory in hand, the proof is straightforward, so we use the occasion to briefly introduce and review basics of parameterized topology as we go along, in order to make the proof reasonably self-contained also for a general mathematical audience.

The construction in itself of the Gauss-Manin connection on fiberwise twisted cohomology groups of locally trivial fiber bundles may be understood without the abstract machinery invoked here; a sketch of such a more low-brow argument is what [EFK98, §7.5] offers. However, it is our abstract re-formulation that provides an elegant handle on Gauss-Manin connections in the language of homotopy type theory, which is discussed in §5 below.

Below we make repeated use of the *pasting law* (in one direction): In any category with pullbacks, the pullback along a composite morphism is the pasting of the pullbacks along the two factors (e.g. [AHS90, Prop. 11.10], see also [GPP21, Prop. 8]):

$$
\begin{array}{ccc}
(f_2 \circ f_1)^* X \longrightarrow X & & f_2^*(f_1^* X) \longrightarrow f_1^* X \longrightarrow X \\
\downarrow \quad \text{(pb)} \quad \downarrow p_X & \simeq & \downarrow \quad \text{(pb)} \quad \downarrow \quad \text{(pb)} \quad \downarrow \\
B'' \xrightarrow{f_1 \circ f_2} B & & B'' \xrightarrow{f_2} B' \xrightarrow{f_1} B
\end{array}
\tag{20}
$$

4.1 For ordinary & generalized cohomology

Given a sufficiently nice fibration $p_X : X \to B$, the ordinary complex cohomology groups $H^n(X_b) :\equiv H^n(X_b; \mathbb{C})$ of its fibers X_b for $b \in B$, form a bundle of abelian groups over B equipped with a flat connection, known as the *Gauss-Manin connection* (Lit. 4.13). This means that a Gauss-Manin connection provides a rule for coherently transporting cohomology classes of spaces X_b as these spaces *vary with a parameter* $b \in B$. Moreover, the flatness of the connection means that the induced transport of cohomology classes depends on parameter paths $\gamma : [0,1] \to B$ only via their homotopy classes $[\gamma]$ relative to their endpoints. If B is connected, this means equivalently that the Gauss-Manin connection is a homomorphism from the fundamental group of B to the automorphism group of $H^n(X_{b_\bullet}; A)$ at any fixed $b_\bullet$.

In fact, this applies also to *twisted* cohomology groups, in which case the Knizhnik-Zamolodchikov connection becomes a special case. We come back to this in a moment (§4.2).

Traditionally, Gauss-Manin connections are constructed algebraically. Here we work entirely homotopy-theoretically and instead make use of the fact that twisted ordinary cohomology of a topological space of CW-type is *representable* (eg. [FSS20-Cha, §1]), in that for $n \in \mathbb{N}$ there exists a topological space $\mathrm{K}(n, \mathbb{C})$ (the *n*th *Eilenberg-MacLane space*) such that cohomology is identified with the connected components of the *mapping space* into it:

$$H^n(X_b) \simeq \pi_0 \mathrm{Map}(X_b, \mathrm{K}(\mathbb{C}, n)). \tag{22}$$

In fact, also generalized cohomology theories are represented by respective classifying spaces (26), and the following construction of their Gaus-Manin connection works uniformly for all these cases by considering the (fiberwise) *mapping spaces* into these classifying spaces:

Mapping spaces. For mapping spaces to work well we may assume without practical restriction that we work in the category

$$\mathrm{kTopSpc} \hookrightarrow \mathrm{TopSpc} \tag{23}$$

of compactly generated spaces ("k-spaces", see [SS21-Bun, Nota. 1.0.15] for pointers). Here, for $X_b \in \mathrm{kTopSpc}$, we have an *adjunction* of the following form (the cartesian tensor/hom-adjunction, e.g. [Bo94II, §7.1, §7.2][Br93, §VII.2]):

$$\mathrm{kTopSpc} \xleftarrow[\underset{\mathrm{Map}(X, -)}{\overset{X \times (-)}{\perp}}]{} \mathrm{kTopSpc}, \tag{24}$$

meaning that there is an *exponential law* for k-topological spaces, namely a natural bijection of maps of the following form:

$$\mathrm{kTopSpc}(X \times Y, Z) \xrightarrow{\ \sim\ } \mathrm{kTopSpc}(X, \mathrm{Map}(Y, Z))$$

$$\left(X \times Y \xrightarrow{(x,y) \mapsto f(x,y)} Z\right) \ \mapsto \ \left(X \xrightarrow{x \mapsto (y \mapsto f(x,y))} \mathrm{Map}(Y, Z)\right). \tag{25}$$

Incidentally, in formal category theory, it is tradition to denote such adjointness relations by displaying generic pairs of *adjunct maps* separated by a horizontal line:

$$\frac{X \times Y \xrightarrow{\ f\ } Z}{X \xrightarrow[\tilde{f}]{} \mathrm{Map}(Y, Z).}$$

This in turn alludes to an old tradition in formal logic of denotig *natural deduction*-steps this way [Ge34] (see [Ge69]): Here we may read this as saying that "Given a map $f : X \times Y \to Z$ we may deduce a map $\tilde{f} : X \to \mathrm{Map}(Y, Z)$, and vice versa". This is more than an analogy, it is the first glimpse of the syntax-semantics relation between (algebraic) topology and (dependent) type theory, which we invoke in §5.1, see around (66).

Generalized cohomology. The following construction of the Gauss-Manin connection over fiber bundles relies only on the existence of a classifying space as in (22), but not on its concrete nature. This means that the construction applies also to "generalized cohomology theories" (cf. [FSS20-Cha, §1]).

- If, for instance, $E^\bullet(-)$ is a Whitehead-generalized cohomology theory, such as topological K-theory, elliptic cohomology or cobordism cohomology, then there exists a *spectrum* of classifying spaces $\{E_n\}_{n \in \mathbb{N}}$ such that

$$E^n(X_b) \simeq \pi_0 \mathrm{Map}(X_b, E_n).$$

- Regarded the other way around, for *any* topological space $\mathscr{A} \in \mathrm{kTopSpc}$ we may regard

$$\mathscr{A}^0(X_b) := \pi_0 \mathrm{Map}(X_b, \mathscr{A}) \tag{26}$$

  as *non-abelian generalized cohomology* with coefficients in A.

- For example, if $\mathscr{A} = BG$ is the classifying space of a discrete or compact Lie group G, then

$$(BG)^0(X_b) \equiv H^1(X_b; G)$$

  is, equivalently, the traditional non-abelian cohomology in degree 1 with coefficients in G, which classifies G-principal bundles.

- Or if $A = S^n \subset \mathbb{R}^{n+1}$ is the topological *n*-sphere, then

$$(S^n)^0(X_b) \simeq \pi^n(X_b)$$

  is unstable *Cohomotopy*.

The fiberwise mapping space. The key fact now is that in *parameterized homotopy theory* (e.g. [CJ98][BM21][MS06]) the mapping space construction (24) generalizes to *slices* if the base space B is (compactly generated and) Hausdorff, which we assume from now on:

$$B \in \mathrm{kHausSpc} \hookrightarrow \mathrm{kTopSpc} \, . \tag{27}$$

Here the *slice category* $\mathrm{kTopSpc}_{/B}$ is the category whose objects (X, p_X) are k-topological space X equipped with a continuous map $p_X : X \to B$ and whose morphisms $(X, p_X) \to (Y, p_Y)$ are compatible maps $X \to Y$, hence:

$$\mathrm{kTopSpc}_{/B}\big((X, p_X), (Y, p_Y)\big) \; = \; \mathrm{kTopSpc}\big(X, Y\big) \underset{\mathrm{kTopSpc}(X, B)}{\times} \{p_X\} \longrightarrow \mathrm{kTopSpc}\big(X, Y\big)$$

$$\big\downarrow \qquad \qquad (\mathrm{pb}) \qquad \qquad \big\downarrow {\scriptstyle p_Y \circ (-)} \tag{28}$$

$$\{p_X\} \lhook\joinrel\longrightarrow \mathrm{kTopSpc}\big(X, B\big) \, .$$

If we understand X as an object in the slice category $\mathrm{kTopSpc}_{/B}$ over B via p_X (21) and if we denote by $p_B^* A$ the trivial fiber bundle over B with fiber A regarded in the slice category, then their *fiberwise mapping space* is a topological space which is itself fibered over B, such that the fibers of the fiberwise mapping space are the ordinary mapping spaces (24) on the fibers (cf. Prop. 4.4 below):

Ordinary mapping space on fiber   Fiberwise mapping space (itself a topological space over B)

$$\mathrm{Map}\big(X_b, \mathscr{A}\big) \longrightarrow \mathrm{Map}\big((X, p_X), p_B^* \mathscr{A}\big) .$$

$$\big\downarrow \qquad \qquad (\mathrm{pb}) \qquad \qquad \big\downarrow \tag{29}$$

$$* \overset{b}{\lhook\joinrel\longrightarrow} B$$

Parameter space

The right choice of topology on the total fiberwise mapping space is subtle[9] and the result that such a topology exists (see [MS06, §1.3.7-§1.3.9], following [BB78, Thm. 3.5]) may be regarded as the engine which powers our slick re-construction of Gauss-Manin connections from the point of view of point-set topology. Here the right topology is that which ensures the sliced analog of the adjunction (24):

$$\mathrm{kTopSpc}_{/B} \; \overset{(X, p_X) \times (-)}{\underset{\mathrm{Map}((X, p_X), -)}{\overset{\perp}{\longleftarrow\joinrel\longrightarrow}}} \; \mathrm{kTopSpc}_{/B} \, , \tag{30}$$

hence, equivalently, the *exponential law* in the slice [BB78, Thm. 3.5][MS06, (1.3.9)]: a natural bijection of the form

Fiberwise product space   Fiberwise mapping space

$$\mathrm{kTopSpc}_{/B}\big((X, p_X) \times (Y, p_Y), (Z, p_Z)\big) \; \simeq \; \mathrm{kTopSpc}_{/B}\big((X, p_X), \mathrm{Map}\big((Y, p_Y), (Z, p_Z)\big)\big) , \tag{31}$$

where now the product on the right is that in the slice, hence is the *fiber product* of k-topological spaces:

$$(X, p_X) \times (Y, p_Y) \; \simeq \; \big(X \times_B Y, p_X \circ \mathrm{pr}_X = p_Y \circ \mathrm{pr}_Y\big) .$$

Of course, the unit for this product is the identity map on the base space:

$$(X, p_X) \times (B, \mathrm{id}_B) \; \simeq \; (X, p_X) . \tag{32}$$

This exponential law in slices implies a wealth of useful structure:

Proposition 4.1 (Base change adjoint triple). *For any map between base spaces* $f : B \to B'$ *(27), there is a "base change" adjoint triple*

$$\mathrm{kTopSpc}_{/B} \; \overset{\overset{f_!}{\overset{\perp}{\longleftarrow f^* \longrightarrow}}}{\underset{f_*}{\underset{\perp}{\longrightarrow}}} \; \mathrm{kTopSpc}_{/B'} \, , \tag{33}$$

where f^* *denotes the pullback operation (formed in* $\mathrm{kTopSpc}$*), its left adjoint* $f_!$ *is given by postcomposition with* f*, and the right adjoint* f_* *is given by the following pullback construction:*

$$f_*(X, p_X) \longrightarrow \mathrm{Map}\big((B, f), (X, f \circ p_X)\big)$$

$$\big\downarrow \qquad \qquad (\mathrm{pb}) \qquad \qquad \big\downarrow \tag{34}$$

$$(B', \mathrm{id}_{B'}) \overset{\widetilde{\mathrm{id}}}{\longrightarrow} \mathrm{Map}\big((B, f), (B, f)\big) \, .$$

---

[9][MS06, p. 15]: "*The point-set topology of parametrized spaces is surprisingly subtle. Parametrized mapping spaces are especially delicate.*"

Proof. For the left adjoint $(f_! \dashv f^*)$ the required hom-isomorphism is immediate from the universal property of the pullback:

For the right adjoint, the required hom-isomorphism is obtained as the following sequence of natural isomorphisms:

$$\mathrm{kTopSpc}_{/\mathrm{B}'}\big((U, p_\mathrm{U}), f_*(\mathrm{X}, p_\mathrm{X})\big)$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{B}'}\Big((U, p_\mathrm{U}), \mathrm{Map}\big((\mathrm{B}, f), (\mathrm{X}, f \circ p_\mathrm{X})\big) \underset{\mathrm{Map}\big((\mathrm{B},f),(\mathrm{B},f)\big)}{\times} \{\widetilde{\mathrm{id}}\}\Big) \qquad \text{by (34)}$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{B}'}\Big((U, p_\mathrm{U}), \mathrm{Map}\big((\mathrm{B}, f), (\mathrm{X}, f \circ p_\mathrm{X})\big)\Big) \underset{\mathrm{Map}\big((U,p_\mathrm{U}),\mathrm{Map}\big((\mathrm{B},f),(\mathrm{B},f)\big)\big)}{\times} \{\widetilde{\mathrm{id}}\} \quad \text{by (28)}$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{B}'}\Big((U, p_\mathrm{U}) \times (\mathrm{B}, f), (\mathrm{X}, f \circ p_\mathrm{X})\Big) \underset{\mathrm{kTopSpc}_{/\mathrm{B}'}\big((U,p_\mathrm{U})\times(\mathrm{B},f), (\mathrm{B},f)\big)}{\times} \{\widetilde{\mathrm{id}}\} \qquad \text{by (31)}$$

$$\simeq \Big(\mathrm{kTopSpc}(f^*U, \mathrm{X}) \underset{\mathrm{kTopSpc}(f^*U, \mathrm{B}')}{\times} *\Big) \underset{\big(\mathrm{kTopSpc}(f^*U, \mathrm{B}) \underset{\mathrm{kTopSpc}(f^*U,\mathrm{B}')}{\times} *\big)}{\times} \{\widetilde{\mathrm{id}}\} \qquad \text{by (28)}$$

$$\simeq \mathrm{Map}\big(f^*U, \mathrm{X}\big) \underset{\mathrm{Map}(f^*U, \mathrm{B})}{\times} * \qquad \text{by (35)}$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{B}'}\big(f^*(U, p_\mathrm{U}), (\mathrm{X}, p_\mathrm{X})\big) \qquad \text{by (28).}$$

Here the penultimate step is observing that the fiber products (limits) may be interchanged: Instead of computing the horizontal fiber product of the vertical fiber product in the following diagram, we may first compute the horizontal fiber products (shown on the right, again by (28)):

$$(35)$$

Therefore the evident remaining vertical fiber product is as claimed. □

Example 4.2 (**Space of sections**). When $\mathrm{B}' = *$ in (33), the right base change (34) constructs *spaces of sections*:

$$\Gamma_{\mathrm{X}_b}(-) \simeq (p_{\mathrm{X}_b})_* : \ \mathrm{kTopSpc}_{/\mathrm{X}_b} \longrightarrow \mathrm{kTopSpc}. \qquad (36)$$

Generally, one may understand the right base change as forming *fiberwise spaces of sections*.

Proposition 4.3 (**Cartesian Frobenius reciprocity**). *For $f : \mathrm{B} \to \mathrm{B}'$ a map of base spaces (27), we have a natural isomorphism*

$$f_!\big((\mathrm{X}, p_\mathrm{X}) \times f^*(\mathrm{Y}, p_\mathrm{Y})\big) \simeq \big(f_!(\mathrm{X}, p_\mathrm{X})\big) \times (\mathrm{Y}, p_\mathrm{Y}), \qquad (37)$$

where $(f_! \dashv f^*)$ is the left base change adjunction (33).

Proof. This follows by the pasting law (20), which here says that the following pullback squares in kTopSpc agree:

□

26

In generalization of (29), we have:

Proposition 4.4 (**Base change is closed functor**). *The pullback (base change) of a fiberwise mapping space along any continuous map of base spaces* (27) $f : B' \to B$ *is the fiberwise mapping space of the pullbacks of the arguments:*

$$f^*\mathrm{Map}\big((X, p_X), (Y, p_Y)\big) \;\simeq\; \mathrm{Map}\big(f^*(X, p_X), f^*(Y, p_Y)\big). \tag{38}$$

Proof. Apply the Yoneda Lemma over $\mathrm{kTopSpc}^{\mathrm{op}}_{/B}$ to the following sequence of natural isomorphisms:

$$
\begin{aligned}
\mathrm{kTopSpc}_{/B'}\Big((U, p_U), f^*\mathrm{Map}\big((X, p_X), (Y, p_Y)\big)\Big) &\;\simeq\; \mathrm{kTopSpc}_{/B}\Big(f_!(U, p_U), \mathrm{Map}\big((X, p_X), (Y, p_Y)\big)\Big) &&\text{by (33)} \\
&\;\simeq\; \mathrm{kTopSpc}_{/B}\Big(\big((f_!(U, p_U)) \times (X, p_X), (Y, p_Y)\big)\Big) &&\text{by (31)} \\
&\;\simeq\; \mathrm{kTopSpc}_{/B}\Big(f_!\big((U, p_U) \times f^*(X, p_X)\big), (Y, p_Y)\Big) &&\text{by (37)} \\
&\;\simeq\; \mathrm{kTopSpc}_{/B'}\Big((U, p_U) \times f^*(X, p_X), f^*(Y, p_Y)\Big) &&\text{by (33)} \\
&\;\simeq\; \mathrm{kTopSpc}_{/B'}\Big((U, p_U), \mathrm{Map}\big(f^*(X, p_X), f^*(Y, p_Y)\big)\Big) &&\text{by (31) .} \quad\square
\end{aligned}
$$

For the case at hand, this has the following consequence:

Proposition 4.5 (**Fiberwise mapping space out of fiber bundle is fiber bundle**). *Let* $(X, p_X) \in \mathrm{kTopSpc}_{/B}$ *be a fiber bundle with local trivialization*

$$\phi : U \xrightarrow[\mathrm{opn}]{} B, \qquad \text{such that} \qquad \phi^*(X, p_X) \simeq p_U^* X_0 \equiv U \times X_0. \tag{39}$$

Then the fiberwise mapping space out of (X, p_X) *into any* $\mathscr{A} \in \mathrm{kTopSpc}$ *is a fiber bundle that locally trivializes over the same open cover:*
$$\phi^*\mathrm{Map}\big((X, p_X), p_B^*\mathscr{A}\big) \simeq p_U^*\mathrm{Map}\big(X_0, \mathscr{A}\big). \tag{40}$$

Proof.

$$
\begin{aligned}
\phi^*\mathrm{Map}\big((X, p_X), p_B^*\mathscr{A}\big) &\;\simeq\; \mathrm{Map}\big(\phi^*(X, p_X), \phi^* p_B^*\mathscr{A}\big) &&\text{by (38)} \\
&\;\simeq\; \mathrm{Map}\big(p_U^* X_0, p_U^*\mathscr{A}\big) &&\text{by (39)} \\
&\;\simeq\; p_U^*\mathrm{Map}\big(X_0, \mathscr{A}\big) &&\text{by (38).} \quad\square
\end{aligned}
$$

Fiberwise homotopy theory. Moreover, the topology on the fiberwise mapping space (29) is also "homotopy correct" in that its map to the base B is an h-fibration as soon as p_X is an h-fibration (by [Bo70, §6.1], see also [MS06, Prop. 1.3.11]), which is the case for B a metrizable space and p_X a fiber bundle. This implies that the fibers of the fiberwise mapping space are in fact homotopy fibers, and that forming path ∞-groupoids (singular simplicial complexes, cf. [SS21-Bun, Pro. 3.3.43]),

$$\mathrm{kTopSpc} \xleftarrow[\mathrm{Pth}]{\overset{|-|}{\underset{\perp}{\longrightarrow}}} \Delta\mathrm{Set}$$

respects this property:

$$
\begin{array}{ccc}
\mathrm{Pth}\,\mathrm{Map}\big(X_b, A\big) & \longrightarrow & \mathrm{Pth}\,\mathrm{Map}\big((X, p_X), p_B^* A\big). \\
\downarrow & {\scriptstyle (\mathrm{pb})} & \downarrow {\scriptstyle \in \mathrm{KanFib}} \\
* & \xrightarrow{\quad b \quad} & \mathrm{Pth}(B)
\end{array}
\tag{41}
$$

Lemma 4.6 (**Fiberwise truncation is preserved by base change**). *For any map of simplicial sets* $f : B' \to B$, *the operation of base change (pullack)* f^* *of Kan fibrations preserves fiberwise Postnikov truncation* (127):

$$f^* \circ \pi_{0/B} \;\simeq\; \pi_{0/B'} \circ f^*.$$

Proof. It is useful to understand this as a special case of a general phenomenon of 0-*truncation* in slice homotopy theories [Lu09, §5.5.6]. Every morphism $p : X \to S$ factors uniquely through $\pi_{0/S}(E)$ as a "0-connected" map followed by a "0-truncated map", and both these classes are preserved by homotopy pullback ([Lu09, Ex. 5.2.8.16 & Lem. 6.5.1.16(6)]). Therefore, the claim follows by the pasting law (20), which also holds for homotopy pullbacks ([Lu09, Lem. 4.4.2.1][GPP21, Prop. 8]):

$$
\begin{array}{ccc}
f^*X & \longrightarrow & X \\
{\scriptstyle 0\text{-cnctd}}\downarrow & {\scriptstyle \text{(hpb)}} & \downarrow{\scriptstyle 0\text{-cnctd}} \\
\pi_{0/B'}(f^*X) & \longrightarrow & \pi_{0/B}(X) \\
{\scriptstyle 0\text{-cnctd}}\downarrow & {\scriptstyle \text{(hpb)}} & \downarrow{\scriptstyle 0\text{-trnctd}} \\
B' & \xrightarrow{\ f\ } & B\,.
\end{array}
\tag{42}
$$

Therefore, applying Lemma 4.6 to the diagram (41), we obtain a homotopy pullback diagram as shown on the left here:

$$
\begin{array}{ccccc}
\overset{\substack{\textit{A}\textbf{-cohomology}\\ \textbf{of fiber}}}{A^0(X_b)} \simeq \pi_0\big(\operatorname{Pth}\operatorname{Map}(X_b,A)\big) & \longrightarrow & \overset{\substack{\textbf{Fiberwise 0-truncation of }(\text{path }\infty\text{-groupoid of)}\textbf{ fiberwise mapping space}}}{\pi_{0/\operatorname{Pth}(B)}\big(\operatorname{Pth}\operatorname{Map}\big((X,p_X),p_B^*A\big)\big)} & \longrightarrow & \operatorname{Set}^{*/} \\
\downarrow & {\scriptstyle\text{(hpb)}} & \downarrow & {\scriptstyle\text{(hpb)}} & \downarrow{\substack{\textbf{Covering space}\\ \textbf{classifier}}} \\
\{b\} & \longrightarrow & \underset{\substack{\textbf{Fundamental groupoid}\\ \textbf{of base space}}}{\operatorname{Pth}(B)} & \xrightarrow{\ \nabla^{\operatorname{GM}}_{X,A}\ } & \operatorname{Set}\,.
\end{array}
\tag{43}
$$

The left square shows that the fiberwise 0-truncation of the fiberwise mapping space is a fibration over the fundamental groupoid of B, whose (homotopy) fibers are the generalized cohomology sets (26) of the fiber space X_b. The homotopy pullback shown on the right follows by:

Lemma 4.7 (**Univalent universe of sets**). *Any homotopy fibration of sets, as in the middle of* (43)*, is classified by – i.e., is the homotopy pullback along – an essentially unique map $\nabla^{\operatorname{GM}}_{X,A}$ to the covering space classifier, as shown in the square on the right of* (43)*.*

Proof. This may be understood as a simple special case of the general fact that ∞-groupoids form an ∞-topos in which there exists a "small fibration classifier" $\operatorname{Grpd}^{*/}_\infty \longrightarrow \operatorname{Grpd}_\infty$ ([Lu09, Prop. 3.3.2.7][Ci19, §5.2][KL21]). $\qquad\square$

Remark 4.8 (**Flat connections as functors on the fundamental groupoid**). Noticing that $\operatorname{Pth}(B)$ is equivalently the disjoint union over connected components $[b] \in \pi_0(B)$ of delooping groupoids $\mathbf{B}\pi_1(B,b)$, this map $\nabla^{\operatorname{GM}}_{X,A}$ (43) is over each connected component equivalently a group homomorphism

$$
\Omega_b\big(\nabla^{\operatorname{GM}_{X,A}}\big) : \pi_1(B,b) \longrightarrow \operatorname{Aut}\big(A^0(X_b)\big)\,.
$$

This is a traditional incarnation of flat connections on a space B (e.g. [De70, §I.1][Di04, Prop. 2.5.1], see Lit. 2.22).

Moreover, from Prop. 4.5 it follows that this local system of sets trivializes over any cover over which p_X trivializes, so that it corresponds to a *covering space*, which we denote as follows:

$$
\underset{\substack{\forall\\ b\in B}}{}\quad
\begin{array}{ccc}
\overset{\substack{\textbf{A-cohomology}\\ \textbf{of fiber}}}{A^0(X_b)} = \overset{\substack{\textbf{Connected components of}\\ \textbf{ordinary mapping space on fiber}}}{\pi_0\operatorname{Map}(X_b,A)} & \longrightarrow & \overset{\substack{\textbf{Parameterized connected components of}\\ \textbf{fiberwise mapping space (a covering space over }B)}}{\pi_{0/B}\operatorname{Map}\big((X,p_X),p_B^*A\big)}\,. \\
\downarrow & {\scriptstyle\text{(pb)}} & \downarrow \\
* & \xleftarrow{\ b\ } & \underset{\textbf{Parameter space}}{B}
\end{array}
\tag{44}
$$

Using this, comparison with [Vo03I, Def. 9.13] readily shows that the flat connection $\nabla^{\operatorname{GM}}_{X,A}$ in (43) is indeed the Gauss-Manin connection. In conclusion, we have shown so far:

Theorem 4.9 (**Gauss-Manin connection in generalized cohomology over fiber bundles via fiberwise mapping spaces**). *Let $B \in \operatorname{kHaus}$ be a metrizable Hausdorff space* (27) *and $(X,p_X) \in \operatorname{kTopSpc}_{/B}$ be a locally trivial fiber bundle whose typical fiber admits the structure of a CW-complex. Then for any $A \in \operatorname{kTopSpc}$ the Gauss-Manin-connection on the A-cohomology sets* (26) *of the fibers X_b is exhibited (under Lem. 4.7) by the fiberwise 0-truncation of the fiberwise mapping space* (30) *from X into A:*

$$
\underset{\substack{\textbf{Gauss-Manin}\\ \textbf{connection on}\\ \textbf{A-cohomology}}}{\nabla^{\operatorname{GM}}_{X,A}} \quad\underset{\text{Lem. 4.7}}{\longleftrightarrow}\quad \underset{\substack{\textbf{Fiberwise 0-truncation of fiberwise mapping space into }A}}{\pi_{0/\operatorname{Pth}(B)}\big(\operatorname{Pth}\operatorname{Map}\big((X,p_X),p_B^*A\big)\big)}\,.
$$

4.2 For twisted generalized cohomology

We generalize the above discussion to the case of fiberwise *twisted* cohomology (Thm. 4.13) and bring out the motivating example of the $\widehat{\mathfrak{su}_2}^k$-Knizhnik-Zamolodchikov equation (Ex. 4.14).

In the existing literature, this is discussed for the special case when the total space X is equipped with a flat complex line bundle $\mathscr{L}$ classified by a map $\tau : X \to B\mathrm{U}(1)^\flat$ into the classifying space of the *discrete* group underlying the circle group. In this case one may consider the τ_b-twisted complex ordinary cohomology of the fibers, namely the cohomology with coefficients in the local system (Lit. 2.22) of flat local sections of $\mathscr{L}$ (e.g. [Vo03II, §5.1.1]):

$$H^{n+\tau_b}(X_b; \mathbb{C}) = H^n(X_b; \mathscr{L}).$$

At least when $p_X : X \to B$ is a fiber bundle, these twisted cohomology groups again carry a flat Gauss-Manin connection, as we explain now.

In the example where $X = \underset{\{1,\cdots,n+N\}}{\mathrm{Conf}}(\mathbb{R}^2)$ is a configuration space of points in the plane equipped with a certain local system, and where p_X the map that forgets the first n of $n+N$ points, then a *hyprgeometric integral construction* (Lit. 2.25) identifies this Gauss-Manin-connection on fiberwise twisted complex cohomology with a Knizhnik-Zamolodchikov connection ([EFK98, §7.5], see Ex. 4.14 below).

This is the main example of interest for the present purpose. However, in [SS22-Def][SS22-Ord] we explained that it is useful to regard this twisted ordinary cohomology as the home of the twisted Chern characters of twisted *K-theory* groups. For this reason and since the theory is best understood in this generality, we consider now Gauss-Manin connections on bundles of *twisted generalized* cohomology groups, where the situation of (21) is generalized to the following picture:
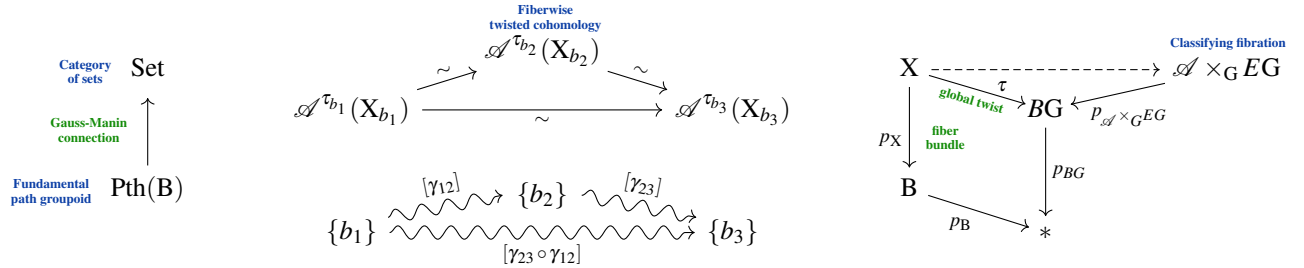
Twisted generalized non-abelian cohomology. In generalization of (26), we have:

Definition 4.10 (**Twisted generalized non-abelian cohomology** [FSS20-Cha, §2.2][SS20-Orb, Rem. 2.94])**.** For
- $G \in \mathrm{Grp}(k\mathrm{TopSpc})$ a topological group;
- $G \curvearrowright \mathscr{A} \in G\,\mathrm{Act}(k\mathrm{TopSpc})$ a topological G-space,
- with $\left(\mathscr{A} \times_G EG, p_{\mathscr{A} \times_G EG}\right) \in k\mathrm{TopSpc}_{/BG}$ its Borel construction;
- $\tau_b : X_b \to BG$ a continuous map;

we say that

$$
\begin{aligned}
\mathscr{A}^{\tau_b}(X_b) \ &:= \ H^{\tau_b}\left(X_b; \mathscr{A}\right) \\
&:= \ \pi_0\,\Gamma_X\left((\tau_b)^*(\mathscr{A} \times_G EG)\right) \\
&\underset{(46)}{\simeq} \ (p_{BG})_*\mathrm{Map}\Big(\underbrace{(X_b, \tau_b)}_{(\tau_b)_!(X_b, \mathrm{id}_{X_b})}, (\mathscr{A} \times_G EG, p_{\mathscr{A} \times_G EG})\Big)
\end{aligned}
\tag{45}
$$

is the τ_b-*twisted A-cohomology* of X_b.

To see the identification shown in (45), of the space of sections with a right base change of the fiberwise mapping space over the classifying space of twists, apply the Yoneda Lemma to the following sequence of natural bijections:

$$\mathrm{kTopSpc}\Big(\mathrm{U}, (p_{BG})_* \mathrm{Map}\big((\tau_b)_!(\mathrm{X}_b, \mathrm{id}_{\mathrm{X}_b}), (\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\big)\Big)$$

$$\simeq \mathrm{kTopSpc}_{/B\mathrm{G}}\Big((p_{BG})^*\mathrm{U}, \mathrm{Map}\big((\tau_b)_!(\mathrm{X}_b, \mathrm{id}_{\mathrm{X}_b}), (\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\big)\Big) \qquad \text{by (33)}$$

$$\simeq \mathrm{kTopSpc}_{/B\mathrm{G}}\Big(\big((p_{BG})^*\mathrm{U}\big) \times (\tau_b)_!(\mathrm{X}_b, \mathrm{id}_{\mathrm{X}_b}), (\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\Big) \qquad \text{by (31)}$$

$$\simeq \mathrm{kTopSpc}_{/B\mathrm{G}}\Big((\tau_b)_!\big((p_{\mathrm{X}_b})^*\mathrm{U} \times (\mathrm{X}_b, \mathrm{id}_{\mathrm{X}_b})\big), (\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\Big) \qquad \text{by (37)}$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{X}_b}\Big((p_{\mathrm{X}_b})^*\mathrm{U} \times (\mathrm{X}_b, \mathrm{id}_{\mathrm{X}_b}), (\tau_b)^*(\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\Big) \qquad \text{by (33)}$$

$$\simeq \mathrm{kTopSpc}_{/\mathrm{X}_b}\Big((p_{\mathrm{X}_b})^*\mathrm{U}, (\tau_b)^*(\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\Big) \qquad \text{by (32)}$$

$$\simeq \mathrm{kTopSpc}\Big(\mathrm{U}, (p_{\mathrm{X}_b})_*\big((\tau_b)^*(\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\big)\Big) \qquad \text{by (33)}$$

$$\simeq \mathrm{kTopSpc}\Big(\mathrm{U}, \Gamma_{\mathrm{X}_b}\big((\tau_b)^*(\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathscr{A} \times_{\mathrm{G}} E\mathrm{G}})\big)\Big) \qquad \text{by (36)}.$$

(46)

Example 4.11 (Twisted ordinary complex cohomology with coefficients in a local system). For each $n \in \mathbb{N}$, the canonical multiplication action of $\mathrm{U}(1)^\flat \subset (\mathbb{C}^\times)^\flat$ (of the discrete circle group on the group of units in the field $\mathbb{C}$ regarded with its discrte topology) induces an action on the nth Eilenberg-MacLane space $\mathrm{U}(1) \curvearrowright \mathrm{K}(\mathbb{C}, n)$. For $\tau_b : \mathrm{Pth}(\mathrm{X}_b) \to \mathbf{B}\mathrm{U}(1)^\flat$ the classifying map of a flat connection (Rem. 4.8) on a complex line bundle (i.e., on the connected component $[b]$ a group homomorphism $\pi_1(\mathrm{B}, b) \to \mathrm{U}(1)$), the corresponding twisted cohomology according to Def. 4.10 is the traditional cohomology with coefficients in the local system $\mathscr{L}(\tau)$ of parallel sections of this flat connection (e.g. [Vo03I, §5.1.1]):

$$\mathrm{K}(\mathbb{C}, n)^\tau(\mathrm{X}_b) = H^{n+\tau}(\mathrm{X}_b; \mathbb{C}) \simeq H^n\big(\mathrm{X}_b; \mathscr{L}(\tau)\big).$$

Now given $\mathrm{G} \curvearrowright \mathrm{A}$, as in Def. 4.10, and a fibration $(\mathrm{X}, p_{\mathrm{X}}) \in \mathrm{kTopSpc}_{/\mathrm{B}}$ as before in §4.1, consider a choice of *global twist*, namely a continuous map

$$\tau : \mathrm{X} \longrightarrow B\mathrm{G}.$$

Via this twist, we may regard X as fibered over the product space $\mathrm{B} \times B\mathrm{G}$, whence its fibers with respect to B are themselves still fibered over $B\mathrm{G}$:

$$\big(\mathrm{X}, (p_{\mathrm{X}}, \tau)\big) \in \mathrm{kTopSpc}_{/\mathrm{B} \times B\mathrm{G}} \qquad \Rightarrow \qquad \underset{b \in \mathrm{B}}{\forall} \ (i_b \times \mathrm{id}_{B\mathrm{G}})^*\big(\mathrm{X}, (p_{\mathrm{X}}, \tau)\big) = (\mathrm{X}_b, \tau_b) = (\tau_b)_! \mathrm{X}_b \ \in \mathrm{kTopSpc}_{/B\mathrm{G}}.$$

Forming the fiberwise mapping space (30) in this sense, we obtain the following twisted generalization of (29):

<div align="center">
Fiberwise mapping space over classifying space of twists   Fiberwise mapping space (itself a topological space over B × BG)
</div>

$$\mathrm{Map}\big((\mathrm{X}_b, \tau_b), (\mathrm{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathrm{A} \times_{\mathrm{G}} E\mathrm{G}})\big) \longrightarrow \mathrm{Map}\big((\mathrm{X}, (p_{\mathrm{X}}, \tau)), p_{\mathrm{B}}^*(\mathrm{A} \times_{\mathrm{G}} E\mathrm{G}, p_{\mathrm{A} \times_{\mathrm{G}} E\mathrm{G}})\big).$$

$$\downarrow \qquad\qquad\qquad\qquad\qquad\qquad (\mathrm{pb}) \qquad\qquad\qquad\qquad\qquad\qquad \downarrow$$

$$\{b\} \times B\mathrm{G} \xleftarrow{\hspace{2cm}} \underset{i_b \times \mathrm{id}_{B\mathrm{G}}}{\hspace{0.5cm}} \xrightarrow{\hspace{2cm}} \mathrm{B} \times B\mathrm{G}$$

(47)

<div align="center">
Classifying space of twists   Space of parameters and twists
</div>

In order to turn this into a pullback diagram over just B we need the *Beck-Chevalley relation* (see [Pav91, §1][Ba15, §7.5] and [Sc14a, Def. 5.5][GL17, §2.4.1]):

Proposition 4.12 (Cartesian Beck-Chevalley property). *Given a fiber product diagram in* $\mathrm{kTopSpc}$ *as shown on the left below, the possible composite base changes* (33) *through the diagram are naturally isomorphic as shown on the right:*

$$
\begin{array}{c}
\mathrm{X} \times_{\mathrm{B}} \mathrm{Y} \\
\overset{\mathrm{pr_X}}{\swarrow} \qquad \overset{\mathrm{pr_Y}}{\searrow} \\
\mathrm{X} \qquad (\mathrm{pb}) \qquad \mathrm{Y} \\
\underset{p_{\mathrm{X}}}{\searrow} \qquad \underset{p_{\mathrm{Y}}}{\swarrow} \\
\mathrm{B}
\end{array}
\qquad \Rightarrow \qquad
\begin{cases}
(p_{\mathrm{Y}})^* \circ (p_{\mathrm{X}})_! \simeq (\mathrm{pr_Y})_! \circ (\mathrm{pr_X})^*, \\
(p_{\mathrm{X}})^* \circ (p_{\mathrm{Y}})_* \simeq (\mathrm{pr_X})_* \circ (\mathrm{pr_Y})^*.
\end{cases}
$$

(48)

Proof. The first isomorphism in (48) follows immediately from the pasting law (20), which for $(\mathrm{E}, p_{\mathrm{E}}) \in \mathrm{kTopSpc}_{/\mathrm{X}}$ gives the following natural identification:

$$
\begin{array}{ccc}
(p_{\mathrm{Y}})^*(p_{\mathrm{X}})_!\mathrm{E} \longrightarrow \mathrm{Y} & & \overset{p_{(\mathrm{pr_Y})_!(\mathrm{pr_X})^*\mathrm{E}}}{\overbrace{\hspace{3cm}}} \\
\downarrow \quad (\mathrm{pb}) \quad \downarrow {\scriptstyle p_{\mathrm{Y}}} & \simeq & (\mathrm{pr_X})^*\mathrm{E} \longrightarrow \mathrm{X} \times_{\mathrm{B}} \mathrm{Y} \xrightarrow{\mathrm{pr_Y}} \mathrm{Y} \\
\mathrm{E} \xrightarrow{p_{\mathrm{X}} \circ p_{\mathrm{E}}} \mathrm{B} & & \downarrow (\mathrm{pb}) \ {\scriptstyle\mathrm{pr_X}} \downarrow (\mathrm{pb}) \downarrow {\scriptstyle p_{\mathrm{Y}}} \\
& & \mathrm{E} \xrightarrow{p_{\mathrm{E}}} \mathrm{X} \xrightarrow{p_{\mathrm{X}}} \mathrm{B}.
\end{array}
$$

This implies the second natural isomorphism by adjointness (33) and the Yoneda Lemma:

$$\mathrm{kTopSpc}_{/X}\big((U,p_U),(p_X)^*(p_Y)_*(E,p_E)\big) \simeq \mathrm{kTopSpc}_{/B}\big((p_X)_!(U,p_U),(p_Y)_*(E,p_E)\big)$$
$$\simeq \mathrm{kTopSpc}_{/Y}\big((p_Y)^*(p_X)_!(U,p_U),(E,p_E)\big),$$

and similarly for the other side of the isomorphism. $\qquad\square$

Now specialize this Beck-Chevalley relation (48) to the following case:

$$\begin{array}{c}
\xymatrix{
& BG \ar[dl]_{p_{BG}} \ar[r]^{i_B \times \mathrm{id}_{BG}} & \\
* & \text{(pb)} & B \times BG, \\
& B
}
\end{array} \qquad \big(i_b\big)^* \circ \big(\mathrm{id}_B \times p_{BG}\big)_* \;\simeq\; \big(p_{BG}\big)_* \circ \big(i_b \times \mathrm{id}_{BG}\big)^*. \tag{49}$$

This implies, from (47), the following pullback diagram:

$$\begin{array}{ccc}
\overset{\text{Space of sections over fiber}}{(p_{BG})_*\mathrm{Map}\big((X_b,\tau_b),(A \times_G EG, P_{A\times_G EG})\big)} & \longrightarrow & \overset{\text{Fiberwise space of sections (itself a topological space over B)}}{(\mathrm{id}_B \times p_{BG})_*\mathrm{Map}\big((X,p_X),p_B^*A \times_G EG\big).} \\
\downarrow & \text{(pb)} & \downarrow \\
\{b\} & \overset{i_b}{\longrightarrow} & \underset{\substack{\text{Space of} \\ \text{parameters} \\ \text{and twists}}}{B}
\end{array} \tag{50}$$

Since the classifying space BG – in its construction due to Milgram: $BG = |N(G \rightrightarrows *)|$ (recalled, e.g., in [SS21-Bun, (2.64)]) – is a CW-complex and hence Serre-cofibrant, the map on the right is still a Serre fibration, so that passing to parameterized connected components works as before in (43) to yield a covering space, generalizing (44), whose fiber over $b \in B$ is the τ_b-twisted A-cohomology (Def. 4.10) of the fiber X_b:

$$\begin{array}{ccc}
\overset{\substack{\tau\text{-twisted} \\ A\text{-cohomology} \\ \text{of fiber}}}{A^\tau(X_b)} \simeq \overset{\text{Connected components of space of sections over fiber}}{\pi_0\Big((p_{BG})_*\mathrm{Map}\big((X_b,\tau_b),(A \times_G EG, P_{A\times_G EG})\big)\Big)} & \rightarrow & \overset{\substack{\text{Fiberwise connected components of} \\ \text{fiberwise space of sections (itself a topological space over B)}}}{\pi_{0/B}\Big((\mathrm{id}_B \times p_B)_*\mathrm{Map}\big((X,(p_X,\tau)),p_B^*A \times_G EG\big)\Big).} \\
\downarrow & \text{(pb)} & \downarrow \\
* & \overset{i_b}{\longrightarrow} & \underset{\substack{\text{Space of} \\ \text{parameters} \\ \text{and twists}}}{B}
\end{array} \tag{51}$$

As before in the untwisted case, Prop. 4.5 again implies that this covering space trivializes compatibly with any local trivialization of (X,p_X), thus exhibiting its corresponding classifying map $\nabla^{\mathrm{GM}}_{X,G \wedge A}$ (via Lem. 4.7) as the Gauss-Manin connection (cf. the description in [EFK98, §7.5]).

In conclusion, we have now shown the following generalization of Thm. 4.9 to twisted cohomology:

Theorem 4.13 (Gauss-Manin connection in twisted generalized cohomology over fiber bundles via fiberwise mapping spaces). *Let* $B \in \mathrm{kHaus}$ *be a metrizable space and* $(X,p_X) \in \mathrm{kTopSpc}_{/B}$ *be a locally trivial fiber bundle whose typical fiber admits the structure of a CW-complex.*

Then, for a group $G \in \mathrm{Grp}(\mathrm{kTopSpc})$ *and local coefficients* $G \wedge \mathscr{A} \in G\,\mathrm{Act}(\mathrm{kTopSpc})$, *the Gauss-Manin-connection on the twisted* $\mathscr{A}$-*cohomology sets* (45) *of the fibers* X_b *is exhibited by the fiberwise 0-truncation of the right base change along* BG (50) *of the fiberwise mapping space* (47) *from* X *into the Borel construction* $\mathscr{A} \times_G EG$:

$$\underset{\substack{\textbf{Gauss-Manin} \\ \textbf{connection on} \\ \textbf{twisted } \mathscr{A}\textbf{-cohomology}}}{\nabla^{\mathrm{GM}}_{X,G \wedge \mathscr{A},\tau}} \quad \overset{\longleftrightarrow}{\underset{\text{Lem. 4.7}}{}} \quad \underset{\substack{\textbf{Fiberwise 0-truncation of right base change along } BG \\ \textbf{of fiberwise mapping space into Borel construction on } A}}{\pi_{0/\mathrm{Pth}(B)}\Big(\mathrm{Pth}\,(\mathrm{id}_B \times p_{BG})_*\mathrm{Map}\big((X,(p_X,\tau)),p_B^*\mathscr{A} \times_G EG\big)\Big).} \tag{52}$$

In the next section §5 we explain how the homotopy-theoretic construction on the right of (52) has a slick implementation in homotopy-typed programming languages such as `Agda`: This is Def. 5.16 below.

But first to make explicit the special case of these general considerations that is of interest for quantum computation:

Example 4.14 (**The Knizhnik-Zamolodchikov connection of $\widehat{\mathfrak{su}_2}^k$-conformal blocks via parameterized homotopy theory**). Consider the specialization of the above setup to the following choice of domain fibration, local coefficients and twist, parameterized by (cf. Lit. 2.19)

$$
\begin{array}{|ll|l|}
\hline
N & \in & \mathbb{N}_{\geq 1} & \text{Number of "defect" points} \\
n & \in & \mathbb{N} & \text{Number of "probe" points} \\
\kappa & \in & \mathbb{N}_{\geq 2} & \text{The "shifted level"} \\
(\mathrm{w}_I)_{I=1}^N \in \{0,1,\cdots,\kappa-2\}^N & & \text{The "weights" carried by the defects} \\
\hline
\end{array}
\tag{53}
$$

as follows:

(i) The **domain fibration**

$$
\left(\mathrm{X} \xrightarrow{p_X} \mathrm{B}\right) :\equiv \left(\underset{\{1,\cdots,N+n\}}{\mathrm{Conf}}(\mathbb{C}) \xrightarrow{\mathrm{pr}_N^{N+n}} \underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C})\right) \simeq \left(B\mathrm{PBr}(N+n) \to B\mathrm{PBr}(N)\right)
\tag{54}
$$

is the fibration (6) of configuration spaces of ordered points in the plane (Lit. 2.18) which forgets the last n of $N+n$ points; to be regarded as the equivalent fibration of delooped pure braid groups (Lit. 2.20) as shown on the right (205).

(ii) The **local coefficient space**

$$
G \wr \mathscr{A} :\equiv \mathbb{C}^\times \wr K(\mathbb{C},n)
\tag{55}
$$

is the complex Eilenberg-MacLane space from Ex. 4.11, in degree n and equipped with its canonical action by the discrete group of units $\mathbb{C}^\times$;

(iii) The **twist**

$$
\tau_{(\kappa,(\mathrm{w}_I)_{I=1}^N)} : \underset{\{1,\cdots,N+n\}}{\mathrm{Conf}}(\mathbb{C}) \simeq B\mathrm{PBr}(N+n) \longrightarrow B\mathbb{C}^\times ,
\tag{56}
$$

is the delooping of the following group homomorphism:

$$
\begin{array}{rcl}
\mathrm{PBr}(N+n) & \xrightarrow{\ \Omega\tau\ } & \mathbb{C}^\times \\[4pt]
b_{I,i} & \longmapsto & \exp\left(2\pi\mathrm{i}\,\frac{\mathrm{w}_I}{\kappa}\right) \\[4pt]
b_{i,j} & \longmapsto & \exp\left(2\pi\mathrm{i}\,\frac{2}{\kappa}\right) \quad \text{for} \quad \begin{array}{l} 1 \leq I \leq N \\ N < i \leq N+n. \end{array} \\[4pt]
b_{I,J} & \longmapsto & \exp\left(2\pi\mathrm{i}\,\frac{\mathrm{w}_I\mathrm{w}_J}{2\kappa}\right)
\end{array}
\tag{57}
$$

Here "b_{ij}" denote the pure braid generators (13). Notice that any such assignment respects the pure braid relations (15) because these are all group commutator relations which are all trivially satisfied in an abelian group such as $\mathbb{C}^\times$.

With these choices, the Gauss-Manin connection (52) specializes to the situation discussed in [EFK98, §7.5] and reviewed in our context in [SS22-Def][SS22-Ord][SS22-TQC], yielding the Gauss-Manin connection on the bundle of fiberwise twisted ordinary cohomology groups (Ex. 4.11)

$$
\left(\{z_I\}_{I=1}^N \in \underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C})\right) \longmapsto H^n\left(\underset{\{1,\cdots,n\}}{\mathrm{Conf}}\left(\mathbb{C}\setminus\{z_I\}_{I=1}^N\right); \mathscr{L}\left(\tau_{(\kappa,(\mathrm{w}_I)_{I=1}^N)}\right)\right)
$$

By [EFK98, §7.5] this is the Knizhnik-Zamolodchikov connection (Lit. 2.24) on — by [FSV94] — the spaces of $\widehat{\mathfrak{su}_2}^{\kappa-2}$ conformal blocks for $N+1$-point correlators on the Riemann sphere with weights $\mathrm{w}_1,\cdots,\mathrm{w}_N$ as specified and $\mathrm{w}_{N+1} = n + \sum_{I=1}^N \mathrm{w}_I$. This is the result of the *hypergeometric integral construction* (Lit. 2.25) of KZ-solutions further reviewed (and referenced) in [SS22-Def][SS22-Ord].

(Notice that the choice of phase for b_{IJ} in the last line of (57), follows [EFK98, (7.14)][Ko12, (6.1)]. Making a different choice here only results in tensoring the resulting flat connection by a flat line bundle, changing the resulting monodromy by these global phases.)

In conclusion, in the case of this Example 4.14, Theorem 4.13 with Example 4.11 says that th KZ-connection on the space of $\widehat{\mathfrak{su}_2}^{\kappa-2}$-conformal blocks is realized as equivalently reflected in the fiberwise 0-truncation of (a right base change of) a

fiberwise mapping space:

$$\nabla^{\mathrm{KZ}}\Big(\kappa, (\mathrm{w}_I)_{I=1}^N, n\Big)$$

$$\updownarrow \quad \text{Lem. 4.7}$$

$$\pi_{0/\mathrm{Pth}\,\underset{\{1,\cdots,N\}}{\mathrm{Conf}}\,(\mathbb{C})} \left(\mathrm{Pth}\left(\mathrm{id}_{\underset{\{1,\cdots,N\}}{\mathrm{Conf}}\,(\mathbb{C})} \times p_{B\mathbb{Z}_\kappa}\right)_* \mathrm{Map}\left(\begin{array}{ccc} \underset{\{1,\cdots,N+n\}}{\mathrm{Conf}}\,(\mathbb{C}) & & \mathrm{K}(\mathbb{C},n) \times_{\mathbb{C}^\times} E\mathbb{C}^\times \\ {\scriptstyle(\mathrm{pr}_N^{N+n},\,\tau_{(\kappa,\mathrm{w}_\bullet)})}\Big\downarrow & , \quad p_{\mathrm{Conf}}^* & \Big\downarrow \\ \underset{\{1,\cdots,N\}}{\mathrm{Conf}}\,(\mathbb{C}) \times B\mathbb{C}^\times & {\scriptstyle\{1,\cdots,N\}} & B\mathbb{C}^\times \end{array} \right)\right) \tag{58}$$

Next we turn to the task of encoding the construction on the right in a form codeable into homotopically typed programming languages such as Agda: This is achieved in §6 below (Thm. 6.8). In preparation, we now first discuss homotopy type theory and its encoding of Gauss-Manin connections in generality (§5).
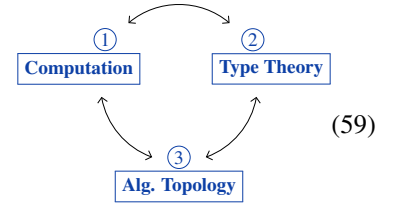
5 ...via dependent homotopy type theory

Here we recast the topological constructions of §4 into a form codeable into the programming language of *homotopy type theory* (HoTT, Lit. 2.27), which we motivate and survey in §5.1: In §5.2, we translate the previous Thm. 4.13 to a construction of data transport (Lit. 2.30) through dependent type families whose "semantics" is given by the monodromy of Gauss-Manin connections (Def. 5.16). Specializing to the situation of Ex. 4.14 provides – further below in §6 – the promised type-theoretic encoding of monodromy of Knizhnik-Zamolodchikov connections and hence of anyon braid quantum gates (Thm. 6.8).

5.1 From data types to homotopy types

We give an informal but detailed exposition of the general principles of programming languages with homotopy types to naturally motivate — from just the idea of declaring for strictly all data a corresponding *type specification* — the homotopy theoretic language structures which in §6 serve to naturally specify anyon braid quantum gates.

While there is by now a fair supply of literature on the subject of this *Homotopy Type Theory* (HoTT, Lit. 2.27), we feel that an exposition along the following lines has been missing; in any case it serves to coherently record and reference the *Rosetta stone*-like dictionary (the homotopical "computational trilogy", see pointers in [SS22-TQC, p. 4], Lit. 2.26) which translates our algebro-topological theorems in §4 to the programming language constructions in §5.2 and §6.

$$
\begin{array}{ccc}
& \text{\textcircled{1}} \longrightarrow \text{\textcircled{2}} & \\
\boxed{\text{Computation}} & & \boxed{\text{Type Theory}} \\
& \text{\textcircled{3}} & \\
& \boxed{\text{Alg. Topology}} &
\end{array}
\qquad (59)
$$

It is worth emphasizing that the type theoretic picture presented here is not only useful for formal verification or in programming. Throughout mathematics, it is crucial to keep in mind what type of mathematical objects we are working with so that we don't end up with gibberish — for example, multiplying two vectors in a general vector space and expecting a vector back. When attended to carefully, this obvious idea of making sure we know what we are talking about gives rise to a foundational system which we will now outline.

The idea of typed programming languages... It is a time-honored idea that every piece of data handled by a programming language be assigned a *type* which specifies its intended nature (e.g. [ML82][Th91][St93][Lu94][Gu95][Co11][Ha16], see also p. 53 below).

For example, declaring a numerical datum d to be of the type of natural numbers — to be denoted "$d : \mathbb{N}$" (111) — instead of, say, the type $\mathbb{Z}$ of integers or the type $\mathbb{Q}$ of rational or the type $\mathbb{R}$ of real or the type $\mathbb{C}$ of complex numbers (all discussed in §6), conveys information about how subsequent parts of the program may or may not operate with this datum. For instance, if a program computes a number which can be certified to be of type $\mathbb{N}$, as opposed to the less constrained type $\mathbb{Z}$, this guarantees the non-negativity of that number, which may be important for the soundness of subsequent computations.

With this understood, the **purpose of typed programming languages is**:

To declare *programs f* which, *given data d of input type D*, are guaranteed to *produce data $f(d)$ of output type A*.

Following tradition in formal logic, such algorithms ("assertions" [RW1910, p. xviii], "proofs" [Ch40, §5], "validations" [Ko61, p. 223]) are denoted [Ho95, §2.1.2][10] by the symbol "$\vdash$" [Fr1879, §2] for *judgements*, cf. [ML84, p. 2][ML96, §1]:

Programming language (syntax)	**Mathematical denotation** (semantics)	
Given... ...and moreover... ...construct		(60)
$\gamma : \Gamma,$ $\quad$ $d : D$ $\quad \vdash \quad$ $f_\gamma(d) : A$ *any data context* $\quad$ *data d of type D* $\quad$ *data of type A*	$\Gamma \times D \xdashrightarrow{\;f\;} A$ space of input data $\quad$ function / map $\quad$ space of output data	

...and their mathematical meaning. As shown on the right, making the data typing explicit allows one to recognize evident "mathematical meaning" or "denotational semantics" of programs (a seminal idea due to [Sc70][ScSt71], for exposition see [SK95, §9]): A program f that (is guaranteed to eventually halt and then) outputs data of type A when run on input of type D is evidently a *function* or *map* from the "space" of all data of type D to that of all data of type A. This tautologous-sounding statement evolves into a remarkably powerful relation between programming languages and algebraic topology as we now progress through the full logical consequences of the principle of typed programming.

---

[10] As is usual, in (60) we are including denotation of a generic context of "constants" c of any type Γ – on which the execution of f may well depend, but which one may not want to regard as external parameters but as implicit arguments passed to the program (e.g. "flags"). As also usual, often we will notationally suppress this generic background context.

The idea of the type of types. Taking the data-typing principle seriously, one realizes that types are clearly a kind of data themselves — and hence ought to be assigned a type: We write "Type" for the ("large") data type of all types — often called the *type universe* and denoted "$\mathcal{U}$" or similar [ML75, §1.10][ML84, pp. 47][Ho95, §2.3.5][Pa98].

The semantics of type universes is given by *Grothendieck universes* in set theory (e.g. [Schu72, §3.2], references in [Kr07, §6.4.4]), or rather – via the univalence axiom (105) – their generalization to *object classifiers* "Obj" [Lu09, §6.1.6], see below around (106).

Type theory	Homotopy theory
$\Gamma \ \vdash \ D : \text{Type}$	$\Gamma \xrightarrow{\ D\ } \text{Obj}$

(61)

The idea of type formation. Given such types, there are fairly self-evident rules (nicely motivated in [ML84][ML96], following "constructive logic" (92)) for forming/constructing further types out of them. For example, given a pair of types X_1 and X_2, one will consider the *product type* $X_1 \times X_2$ of *pairs* of data (d_1, d_2) with $d_1 : X_1$, $d_2 : X_2$. As the notation suggests, the denotational semantics of the syntactic rules for product types are the characteristic properties of *product spaces*:

(62)

Here and henceforth, we use the following traditional notation for *syntactic rules* of the programming language:

- A horizontal line denotes a *natural deduction* rule ([Ge34], see [Ge69]) for passing from the *judgement* (60) above to that below the line. By such rules, valid typed programs are incrementally formed (e.g. [DA18]).
- The symbol "$\equiv$" relates terms which are regarded as *syntactically equal* ("*definitional equality*", e.g. [Ch07, §10.1][Th91, §5.2.1][UFP13, p. 19]), to be distinguished from "*identification*" (71) and "*propositional equality*" (97).

35

The idea of dependently typed programming languages. In general, not only the data itself but also its type may *depend* on the given data context (see [Ho97][Ch13, §1.2.2]). For example, depending on a natural number $d : \mathbb{N}$ previously computed, a (quantum state-)vector ψ_d might be specifically declared to be of the type $\mathbb{C}^d$ of elements of the d-dimensional complex Hilbert space. Here these finite-dimensional Hilbert spaces jointly form a *dependent type*, namely depending on data of the type of natural numbers (this being their dimension).

Trivial as this may superficially seem at this syntactic level (it is not, though), a programming language that computes such ψ_d, while adhering to the principle of its dependent typing, has a conceptually interesting semantics, namely the function (60) that it computes is now a *section of a fibration*:

Programming language (syntax)	**Mathematical denotation** (semantics)
Given... ...construct $\Gamma,\ d : \mathbb{N} \quad \vdash \quad \psi_d : \mathbb{C}^d$ *anything* *a nat. number* *a vector* *in d-dim space*	$\mathscr{V} = \Gamma \times \coprod_{d \in \mathbb{N}} \mathbb{C}^d$ $\psi_{(-)}$ **section** **fibration** $\Gamma \times \mathbb{N} = \Gamma \times \mathbb{N}$ **base**

Therefore the denotational semantics of dependent types is that of fibrations ("bundles", often called "display maps" in this context, see e.g. [Ja98, §10][Jo17]), such as known from homotopy theory (e.g. [Sh15, §2, §3]).

Dependent types and dependent data	**Iterated fibrations and their relative sections**
context type **dependent type** $\Gamma,\ x_c : X_c \quad \vdash \quad E_c(x_c) : \mathrm{Type}$ $\Gamma,\ x_c : X_c \quad \vdash \quad \sigma_c(x_c) : E_c(x_c)$ **dependent term**	**fibration** E σ **section** $X = X$ Γ
$\Gamma,\ x : X \quad \vdash \quad \sigma_c(x_c) : E_c\big(f_c(x_c)\big)$ *substitution* **variable**	**pullback fibration** E $f^*E \longrightarrow E$ σ **relative section** σ **section** (pb) $X \xrightarrow{\ f\ } Y \quad \Leftrightarrow \quad X = X \xrightarrow{\ f\ } Y$ Γ Γ

(63)

The idea of aggregating dependent data. Given such a dependent type $d : D \vdash C_d : \mathrm{Type}$ (63), there are two natural ways to "aggregate" all the types C_d into a single independent type:

(68) by forming the *dependent product* $\prod_{d:D} C_d$ of all C_d — such that to give data of this aggregated type is to give data of type C_d *for each* $d : D$, hence to give a *dependent function* of the form $(d : D) \to C_d$ (a section of the fibration);

(65) by forming the *dependent co-product* $\coprod_{d:D} C_d$ of all C_d — such that to give data of this aggregated type is to give data of type C_d *for one* $d : D$, hence to give a *pair* of the form $(d : D) \times C_d$ (a point in the fibration's total space):

Given types:	the type former:		gives type of:	which:
$\vdash D : \mathrm{Type}$ $d : D \vdash C_d : \mathrm{Type}$	**dependent product**	$\prod_{d:D} C_d \ = \ (d : D) \to C_d$	**dependent functions**	*map* data d of type D *to* data of type C_d
	dependent co-product[11]	$\coprod_{d:D} C_d \ = \ (d : D) \times C_d$	**dependent pairs**	*pair* data d of type D *with* data of type C_d

(64)

Concretely, the inference rules for dependent pair types are as follows, in evident dependent generalization of (62):

Given one type...   **indexing another one...**

$$\frac{\Gamma \;\vdash\; I : \text{Type} \qquad \Gamma,\, i : I \;\vdash\; X_i : \text{Type}}{\Gamma \;\vdash\; (i : I) \times X_i \;:\; \text{Type}}$$

Dependent pair formation rule

...we infer the type of X_i data paired with its index $i : I$.

Given a program which computes I-data...   **... and a program which computes X_i-data...**

$$\frac{\Gamma \;\vdash\; i : I \qquad \Gamma \;\vdash\; x_i : X_i}{\Gamma \;\vdash\; (i,x) : (i : I) \times X_i}$$

Dependent pair introduction rule

... we infer a program which computes the pairs (i, x_i).

Given a program which computes pairs...

$$\frac{\Gamma \;\vdash\; p : (i : I) \times X_i}{\Gamma \;\vdash\; \text{pr}_I(p) : I}$$

$$\Gamma \;\vdash\; \text{pr}_X(p) : X_{\text{pr}_I(p)}$$

Dependent pair elimination rule

... we infer programs which compute the components...

...such that these are indeed the components of the pair...

$$\frac{\Gamma \;\vdash\; i : I \qquad \Gamma \;\vdash\; x : X_i}{\Gamma \;\vdash\; \text{pr}_I(i,x) \equiv i \;:\; I}$$

$$\Gamma \;\vdash\; \text{pr}_X(i,x) \equiv x \;:\; X_i$$

Dependent pair computation rule

$$\frac{\Gamma \;\vdash\; p : (i : I) \times X_i}{\Gamma \;\vdash\; p \equiv \big(\text{pr}_I(p),\, \text{pr}_X(p)\big) \;:\; (i : I) \times X_i}$$

Dependent pair uniqueness rule

...and any pair is the pair of its components.

(65)

The adjoint inference rules for dependent function types are shown in (68). Notice (66) how the term introduction rule (68), when specialized to the case that the various dependencies are trivial, expresses but the subtle yet crucial distinction between a function as such (60) and that function regarded as data *of function type* — the denotational semantics of which is given by *mapping space adjunctions* (24).

Term introduction of function type	**Internal hom adjunction**
function	**map**
$\dfrac{\Gamma,\, d : D \;\vdash\; c_d : C}{\Gamma \;\vdash\; (d \mapsto c_d) : D \to C}$	$\dfrac{\Gamma \times D \xrightarrow{\;f\;} C}{\Gamma \xrightarrow{\;\tilde{f}\;} \text{Map}(D, C)}$
data of function type	**element of mapping space**

(66)

The semantics of *general* dependent function types $(\delta : \Delta_\gamma) \to (-)$ is that of forming spaces of relative sections via the *slice mapping space* adjunction (31), hence is (e.g. [Sh15, Ex. 2.9]) the right base change operation $(p_D)_*$ from base Δ to base Γ (33).

$$\text{Type}_\Delta \xleftarrow{\quad \big((\delta{:}\Delta) \vdash (-)_{p(\delta)}\big) \quad} \text{Type}_\Gamma$$
with $\overline{\quad \big((\gamma{:}\Gamma) \vdash (\delta_\gamma{:}\Delta_\gamma) \times (-)\big) \quad}$ above and $\underline{\quad \big((\gamma{:}\Gamma) \vdash (\delta_\gamma{:}\Delta_\gamma) \to (-)\big) \quad}$ below

Aggregation of dependent types	**Base change via Local Cartesian Closure**
$\dfrac{(\gamma : \Gamma) \times (\delta : \Delta_\gamma),\, (d_\gamma : D_\gamma) \;\vdash\; c_{(\delta, d_\gamma)} : C_\delta}{\begin{array}{l}(\gamma : \Gamma),\, (d_\gamma : D_\gamma) \quad\;\; \vdash\; (\delta : \Delta_\gamma) \to\; C_\delta \\ \qquad\qquad\qquad\quad \delta \quad\;\; \mapsto c_{(\Delta_\gamma, d_\gamma)}\end{array}}$	
$\dfrac{(\gamma : \Gamma) \times (\delta : \Delta_\gamma),\, (d_\gamma : D_\gamma) \;\vdash\; (c_\gamma : C_\Gamma)}{(\gamma : \Gamma),\, (\delta : \Delta_\gamma) \times (d_\gamma : D_\gamma) \;\vdash\; (c_\gamma : C_\Gamma)}$	

(67)

---

[11] The traditional notation for the dependent pair type is some variant of "$\sum_{d : D} C_d$", pronounced the "dependent sum". While widely used, this is clearly a misnomer — and it becomes a fatal misnomer once we generalize dependent type theory to dependent linear type theory [SS23-QM][SS23-EoS], where an actual dependent *sum* (in the sense of linear algebra) does appear beside the dependent co-product. Luckily, the alternative pair-type notation $(d : D) \times C_d$ (cf. [Wä, p. 1]) not only circumvents this clash but is arguably also more convenient and more suggestive in general — as seen for instance in (67) and on p. 53 below).

This means (as originally understood by [See84], see also [CGH14] and [Sh15, p. 11]) that dependent type systems with dependent function type formation (68) have denotational semantics in categories of spaces such as kHausSpc (27) where each slice category has a mapping space adjunction, called *locally cartesian closed categories* (LCCC), cf. (107) below.

$$(68)$$

The type of dependent types. For example, the application of the function data introduction rule (66) to the type universe (61) shows that D-dependent types (63) are the same as functions taking D-data to types (61):

$$
\frac{d : D \;\vdash\; C_d : \mathrm{Type}}{\vdash\; (d \mapsto C_d) : D \to \mathrm{Type}}
$$
D-dependent type

function data into type universe

$$(69)$$

Therefore it makes sense to define the (large) *type of D-dependent types* (also known as *type families over D*) as

$$
\mathrm{Type}_D \;:\equiv\; \big(D \to \mathrm{Type}\big).
$$

$$(70)$$

The idea of homotopy-typed programming languages. Taking seriously the idea that all data should be (dependently) typed suggests that the same should be true for *data identifications*: A certificate p which identifies two data structures $d_1, d_2 : D$ – witnessing that they are computationally equivalent given the defining nature of their type D – should itself be data of the *type of identifications* of D-data. For simple types – such numbers – we may identify two elements just when they are equal; in this way, identification of general data is a extension of the notion of the notion of equality. Every datum should be trivially

identified with itself. This acts as reflexivity of equality:

Type of identifications		Path space fibration
Given... ...**we have**		**path space** (mapping space out of interval) **interval**
$\Gamma,\quad X:\text{Type},\quad (x_1,x_2):X\times X\ \vdash\ \text{Id}_X(x_1,x_2):\text{Type}$		$X^I:\equiv\text{Map}(I,X)\qquad I$ $(\text{ev}_0,\text{ev}_1)\downarrow \qquad\qquad \uparrow$ $X\times X\qquad\qquad \{0,1\}$

(71)

any data   *in this context*   *a type*   *a pair of data of this type*   *the type of identifications of these terms*

(I) *All data is reflexively identified with itself.*

Reflexivity certificates	Constant paths sectioning the path fibration
Given... ...**we obtain** $\Gamma,\ x:X\ \vdash\ \text{id}_X(x):\text{Id}_X(x,x)$ *some data* *a certificate of its trivial self-identification.*	$\text{diag}^*X^I \longrightarrow X^I$ (**path fibration**) $\Leftrightarrow$ (**path fibration**) X^I with **diagonal map** $\text{diag}:x\mapsto(x,x)$, **assign constant paths** $x\mapsto\text{const}_x$

(72)

As indicated on the right, the foundational insight of *homotopy type theory* is that the semantics of such Id-types is that of *path space fibrations* in algebraic topology ([AW09]). One may understand this in two equivalent ways:

Inductive notion of identification. Naively, reflexive self-identifications (72) should freely generate the identification types (71). Expressed operationally this should mean that a (dependent) function out of an identification type is given as soon as it is specified on these reflexivity certificates. This *inductive* definition (cf. pp. 47 below) of identifications is the profound insight of [ML75, §1.7 and p. 94], who labeled this inference rule "J" under which name it became widely known:

(J) $\Leftrightarrow^{12}$ (IIa) and (IIb)   (Martin-Löf, Coquand)

Induction principle for identification certificates	Constant path map lifts against fibrations
Given... ...**an identification-dependent...** ...**type...** $x_1:X,x_2:X,\quad p_{12}:\text{Id}_X(x_1,x_2)\ \vdash\ E(x_1,x_2,p_{12}):\text{Type}$ $x:X\qquad\qquad\vdash\quad \sigma:E\big(x,x,\text{id}_X(x)\big)$ ...**and...** ...**data for all reflexive identifications...** $x_1:X,x_2:X,\quad p_{12}:\text{Id}_X(x_1,x_2)\ \vdash\ \widehat{\sigma}:E(x_1,x_2,p_{12})$ ...**obtain...** ...**for any identification...** ...**compatible data.** such that $\widehat{\sigma}\big(\text{id}_X(x)\big)\equiv\sigma\big(\text{id}_X(x)\big)$	$X\xrightarrow{\ \sigma\ } E=f^*E'\longrightarrow E'$ $\text{id}_X\downarrow\quad \widehat{\sigma}\nearrow\ \textbf{lift}\qquad\downarrow(\text{pb})\qquad\downarrow$ $X^I ===== X^I\xrightarrow{\ f\ } B$

(73)
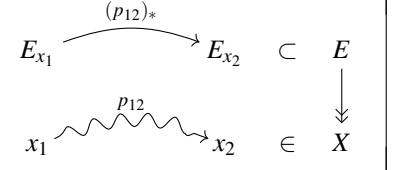
Remarkably, as indicated on the right of (73), the semantics of the Id-induction rule is ([AW09, §3.4], review in [Sh12, 3, pp. 26]) the *lifting property* that characterizes 'very good path space objects" in abstract homotopy theory ([DS95, §4.12]):

Notice that naive interpretation of the reflexive self-identifications in $\text{Id}_X(x_1,x_2)$ would be the diagonal inclusion diag : $X\to X\times X$ of the pairs of equal elements (x,x); but this map is not in general a fibration; and in homotopy theory demands that it be *resolved* (up to weak equivalence) by a path space fibration $X^I\twoheadrightarrow X$, which also makes the resulting inclusion of reflexive self-identifications $\text{id}_X:X\to X^I$ an "acyclic cofibration", thus implying the lifting property on the right above.

Identification as transport. Much older than this inductive understanding of identification is the characterization of identifications as those processes which *preserve all properties* ("salva veritate", Leibniz $\sim$ 1700, cf. [Le18, p. 373]). Understood as: *preserve all dependent data* this is the following type-theoretic rule – which is in fact implied (cf. [UFP13, §2.3]) by Id-induction (73) and which has the striking semantic interpretation of *path lifting* and *fiber transport* (Lit. 2.30):

---
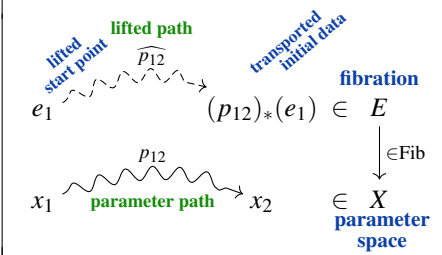
[12]In type theory literature, the Id-induction J-rule (73) is traditionally postulated directly (going back to [ML75, §1.7 and p. 94][NPS90, §8.1] in the general context of inductive types, cf. pp. 47). Its equivalence to the combination of "transport" (74) with "reversal" (75) was expressed in [Co11] and further amplified in [LP15]; a detailed proof is spelled out in [Gö18, §4]. The implication (J) $\Rightarrow$ (II) has an evident meaning: The J-rule is the application of the transport rule (IIa) to just those identifications of identifications given by the uniqueness rule (IIb)

(IIa) *Substitution of identifications preserves computations.*

Transport of data along identifications of variables	Fiber transport in fibrations
Given an X-dependent type... $$\frac{\Gamma \;\vdash\; X : \mathrm{Type} \qquad \Gamma,\; x : X \;\vdash\; E_x : \mathrm{Type}}{\Gamma,\; x_1, x_2 : X,\; p_{12} : \mathrm{Id}_X(x_1, x_2) \;\vdash\; (p_{12})_* : E_{x_1} \to E_{x_2}}$$ **...and an identification of X-data...** $\qquad$ **...obtain transformation of all dependent data.** such that $\mathrm{id}_X(x)_* : e \mapsto e$	(74)
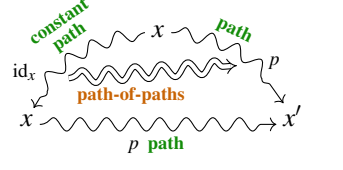
We furthermore require that $\mathrm{id}_X(x)_* \equiv \mathrm{id}_{E_x}$; transporting along the reflexive self-identification gives the identity function.

(IIa') *Substitution of identifications preserves identifications.*

Given... a parameter type ... and a type depending on it, ... $$\frac{\Gamma \;\vdash\; X : \mathrm{Type} \qquad \Gamma,\; x : X \;\vdash\; E_x : \mathrm{Type}}{\Gamma,\; p_{12} : \mathrm{Id}_X(x_1, x_2),\; e_1 : E_{x_1} \;\vdash\; \widehat{p_{12}} : \mathrm{Id}_{(x:X)\times E_x}\big((x_1, e_1),\, (x_2, (p_{12})_*(e_1))\big)}$$ **for any parameter-identification...** **and data at the initial parameter value, obtain** **an identification with the transported data.**	

When combined with the following rule (75), the transport/lifting rule (74) is *equivalent* to Id-induction (cf. prev. footnote):
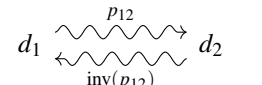
(IIb) *Identifications are preserved by composition with identities.*

Essential uniqueness of identification certificates	Contraction of based path space
Given... $$\Gamma,\; x, x' : X,\; p : \mathrm{Id}_X(x, x') \;\vdash\; p_* : \mathrm{Id}_{(x':X)\times \mathrm{Id}_X(x,x')}\big((x, \mathrm{id}_x),\, (x', p)\big)$$ **an identification** **...we obtain** **a certificate** **of identification-of-identifications** **with the reflexive identif.**	(75)
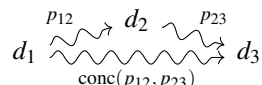
We furthermore require that $\widehat{\mathrm{id}_X x_1} \equiv \mathrm{id}_{(x_1, e_1)}$. It follows that **types** of a dependently typed language implementing these inference rules for identification types **are** (fibrations of) **homotopy types** [Aw12], in that they behave like topological spaces up to (weak) homotopy equivalences:

The idea of homotopy type structure. Informally, path induction (73) says that to define data $\hat{\sigma}$ of type $E(p_{12})$ dependent on an identification $p_{12} : \mathrm{Id}_D(d_1, d_2)$ (for *free variables* d_1 and d_2), it suffices to assume that d_2 is d_1 and p_{12} is $\mathrm{id}_D(d_1)$ and to define (just) the corresponding data $\sigma : E\big(\mathrm{id}_X(x_1)\big)$ in this special case. Using this one finds, for instance:

- *Inversion of identifications* by declaring that that reflexive self-identification is its own inverse (cf. [UFP13, Lem. 2.1.1]). This acts as symmetry of equality:

$$\mathrm{inv}_D \;:\; (d_1, d_2 : D) \to \begin{pmatrix} \mathrm{Id}_D(d_1, d_2) & \to & \mathrm{Id}_D(d_2, d_1) \\ \mathrm{id}_{d_1} & \mapsto & \mathrm{id}_{d_1} \end{pmatrix}$$	(76)

- *Concatenation of identifications* by declaring them to be trivial for reflexive identificiations (cf. [UFP13, Lem. 2.1.2]). This acts as transitivity of equality:

$$\mathrm{conc}_D \;:\; (d_1, d_2 : D) \to \begin{pmatrix} \mathrm{Id}_D(d_1, d_2) & \to & \big(\mathrm{Id}_D(d_2, d_3) \to \mathrm{Id}_D(d_1, d_3)\big) \\ \mathrm{id}_{d_1} & \mapsto & (p \mapsto p) \end{pmatrix}$$	(77)

It is **the magic of homotopy type theory** that tautologous-seeming constructions like (76) and (77) provide — all through the induction principle (73) — just the right information to know everything there is to know about homotopy types — a subject that is far from trivial (Lit. 2.12). Notably, it follows that:

- *Reverse identifications* (76) *are inverses* under concatenation (77) (up to higher identification) simply by checking that this holds trivially for reflexive self-identifications (cf. [UFP13, Lem. 2.1.4 (ii)]):

$$
\begin{aligned}
\mathrm{ivt}_D \,:\, (d_1, d_2 : D) \to \Big((p : \mathrm{Id}_D(d_1, d_1)) &\;\to\; \mathrm{Id}_{(\mathrm{Id}_D(d_1,d_1))}\big(\mathrm{conc}(p, \mathrm{inv}(p)), \mathrm{id}_{d_1}\big) \Big) \\
\mathrm{id}_{d_1} &\;\mapsto\; \mathrm{id}_{(\mathrm{id}_{d_1})}
\end{aligned}
$$

(78)

- *Concatenation of identifications is associative* (up to higher identification), simply by checking that this holds trivially for reflexive self-identifications:

$$
\begin{aligned}
\mathrm{ast}_D \,:\, (d_1, d_2, d_3, d_4 : D) \to \left(\begin{pmatrix} p_{12} : \mathrm{Id}_D(d_1, d_2) \\ p_{23} : \mathrm{Id}_D(d_2, d_3) \\ p_{34} : \mathrm{Id}_D(d_3, d_4) \end{pmatrix} \to \mathrm{Id}_{(\mathrm{Id}_D)}\Big(\mathrm{conc}\big(p_{12}, \mathrm{conc}(p_{23}, p_{34})\big), \mathrm{conc}\big(\mathrm{conc}(p_{12}, p_{23}), p_{34}\big) \Big) \right) \\
\big(\mathrm{id}_{d_1}, \mathrm{id}_{d_1}, \mathrm{id}_{d_1}\big) \;\mapsto\; \mathrm{id}_{(\mathrm{id}_{d_1})}
\end{aligned}
$$
(79)

As first observed in [HS98], translating to algebraic topology this kind of structure (namely "arrows" $d_1 \rightsquigarrow d_2$ between "objects" d_1, d_2, equipped with invertible and associative composition) is known as constituting a *groupoid* (jargon for: "akin to a group but possibly with several objects", see e.g. [Wei96]), specifically the *fundamental path groupoid* (e.g. [San11]) of the corresponding topological space D.

Better yet, since invertibility (78) and associativity (77) appear themselves as identifications-of-identifications (2-arrows), which by repeated use of the Id-induction rule (73) are found to come with appropriate higher-dimensional analogs of composition, inverses and associativity, the type theory reflects "higher dimensional" groupoid structure [vdBG11], as known from the "fundamental 2-groupoids" of topological spaces [HKK02] and more generally from their "fundamental ∞-groupoids" typically modeled (e.g. [Lu09, §1.1.2]) by Kan fibrant singular simplicial sets [KLV12] or more general fibrant models [AW09].

(80)

With types thus revealed as secretly *being* ∞-groupoids, one discovers similarly that functions (programs!) from one type to another secretly *are* ∞-*functors*, in that they respect this ∞-groupoid structure, again all by Id-induction (73) (cf. [UFP13, Lem. 2.2.1]):

$$
\begin{aligned}
f : D \to D' \;\vdash\; \mathrm{ap}_f \,:\, (d_1, d_2 : D) \to \;\Big(\mathrm{Id}_D\big(d_1, d_2\big) &\;\to\; \mathrm{Id}_{D'}\big(f(d_1), f(d_2)\big) \Big) \\
\mathrm{id}_{d_1} &\;\mapsto\; \mathrm{id}_{f(d_1)}
\end{aligned}
$$

(81)

$$
\begin{aligned}
\mathrm{cmpstr}_f \,:\, \begin{pmatrix} d_1, d_2, \\ d_3 : D \end{pmatrix} \to \left(\begin{pmatrix} p_{12} : \mathrm{Id}_D(d_1, d_2), \\ p_{23} : \mathrm{Id}_D(d_2, d_3) \end{pmatrix} \to \mathrm{Id}_{\mathrm{Id}_D} \begin{pmatrix} \mathrm{ap}_f\big(\mathrm{conc}(p_{12}, p_{23})\big), \\ \mathrm{conc}\big(\mathrm{ap}_f(p_{12}), \mathrm{ap}_f(p_{23})\big) \end{pmatrix} \right) \\
\big(\mathrm{id}_{d_1}, \mathrm{id}_{d_1}\big) \;\mapsto\; \mathrm{id}_{\mathrm{id}_{f(d_1)}}
\end{aligned}
$$

(82)

41

Transport (74) is also functorial in identifications, which is provable just as easily by appealing to path induction (73).

$$
\begin{array}{l}
p_{12} : \mathrm{Id}_X(x_1,x_2),\, p_{23} : \mathrm{Id}_X(x_2,x_3) \quad \vdash \quad \mathrm{func} : \mathrm{Id}\big((p_{23})_* \circ (p_{12})_*, \mathrm{conc}(p_{12},p_{23})_*\big) \\[1mm]
\qquad\qquad \mathrm{id}_{x_1},\quad p_{23} \qquad\qquad \longmapsto \qquad\qquad \mathrm{id}_{(p_{23})_*}
\end{array}
\qquad (83)
$$

Basic notions of homotopy theory among types. In such a homotopically typed programming language, basic constructions on types correspond to fundamental construction in homotopy theory (Lit. 2.12). For instance:

Colloquial	Homotopy type theory	Homotopy theory	
Cospan	$\vdash f : Y' \to Y$ $\vdash p_X : X \to Y$		(84)
Fibrant resolution	$y : Y,\, x : X \;\vdash\; \mathrm{Id}_Y\big(y, p_X(x)\big)$	Brown's factorization lemma [Br73, p. 421]	(85)
Homotopy pullback	$\vdash \big((y', x) : Y' \times X\big)$ $\times\, \mathrm{Id}_Y\big(f(y'), p_X(x)\big)$		(86)
Homotopy fiber	$\vdash (x : X) \times \mathrm{Id}_Y\big(y, p_X(x)\big)$		(87)
Loop space	$\vdash \mathrm{Id}_Y(y, y)$		(88)
Contraction	$\vdash \; \mathrm{contr}_{x_0} \; : \; (x \; : \; X) \;\to\; \mathrm{Id}_X(x_0, x)$		(89)

42

The idea of propositions as types. *Unique existence* is a crucial logical notion — many theorems in mathematics can be expressed as the statement that a concept uniquely defines a given object. In the homotopical interpretation of type theory, the statement that there is a unique element of a type X becomes the statement that X is *contractible*. To say that a type X has a unique element, we must give an element of it and a way to identify every other element with that element. Such certificates are evidently of this type:

$$\underset{\substack{\text{Certificates that there exists}\\\text{unique } X\text{-data}}}{\exists!\, X} \;:\equiv\; \text{isContractible}(X) \;:\equiv\; (x_0 : X) \times \Big(\underset{\substack{\text{There is}\\\text{a datum}}}{(x : X)} \to \underset{\substack{\text{onto which all}\\X\text{-data contracts.}}}{\text{Id}_X(x_0, x)} \Big). \tag{90}$$

In particular, if all the *identification types* $\text{Id}_P(p, p')$ (71) of a given type P are certified to be contractible (90), this means that all pairs of data $p, p' : P$ are identified – *if* there are any. There is at most one element of such types P, which is then canonically understood to be a *certificate of the truth* – hence a *proof* – of the *proposition* that "P-data exists" at all:

$$\text{isProposition}(P) \;:\equiv\; (p, p' : P) \to \exists!\, \text{Id}_P(p, p') . \tag{91}$$

As as example, for any type X the type $\exists! X$ (90) which asserts that X has a unique element is a proposition; there is at most one way to give a unique element of a type. To give an element of $\exists! X$ is simply to prove that X has a unique element.

This way, propositions and their (first order) *propositional logic emerge* inside (dependent) type theory – a profound statement famous as the "Curry-Howard isomorphism" (e.g. [SU06]) or as the slogan "propositions as types" (e.g. [AB04][Wa15], for review in our homotopical context see [UFP13, §1.11]). Moreover, under this identification, the logical connectives on propositions are nothing but the various type formation rules, an observation known as the *BHK correspondence* [13] [Tr77, §2][TvD88, §3.1][Br99, p. 96]:

- For $P, P' :$ Prop, the product type $P \times P'$ (62) reflects the proposition that P *and* P' hold.
- For $P, P' :$ Prop, the coproduct type $P \sqcup P'$ (115) reflects the proposition that P *or* P' hold.
- For $P, P' :$ Prop, the function type $P \to P'$ reflects the proposition that P *implies* P'.
- For $D :$ Set, $d : D \vdash P(d) :$ Prop
  - the dependent function type $(d : D) \to P(d)$ (68) reflects the proposition $\forall_{d:D} P(d)$ that $P(d)$ holds *for all d*.
  - the (-1)-truncated (128) dependent pair type $\exists\big((d : D) \times P(d)\big)$ (65) reflects the proposition $\exists_{d:D} P(d)$ that *there exists d* for which $P(d)$ holds.

Type theory		Logic
(-1)-type	$d : D \vdash P(d) :$ Prop	Proposition
product type	$P \times P'$	P and P'
Coproduct type	$P \sqcup P'$	P or P'
Function type	$P \to P'$	P implies P'
Dependent function type	$(d : D) \to P(d)$	$\forall_{d:D} P(d)$
(-1)-truncated dependent pair type	$\exists\big((d : D) \times P(d)\big)$	$\exists_{d:D} P(d)$

(92)

It is this intimate relation between proofs in logic on the one hand and data of types on the other which makes programs written in typed languages automatically come with proofs of their correctness (Lit. 2.26)

Notice that this correspondence is contentful (only) because we consider *dependent* types (63), so that these propositional types $d : D \vdash P(d) :$ Prop are propositions *about* data of the type D that they depend on. For instance, the traditional way to build sets of data satisfying a given proposition is given by the (untruncated) dependent pair type (65) of such dependent propositions:

$$\frac{d \,:\, D \;\vdash\; \overset{\text{proposition about } D\text{-data}}{P(d) \,:\, \text{Prop}}}{\underset{D\text{-data verifying this proposition}}{\{d : D \mid P(d)\}} \;:\equiv\; (d : D) \times P(d) \,:\, \text{Type}}. \tag{93}$$

Here to give an element of $\{d : D \mid P(d)\}$ is precisely to give data d equipped with a certificate that $P(d)$ holds. For instance, we may define the proposition that a number $n : \mathbb{N}$ is even as the type of numbers which divide it evenly in half:

$$\text{isEven}(n) \;:\equiv\; (k : \mathbb{N}) \times \text{Id}_{\mathbb{N}}(2k, n) . \tag{94}$$

We can then define the type of even numbers as $\{n : \mathbb{N} \mid \text{isEven}(n)\}$. While this type technically has as elements triples (n, k, p) where $p : \text{Id}_{\mathbb{N}}(2 \cdot k, n)$ is an identification of $2k$ with n, we note that k is uniquely determined by n and the property that $2k$ equals n, so it is harmless to identify the elements of this type with the even natural numbers.

---

[13]The BHK correspondence (92) originates in the school of mathematical *intuitionism* [Ko32, p. 59][Tr69] and eventually lead to the formulation of dependent type theory in [ML75] (exposition in [ML96, Lec. 3]), which is the historical reason that one also speaks of "intuitionistic type theory".

As another example, since being a proposition is itself a proposition, we may form the type of all propositions among all types (61) is:

<div align="center">Type of all propositional types</div>

$$\mathrm{Prop} \;:\equiv\; (P : \mathrm{Type}) \times \mathrm{isProposition}(P) \tag{95}$$

Examples of propositional types in this sense (95) include: $\mathrm{isContractible}(-)$ (90), $\mathrm{isProposition}(-)$ (95), and $\mathrm{isEquivalence}(-)$ (101).

The idea of data sets. How should one define data *sets* in a homotopy typed programming language? Here our usual intuition for equality helps us: The equality of elements in sets should be a *proposition*. Conversely, a data *set* should be a type X whose identification types $\mathrm{Id}_X(x, y)$ (71) are all propositional (91) — namely, $\mathrm{Id}_X(x,y)$ should reflect the proposition: "x equals y":

$$\mathrm{isSet}(X) \;:\equiv\; \mathrm{isType}_{\leq 0}(X) \;:\equiv\; (x, y : X) \to \mathrm{isProposition}\big(\mathrm{Id}_X(x, y)\big),$$
$$\mathrm{Set} \;:\equiv\; (S : \mathrm{Type}) \times \mathrm{isSet}(S). \tag{96}$$

It is the data of these propositional identification types which deserves[14] to be denoted by the traditional notation for equality (whence: "propositional equality"):

<div align="center">Given
a data set with a pair of identifiable elements   obtain
equality as the certificate of their identification.</div>

$$X : \mathrm{Set}, \quad x, y : X, \quad \mathrm{Id}_X(x, y) \quad \vdash \quad x = y \;:\equiv\; x \rightsquigarrow y \;:\; \mathrm{Id}_X(x, y). \tag{97}$$

With concatenation "conc" (77) of identifications denoted just by their juxtaposition, this allows to obtain classical-looking proof certificates of equality in homotopy data sets, such as:

$$X : \mathrm{Set}, \quad x, y, z : X, \quad (x = y) : \mathrm{Id}_X(xy), \quad (y = z) : \mathrm{Id}_X(yz) \quad \vdash \quad x = y = z \;:\; \mathrm{Id}_X(x, z). \tag{98}$$

These homotopy data sets (96) encode inside homotopy-type languages the ordinary kind of data which is available in non-homotopic programming languages (cf. [RS15]), such as the type of bits (110), or of natural numbers (111): $\mathrm{Bit}, \mathbb{N} : \mathrm{Set}$.

The idea of higher homotopy types. Continuing the pattern by which the notion of propositional types (95) leads to that of data sets (96) yields the notion of higher homotopy types:

The next stage in the hierarchy corresponds (as first observed by [HS98]) to what in homotopy theory are called *groupoids* or *homotopy 1-types* (exposition and introduction in [Wei96][San11][Ri20, §1.2]), where for a pair of data of such type there may be not just one (or none), but a whole set of different ways of identifying them:

$$\mathrm{isGroupoid}(X) \;:\equiv\; \mathrm{isType}_{\leq 1}(X) \;:\equiv\; (x, y : X) \to \mathrm{isSet}\big(\mathrm{Id}_X(x, y)\big),$$
$$\mathrm{Grpd} \;:\equiv\; \mathrm{Type}_{\leq 1} \;:\equiv\; (\mathscr{G} : \mathrm{Type}) \times \mathrm{isGroupoid}(\mathscr{G}). \tag{99}$$

Historically, the passage from understanding such homotopy 1-types to understanding their generalization to higher homotopy n-types was a long and convoluted one (cf. [Hi88]); but in homotopy-typed language this is now immediate (the key insight of [Vo10]):

$$\mathrm{isType}_{\leq n+1}(X) \;:\equiv\; (x, y : X) \to \mathrm{isType}_{\leq n}\big(\mathrm{Id}_X(x, y)\big),$$
$$\mathrm{Type}_{\leq n+1} \;:\equiv\; (X : \mathrm{Type}) \times \mathrm{isType}_{\leq n}(X). \tag{100}$$

The mathematical semantics of these homotopical data n-types is by what in homotopy theory has long been known (long before any relation to type theory was even thought of) as *homotopy n-types* and generally as *homotopy types* (e.g. [Sp66, p. 25 & §8][Bau95]).

The basic examples of higher homotopy n-types are the *higher spheres* (119); see also the comments on higher structures on pp. 55. Often one is interested in *truncating* the homotopy level of a type, see p. 5.1 below.

---

[14]It is famously popular in the homotopy type theory literature to use the notation "$x =_X y$" for what we denote $\mathrm{Id}_X(x, y)$, even when X is a higher homotopy type. We humbly suggest that mathematical intuition is served and much debate is avoided by using the equality sign only for the actual notion of (propositional) equality – which is that of identifications of elements of sets.

The idea of equivalence/isomorphy of types. Two types D, C: Type may appear nominally different, but if one can transform data $f : D \to C$ such that this transformation may be *inverted up to identification*, then D-data is *equivalent* ("isomorphic" [DiC95][HS98, §5.4][KL21, Def. 3.1.1]) to C-data: Whatever program operates on D-data may then be transformed into a program operating on C-data, and vice versa (e.g. [BP01][RPYLG21]).

Alternatively, one may ask [Vo10, p. 8, 10] that $f : D \to C$ be a *bijection* (or *weak equivalence*, cf. below (102)), in that for $c : C$ we have a unique (90) inverse image $d : \mathrm{fib}_c(f)$ (87):

type equivalence	**Homotopy equivalence**	
$f : D \to C \quad \vdash$ **a function is** **an** *equivalence* **iff** **we have a reverse function which is left inverse** **or** *isomorphism* $\mathrm{isEquivalence}(f) :\equiv \begin{cases} \left(\overline{f}_l : C \to D\right) \times \underset{(D \to D)}{\mathrm{Id}}\left(\overline{f}_l \circ f, (d \mapsto d)\right) \\ \times \left(\overline{f}_r : C \to D\right) \times \underset{(C \to C)}{\mathrm{Id}}\left(f \circ \overline{f}_r, (c \mapsto c)\right) \end{cases}$ **and a reverse function which is right inverse**		(101)
$f : D \to C \quad \vdash$ **a function is** **a** *bijection* **iff** **its pre-images are ess. unique** $\mathrm{isBijection}(f) \quad :\equiv \quad (c : C) \to \exists! \, \mathrm{fib}_c(f)$	$\underset{c \in C}{``\forall"} \quad \mathrm{fib}_c(f) \simeq *$	
type bijection	**Contractible fibers**	

These two notions happen to coincide [UFP13, §4.4]:

$$\mathrm{isBijection}(f) \xleftrightarrow{\qquad\qquad\qquad\qquad\qquad} \mathrm{isEquivalence}(f)$$

$$\left((c : C) \mapsto \left(\left(d, f(d) \rightsquigarrow c\right) : \mathrm{fib}_c(f)\right) \mapsto \left(d_0(c) \overset{\mathrm{ctr}_d}{\rightsquigarrow} d, \cdots\right)\right) \longmapsto \left(\begin{array}{l} \overline{f}_l : c \mapsto d_0(c), \ d \mapsto \left(d_0(c) \overset{\mathrm{ctr}_d}{\rightsquigarrow} d\right) \\ \overline{f}_r : c \mapsto d_0(c), \ c \mapsto \mathrm{id}_c \end{array}\right). \qquad (102)$$

If one reads the "for all" quantifier on the right of (101) naively as in (92) (even if C is not 0-truncated), then in the classical homotopy theory (107) of topological spaces this state of affairs is the content of the *classical Whitehead theorem* (e.g. [Br93, Cor. 11.14][AGP02, Thm. 6.3.31]) which says that maps between cofibrant spaces (CW-complexes) are homotopy equivalences as soon as they are *weak* homotopy equivalences in that the homotopy groups of all their homotopy fibers vanish. For this reason the type-theoretic "bijections" above were originally called *weak equivalence* in [Vo10].

However, the beauty of simply recasting (92) the naive quantifier "$\forall_{c:C}$" as the dependent function constructor $(c : C) \to (-)$ dramatically increases the generality of the statement: While various versions of the Whitehead theorem actually fail in general model toposes ([Lu09, §6.5][UFP13, §8.8]), the equivalence (102) holds generally.

We denote the type of equivalence like the function type but equipped with a "$\sim$"-symbol:

$$D, C : \mathrm{Type} \qquad \vdash \qquad \left(D \overset{\sim}{\to} C\right) :\equiv (f : D \to C) \times \mathrm{isEquivalence}(f) \qquad (103)$$

The tautological example of an equivalence is of course the identity function $a \mapsto a$, which is canonically its own left and right inverse, as certified by its reflexive identification $\mathrm{id}_{(a \mapsto a)}$:

$$A : \mathrm{Type} \quad \vdash \quad \left((a \mapsto a), \left(\left(\overline{(a \mapsto a)}^l :\equiv (a \mapsto a), \mathrm{id}_{(a \mapsto a)}\right), \left(\overline{(a \mapsto a)}^r :\equiv (a \mapsto a), \mathrm{id}_{(a \mapsto a)}\right)\right)\right) : \mathrm{Equiv}(A, A). \quad (104)$$

The idea of univalent universes of types. However, there is a priori *another* sensible way to understand identification of types. Namely, since we regard data types A, B themselves as being data of type "type", denoted $A, B : \mathrm{Type}$ (61), there is the notion of their identification certificates (71) $c : \mathrm{Id}_{\mathrm{Type}}(A, B)$, just as for data of any other type. Now, the Id-induction principle (73) gives a transformation from such *identifications* of data to operational *equivalences* (103) between data, induced from taking the identity equivalence $a \mapsto a$ (104) to be the operationalization of the reflexive self-identification id_A (72):

$$\mathrm{operationalize} : \prod_{A,B : \mathrm{Type}} \left(\begin{array}{c} \text{identification certificates} \quad \text{operational equivalences} \\ \text{between types} \qquad\qquad \text{between types} \\ \mathrm{Id}_{\mathrm{Type}}(A, B) \longrightarrow \mathrm{Equiv}(A, B) \\ \mathrm{id}_A \longmapsto (a \mapsto a) \end{array} \right), \quad \mathrm{univalence} : \mathrm{isEquivalence}(\mathrm{operationalize}). \quad (105)$$

One says[15] that a homotopically typed language satisfies *type universe extensionality* [HS98, §5.4] or that it has a *univalent type universe* [Vo10, p. 11] if this comparison function is itself an equivalence (101) (cf. [Es19, §3.11][1lab, §Univalence]).

Another incarnation of the Univalence Axiom (105) says [UFP13, Theorem 4.8.3] that the operation of recording homotopy fibers (87) for all base points constitutes an equivalence (101) between functions into a given type X and X-dependent types (70). Semantically this means (originally conjectured by [Aw12], proven in special cases in [KLV12][Sh15] and generally in [Sh19], review in [Ri22]) that the type universes of univalent homotopy-typed languages are "object classifiers" [Lu09, §6.1.6] reflecting the ambient category of types as an "∞-topos" [Si99][TV][Lu09][Re10]:

Univalent type Universe	**Object classifier**
Functions into X $(Y:\mathrm{Type})\times(Y\to X)$ $\xrightarrow[\text{total space}]{\substack{(Y,f)\mapsto(x\mapsto\mathrm{fib}_x(f))\\ \text{fibers}\\ \simeq\\ ((x:X)\times E_x,\mathrm{pr}_X)\ \leftarrow\!\mapsto\ (x\mapsto E_x)}}$ **X-Dependent types** $(X\to\mathrm{Type})$	$Y\longrightarrow\widehat{\mathrm{Obj}}$ $\downarrow\ \ _{(\mathrm{pb})}\ \ \downarrow$ $X\longrightarrow\mathrm{Obj}$	(106)

In more detail, the homotopy type-theoretic syntax directly interprets into 1-categories (cf. Lit. 2.26) and here those understood to "present" these ∞-toposes (this is the key point of [Lu09]), known as *model toposes* [Re10] or more specifically as *type-theoretic model toposes* [Sh19, §1.3] (building on [Sh15], review in [Ri22, §6.1]), in which the object classifier exists as a universal fibration of small objects.

The archetypical example of such model toposes is the "classical model topos" of classical homotopy theory, traditionally known as the *Kan-Quillen model category of simplicial sets*. This is suitably equivalent (namely Quillen equivalent, presenting the same ∞-topos — for review and references see [FSS20-Cha, §A]) to the *Serre-Quillen model category of k-topological spaces* which tacitly underlies the considerations in §4 but which fails to be strictly "type-theoretic" in that it is not *quite* locally Cartesian closed (only over base spaces which are Hausdorff, cf. (27)):

$$\underset{\S5}{\underset{\substack{\text{syntactic category of}\\ \text{homotopy types}}}{\mathrm{HoTTSyntax}}}\xrightarrow{\ \text{semantics}\ }\underset{\substack{\text{type-theoretic}\\ \text{model topos}\\ \text{of simplicial sets}}}{\Delta\mathrm{Set}_{\mathrm{Qu}}}\xrightarrow{\ \text{Quillen equivalence}\ }\underset{\substack{\text{model topos}\\ \text{of topological spaces}\\ \S4}}{\mathrm{kTopSpc}_{\mathrm{Qu}}}\qquad(107)$$

(The categorical semantics of dependent type theory on the left is due to [See84][Ho97][Ja98], the homotopy/model-category theoretic aspect due to [AW09], and finally the construction of the univalent object classifier due [KLV12] and then in generality due to [Sh19]. The Quillen equivalence on the right is classical, going back to [Qu67], cf. [GJ99, §I.11])

Incidentally, under restriction to propositional types (93) the equivalence (106) is a homotopy-theoretic generalization of the classical fact that a proposition about data of type X is equivalently encoded in the sub-type of data satisfying this proposition: Semantically this is the existence of *sub-object classifiers* known from topos theory [La70] (see e.g. [Bo94III, §5.1])

Propositional Type Universe	**Sub-Object classifier**
Injections into X $(Y:\mathrm{Type})\times(Y\hookrightarrow X)$ $\xrightarrow[\text{total space}]{\substack{(Y,f)\mapsto(x\mapsto\mathrm{fib}_x(f))\\ \text{fibers}\\ \simeq\\ ((x:X)\times P_x,\mathrm{pr}_X)\ \leftarrow\!\mapsto\ (x\mapsto P_x)}}$ **Propositions about X** $(X\to\mathrm{Prop})$	$Y\longrightarrow\widehat{\mathrm{Obj}_{-1}}$ $\downarrow\ \ _{(\mathrm{pb})}\ \ \downarrow$ $X\longrightarrow\mathrm{Obj}_{-1}$	(108)
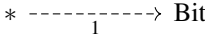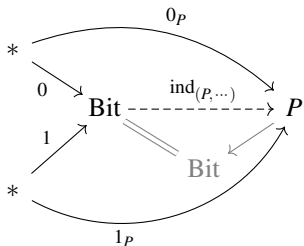
For example, given a data set $D:\mathrm{Set}$, then a *relation* on such data is given by the sub-type $R\hookrightarrow D\times D$ of those pairs of data which are in relation to each other, which is equivalently the proposition $D\times D\to\mathrm{Prop}$ asserting about any pair that its data are in relation to each other:

$$\big(R:\mathrm{Type}\big)\times\big(r:R\to D\times D\big)\times\Big((d_1,d_2:D)\to\mathrm{isProp}\big(\underbrace{\mathrm{fib}_{(d_1,d_2)}(r)}_{\substack{\textbf{proposition that}\\ d_1,d_2\ \textbf{are in relation}}}\big)\Big)\quad\xleftrightarrow{\ \sim\ }\quad\big(D\times D\to\mathrm{Prop}\big)\qquad(109)$$

---

[15] On this point see again footnote 6.

The idea of inductive types.   While we have seen how to construct new types from given ones — by forming dependent function types, dependent pair types (64) and identification types (71) — it remains to discuss how to introduce definite types in the first place.

The archetypical example is the type Bit : Type of Boolean truth values (often denoted Bool, instead). It is clear that its *term introduction rule* should say that there are two pieces of data of this type, namely $0, 1$: Bit. But it remains to introduce a language construct ensuring that there is *no other* data of this type. An elegant idea for achieving this is to declare that we obtain a function Bit $\to P$ to any other type D *as soon* as we have correspondingly two pieces of data $0_D, 1_D$: D. But a subtle point has to be taken care of for this and analogous *term elimination rules* to work as expected: Since we are working in the generality of *dependent* types (63), we need to declare this in the generality of *dependent functions*. But it turns out that it suffices to consider the case when D depends on Bit itself (e.g. [ML84, p. 35-37][AGS12, §3.1]):

Bits.		
	Homotopy type-theory	**Homotopy theory**
Formation	$\vdash \text{Bit : Type}$	
Introduction	$\vdash 0 : \text{Bit} \qquad \vdash 1 : \text{Bit}$	
Elimination	$b : \text{Bit} \ \vdash \ P(b) : \text{Type} ;$ $\vdash 0_P : P(0); \ \vdash 1_P : P(1)$ $\overline{b : \text{Bit} \vdash \text{ind}_{(P,0_P,1_P)}(b) : P(b)}$	
Computation	$\text{ind}_{(P,0_P,1_P)}(0) \ \equiv \ 0_P$ $\text{ind}_{(P,0_P,1_P)}(1) \ \equiv \ 1_P$	

(110)

In homotopy theory this defines the space which is the disjoint union of two points – the *0-sphere* – , universally characterized as the *coproduct* of two copies of the 1-point space $*$. Here the universal map out of the coproduct (the "term eliminator") is denoted "$\text{ind}_{(\cdots)}$" because this turns out to be an example of the same general notion of *induction* which also controls the classical notion of induction over the natural numbers (111). Therefore, one refers to this and analogous types as *inductive types* ([CP90][Dy94], early review in [PM93][Lu94, §9.2.2], exposition in [UFP13, §5.6]).

Concretely, there are two ways to introduce data of natural number type: On the one hand, there is certainly the datum $0 : \mathbb{N}$; on the other hand, if data $n : \mathbb{N}$ is already given, then there is the datum $\text{succ}(n) = n+1 : \mathbb{N}$. As before with the type of bits, the idea now is to enforce that these two introduction rules produce *all* $\mathbb{N}$-data by declaring that we obtain a function $\mathbb{N} \to D$ to any other type *as soon* as that type D is equipped with images 0_D and succ_D of these two constructors. Saying this in the generality that D is a type depending (63) on $\mathbb{N}$ yields the following inference rules (111) for natural numbers [ML84, pp. 38][CP90, p. 52-53][Dy94, §3]:

	Homotopy type-theory	**Homotopy theory**
Formation	$$\frac{}{\ \vdash \mathbb{N} : \mathrm{Type}}$$	
Introduction	$$\frac{}{\ \vdash 0 : \mathbb{N}} \qquad \frac{\vdash n : \mathbb{N}}{\ \vdash \mathrm{succ}(n) : \mathbb{N}}$$	$* \sqcup \mathbb{N} \ \xdashrightarrow{\ (0,\mathrm{succ})\ } \ \mathbb{N}$
Elimination	$$\frac{\begin{array}{c} n : \mathbb{N} \ \vdash \ D(n) : \mathrm{Type} \\ \vdash \ 0_D : D(0) \\ n : \mathbb{N}, \ d : D(n) \ \vdash \ \mathrm{succ}_D(n,d) : D\big(\mathrm{succ}(n)\big) \end{array}}{n : \mathbb{N} \ \vdash \ \mathrm{ind}_{(D,0_D,\mathrm{succ}_D)}(n) : D(n)}$$	
Computation	$$\mathrm{ind}_{(D,0_D,\mathrm{succ}_D)}(0) \equiv 0 \ ; \\ \mathrm{ind}_{(D,0_D,\mathrm{succ}_D)}\big(\mathrm{succ}(n)\big) \\ \equiv \mathrm{succ}_D\big(n, \mathrm{ind}_{(D,0_D,\mathrm{succ}_D)}(n)\big)$$	

(Natural numbers. — Homotopy theory column, rows Elimination and Computation, contains commutative diagrams with arrows labelled $(0,\mathrm{succ})$, $\mathrm{id}_* \sqcup \mathrm{ind}$, $\mathrm{id}_* \sqcup \pi_D$, $(0_D,\mathrm{succ}_D)$, π_D, ind, between objects $* \sqcup \mathbb{N}$, $* \sqcup D$, $* \sqcup \mathbb{N}$, $\mathbb{N}$, D, $\mathbb{N}$.)

$$(111)$$

In the denotational semantics on the right we see that $\mathbb{N}$ has the structure of a (homotopy-)*initial algebra over the endofunctor* $* \sqcup \mathrm{Id}$ on spaces (i.e. the endofunctor which reflects the domains of a nullary constructor 0 and of a unary constructor succ). In general, the denotational semantics of "well-founded" inductive types ("$\mathscr{W}$-types" [ML84, pp. 43]) in homotopy theory is given by (homotopy-)initial algebras of polynomial endofunctors [Dy97][AGS12][AGS15].

The inductive rules for the natural number type capture the classical notions both of *proof of propositions by induction* and of *construction of functions by recursion*:

- $\mathbb{N}$-**Induction.** When the dependent type in (111) is propositional (95), so that $n : \mathbb{N} \ \vdash D(n)$: Prop is a *proposition about natural numbers* (93), then:
  - the assumption of the elimination rule (111) is that we have a certificate $0_P : P(0)$ – hence a *proof* of the proposition about 0 – and moreover with each certificate/proof $d : P(n)$ of the proposition about any n also a a proof $\mathrm{succ}_P : P(n+1)$ of the proposition about $n+1$;
  - in which case the conclusion of the elimination rule is a proof of the proposition for all natural numbers: $n : \mathbb{N} \ \vdash \mathrm{ind}(n) : P(n)$ – thereby recovering the classical induction principle.

- $\mathbb{N}$-**Recursion.** When D happens to be independent of $n : \mathbb{N}$ then the induction principle (111) is that of *recursive functions*. For example, addition and multiplication of natural numbers may be recursively defined as follows:

$$
\begin{array}{ll}
D :\equiv \mathbb{N} \to \mathbb{N} & D :\equiv \mathbb{N} \to \mathbb{N} \\
0_D :\equiv (k \mapsto k) & 0_D :\equiv (k \mapsto 0) \\
\dfrac{\mathrm{succ}_D(n,f) :\equiv \big(k \mapsto \mathrm{succ}(f(k))\big)}{+ \ :\equiv \mathrm{ind}_{(D,0_D,\mathrm{succ}_D)} : \mathbb{N} \to \big(\mathbb{N} \to \mathbb{N}\big)} & \dfrac{\mathrm{succ}_D(n,f) :\equiv \big(k \mapsto f(k) + k\big)}{\cdot \ :\equiv \mathrm{ind}_{(D,0_D,\mathrm{succ}_D)} : \mathbb{N} \to \big(\mathbb{N} \to \mathbb{N}\big)}
\end{array}
\qquad (112)
$$

Unique and non-existent data. Finally, the most simple but important examples of inductive types:
The *singleton type* (often: "unit type", e.g. [UFP13, p. 30]), whose induction rule witnesses it as a (necessarily contractible) type (90) with an unique datum:

$$
\frac{}{\ \vdash * : \mathrm{Type}} \qquad \frac{}{\ \vdash \bullet : *} \qquad \frac{x : * \ \vdash D(x) : \mathrm{Type} \ ; \quad \bullet_D : D(\bullet)}{x : * \ \vdash \mathrm{ind}_{(D,\bullet_D)}(x) : D(x)} \qquad \mathrm{ind}_{(D,\bullet_D)}(\bullet) = \bullet_D \qquad (113)
$$

The *empty type* is inductively generated by *no* data (e.g. [UFP13, §1.7, §A.8][Es19, §2.6]):

$$\frac{}{\vdash \varnothing : \text{Type}} \qquad \frac{x : \varnothing \;\vdash\; D(x) : \text{Type}}{x : \varnothing \vdash \text{ind}_D : D(x)} \quad (114)$$

This degenerate case of the general rules for inductive types gives the ancient logical rule of *ex falso quodlibet* — from absurdity we may derive anything — when we regard propositions as types (91). There can be no data of type $\varnothing$ since it would take *no* conditions to produce data $\text{ind}(x) : P$ of any type P (which is an absurdity) *assuming* $x : \varnothing$ as given (which hence must not exist). In practice, the induction principle of $\varnothing$ is used when doing a case analysis to escape from cases that cannot happen; a case that cannot happen will imply $\varnothing$, and so we may still prove our goal using the induction principle of $\varnothing$ in that case.

$$\frac{P : \text{Type}}{\text{const}_P :\equiv \big((x : \varnothing) \mapsto D\big) : \big(\varnothing \to \text{Type}\big)}$$

$$\text{ind}_{\text{const}_P} : (x : \varnothing) \to D$$

The idea of higher inductive types. But in a homotopically typed language, these induction principles for constructing concrete types are to be generalized to account for the introduction not just of plain data, but also of (re-)identitications (71) of such data ("higher inductive types" [UFP13, §6][vD18, §2.2.6][VMA19, §4]). For example, in practice, one often considers data that is either of some type Y or of some type Y', except that a datum $y : Y$ is meant to be identified with data $y' : Y'$ whenever the pair (y, y') (62) arises as the output of a given program

$$(f, f') : X \longmapsto Y \times Y'.$$

The type of such combined data is called the homotopy-*pushout* (or *cofiber coproduct*) of f and (along) f', denoted $Y^{f} \underset{X}{\sqcup}{}^{f'} Y'$ or usually just $Y \underset{X}{\sqcup} Y'$, for brevity. The inference rules for such homotopy-cofiber/pushout types (e.g. [HFLL1616, p. 4]) are an evident expression of their homotopy-theoretic interpretation as homotopy cofiber/pushout spaces (e.g. [Str11, §7.1][Ar11, §6]), as shown in (115):

(115)

49

The computation rule for the pushout is a little different than for the other inductive types. For the generatoring hmt, we do not have a definitional equality but rather an identity comp which identifies the application of the induction function ind applies to hmt with thmt$_P$, the homotopy in the codomain. The function apd is the dependent version of ap (81), and is defined by path induction in the same way:

$$f : (c : C) \to D(c) \quad \vdash \quad \mathrm{apd}_f : (c_1, c_2 : C) \longrightarrow \left((p : \mathrm{Id}_D(c_1, c_2)) \quad \to \quad \mathrm{Id}_{D(c_2)}\big(p_* f(c_1), f(c_2)\big) \right) \tag{116}$$
$$\mathrm{id}_{c_1} \qquad \mapsto \qquad \mathrm{id}_{f(c_1)}$$

For more on the computation rules of higher inductive types, see [UFP13, §6]. In other variants of homotopy type theory such as *cubical type theory* ([CCHM15] [CHM18]), the computation rules for the pushout and other higher inductive types can be given by definitional equalities like those for other inductive types.

The idea of higher homotopy bits. As an example of a pushout, the (un-reduced) *suspension* of a type X, is the homotopy pushout (115) of $X \xrightarrow{x \mapsto \bullet} *$ (113) along itself, hence the type where each datum $x : X$ is promoted to a certificate of identification of a data pair $(\mathrm{nth}, \mathrm{sth})$:

Suspensions.	
Homotopy type-theory	**Homotopy theory**
$S(X) :\equiv * \sqcup_X * : \text{Type}$	(diagram)
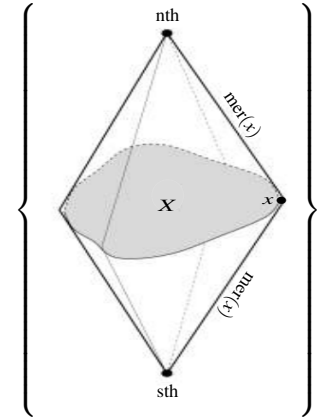
$$SX \quad = \tag{117}$$

Indicated on the right[16] is the corresponding semantics as the topological suspension space (e.g. [Ja84, p. 41][Ha02, p. 8]). For example, the suspension (117) of the empty type (114) is inductively generated by two terms with *no* homotopy between them — this is equivalently the type of bits: $S\varnothing \simeq \text{Bit}$ (110).

Next, the suspension of the type of bits is freely generated by (1.) a pair of data points with (2.) a pair of identifications between them. This is the homotopy type of the circle, cf. the illustration in (119).

Inductively, if $X = S^n$ is an n-sphere homotopy type, then its suspension is the $n + 1$-sphere homotopy type, realized as the union of (1.) a pair of poles nth, sth and (2.) the *meridians* $\mathrm{mer}(s)$ through all points s in the *equator* n-sphere.

n-Spheres.	
Homotopy type-theory	**Homotopy theory**
$S^{-1} :\equiv \varnothing$	(diagram)
$S^0 :\equiv \text{Bit}$	
$S^{n+1} :\equiv SS^n$	

$$\tag{118}$$

In other words, in homotopy-typed programming languages, the archetypical type Bit of bits (110) is accompanied by a tower of higher homotopy types, obtained as its iterated suspensions (117) via $\mathbb{N}$-induction (111):

By (130) below, these higher sphere types serve to detect all the higher homotopy nature of types. In this sense, homotopy-typed programming is all about *generalizing bits to "higher homotopy bits"*:
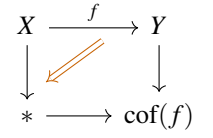
While this perspective on homotopy type-theory is mathematically as compelling as it is intriguing, the practical content of "computation on higher homotopy bits" in actual computer science (as opposed to its interpretation in mathematical homotopy theory) has arguably remained somewhat elusive. Our claim in §6 is that (certification of) topological quantum computation fills this gap.

Higher homotopy bits		
Classical bits	**Circle type**	**Sphere type**
$S^0 = \text{Bit}$	$S^1 = S\,\text{Bit}$	$S^2 = S^2\,\text{Bit}$
(diagram)	(diagram)	(diagram)
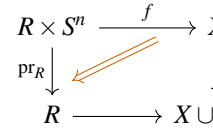
$$\tag{119}$$

16The graphics on the right of (117) is adapted from [Mu10, p. 14].

Homotopy cofibers, CW-complexes, and sequential colimits. Another important special case of homotopy pushouts (115) are *homotopy cofibers* (the notion "dual" to homotopy fibers (87)), *cell attachments* (e.g. [AGP02, §3.1]) and the resulting cell complexes (specifically "CW-complexes", e.g. [Ha02, pp. 5][AGP02, §5.1]):
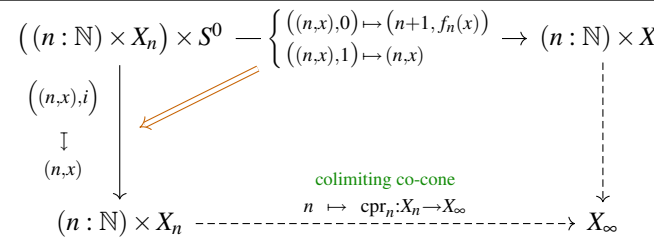
The **homotopy cofiber** of a function $f : Y \to X$ is the special case of the cofiber coproduct type (115) where one summand is the singleton type (113):	$\mathrm{cof}(f) :\equiv * \underset{X}{\sqcup} Y$	$\begin{array}{ccc} X & \xrightarrow{f} & Y \\ \downarrow & & \downarrow \\ * & \longrightarrow & \mathrm{cof}(f) \end{array}$	(120)

A single n-**cell attachment** to a type X is a cofiber (120) of a function $f : S^n \to X$ out of the n-sphere type (118).	$\mathrm{cof}(S^n \xrightarrow{f} X)$	$\begin{array}{ccc} S^n & \xrightarrow{f} & X \\ \downarrow & & \downarrow \\ * & \longrightarrow & X \cup_f D^n \end{array}$	(121)
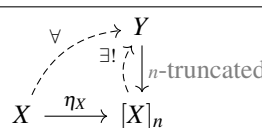
More generally: An indexed n-**cell attachment** to a type X is such a pushout (115) relative to a parameter type R.	$\mathrm{po}\left(\begin{array}{ccc} R \times S^n & \xrightarrow{f} & X \\ \downarrow{\scriptstyle \mathrm{pr}_R} & & \\ R & & \end{array} \right)$	$\begin{array}{ccc} R \times S^n & \xrightarrow{f} & X \\ \downarrow{\scriptstyle \mathrm{pr}_R} & & \downarrow \\ R & \longrightarrow & X \cup_f D^n \end{array}$	(122)

A **finite CW-complex** is obtained from the empty type (114) by a finite sequence of cell attachments (121) of increasing dimension $n_{k+1} \geq n_k$	$\begin{aligned} X^{-1} &:\equiv \varnothing \\ X^{k+1} &:\equiv \mathrm{cof}\left(S^{n_{k+1}} \xrightarrow{f^{(k)}} X^k\right) \\ X &:\equiv X^{k_{\max}} \end{aligned}$	$\begin{array}{c} \varnothing \hookrightarrow \cdots \hookrightarrow \\ X^k \hookrightarrow X^k \cup_{f^{(k)}} D^{n_{k+1}} =: X^{k+1} \\ \hookrightarrow \cdots \hookrightarrow X \end{array}$	(123)

A **finite set** is a finite CW-complex (123) of dimension 0.	$\begin{aligned} X^{-1} &:\equiv \varnothing \\ X^{k+1} &:\equiv \mathrm{cof}(\varnothing \to X^k) \\ X &:\equiv X^{k_{\max}} \end{aligned}$	$\left\{ x_1, x_2, \cdots, x_n \right\}$	(124)

A **sequential colimit** over functions $n : \mathbb{N} \vdash X_n : \mathrm{Type}$, $n : \mathbb{N} \vdash f_n : X_n \to X_{n+1}$ indexed by the natural numbers (111) is also obtained by a homotopy pushout (115) (e.g. [Ri18, §15]).	$\begin{array}{ccc} \left((n : \mathbb{N}) \times X_n\right) \times S^0 & \xrightarrow{\begin{cases} ((n,x),0) \mapsto (n+1, f_n(x)) \\ ((n,x),1) \mapsto (n,x) \end{cases}} & (n : \mathbb{N}) \times X_n \\ {\scriptstyle \left((n,x),i\right)} \downarrow {\scriptstyle \updownarrow} & & \vdots \\ {\scriptstyle (n,x)} & & \vdots \\ (n : \mathbb{N}) \times X_n & \xrightarrow[\quad n \,\mapsto\, \mathrm{cpr}_n : X_n \to X_\infty \quad]{\text{colimiting co-cone}} & X_\infty \end{array}$	(125)

The idea of homotopy n-truncation of types. When $X : \mathrm{Type}$ is not necessarily n-truncated (100), we may construct (e.g. as in [UFP13, §7.3, p. 223]) its *n-truncation* $[X]_n : \mathrm{Type}_{\leq n}$: which is the "best approximation" to X by a type that *is* n-truncated, in that it comes with a function of the form $\eta_n : X \to [X]_n$ which is *initial* among functions from X to n-truncated types, namely uniquely factoring any such function up to equivalence (101). As before with other inductive types, it is natural to state such a *universal mapping property* (ump) in the generality of dependent types and dependent functions, where it looks as follows:

Homotopy type theory	Homotopy theory	
$Y : [X]_n \to \mathrm{Type}_{\leq n} \;\vdash\; \mathrm{ump} : \mathrm{isEquiv}\left(\left((-) \circ \eta_n\right) : \prod_{c:[X]_n} Y(c) \to \prod_{x:X} Y\left(\eta_n(x)\right) \right)$	$\begin{array}{ccc} & & Y \\ & \nearrow^{\forall} \; {\scriptstyle \exists!} \nearrow & \downarrow {\scriptstyle n\text{-truncated}} \\ X & \xrightarrow{\eta_X} & [X]_n \end{array}$	(126)

For example, since any n-truncated type is also $(n+1)$-truncated, these factorizations yield for each $X : \mathrm{Type}$ a tower of truncations

$$X \xrightarrow{\quad} \cdots \xrightarrow{\quad} [X]_{n+1} \xrightarrow{\quad\overset{\eta_n}{\frown}\quad} [X]_n \xrightarrow{\quad} \cdots \xrightarrow{\quad} [X]_0 \xrightarrow{\quad} [X]_{-1} \equiv: \exists X \qquad (127)$$

which in the homotopy theoretic semantics is known as the *Postnikov tower* ([GJ99, Cor. 3.7][Lu09, §5.5.6 & §6.5]). Classical homotopy theory has been interested mostly in the higher stages of the Postnikov tower, but homotopy type theory brings out that its low stages are of profound *logical* relevance:

- (-1)-truncation of a type X – also called *propositional truncation* [AB04][UFP13, §3.7][Kr15] – may be understood as producing the proposition (91) that *there exists data of type X* (that X is "inhabited"), cf. (92):

$$
\begin{array}{|c|c|}
\hline
\textbf{Type theory} & \textbf{Logic} \\
\hline
\Gamma \;\vdash\; \exists X := [X]_{-1} \,:\, \mathrm{Prop} & \exists_{x \in X} \\
\hline
\end{array}
\tag{128}
$$

  The universal property (126) of propositional truncation is the usual rule for using an existential quantifier in a proof: to prove a proposition P assuming $\exists X$, we may assume we have an $x : X$. Explicitly, if P is any proposition then to prove $\exists X \to P$ it suffices by the universal property to give a function $X \to P$ which (like any function) may be defined by assuming an element $x : X$ and then proving P.

- 0-truncation of a type X – also called *set truncation* – may be understood as producing the set (96) of equivalence classes of X-data; semantically this is the passage to the set π_0 of connected components of a space, and for dependent types all this applies fiber-wise (106), cf. Lem 4.6:

$$
\begin{array}{|c|c|}
\hline
\textbf{Homotopy type theory} & \textbf{Homotopy theory} \\
\hline
\Gamma \;\vdash\; [X]_0 \,:\, \mathrm{Set} & \pi_{0/\Gamma}(X) \\
\hline
\end{array}
\tag{129}
$$

Cell-complex construction of n-truncation. While in homotopy type theory it is popular, following [UFP13, §7.3, p. 223], to construct n-truncation (126) in one step as a clever higher inductive type construction; we now highlight an alternative construction of n-truncation, which is implicit in [Ri19] and whose semantics is closer to the classical construction of n-truncations in homotopy theory.

First notice that from unwinding the definition of *n-truncation* (100), *looping* (88) and *higher spheres* (118) one finds ([UFP13, Thm. 7.2.9][CR21, Thm. 3.10]), in close analogy to classical homotopy theoretic arguments, that a type is n-truncated precisely if its $(n+1)$-fold loopings are contractible (90), and that an $n+1$-fold looping is equivalently the function type out of the $n+1$-sphere preserving the given base datum ([UFP13, Lem. 6.5.4]):

$$
\begin{array}{l}
n : \mathbb{N}, \\
X : \mathrm{Type}
\end{array}
\;\vdash\;
\mathrm{isType}_{\leq n}(X) \;\simeq\; \Big((x : X) \to \big(\exists! \, \Omega_x^{n+1} X \big) \Big) \;\simeq\; \Big((x : X) \to (f : S^{N+1} \to X) \times \mathrm{Id}\big(f(\mathrm{nth}), x \big) \Big).
\tag{130}
$$

Therefore one may expect that the n-truncation $[X]_n$ is obtained by adjoining trivializations of all $f : S^{n+1} \to X$, via the following indexed cell-attachment (122):

$$\tag{131}$$

While this is the right idea, the result X_1 may still fail to be n-truncated., but it is getting closer: We must pass to the colimit of iterating this construction (following [Ri19, §7.2]):

Definition 5.1 (Truncation). For $X : \mathrm{Type}$ and $n : \mathbb{N}$ we define $\eta_n : X \to [X]_n$ to be the colimiting co-cone of the sequential colimit (125)

$$
[X]_n \;:\equiv\; X_\infty \;=\; \mathrm{colim}\big(X :\equiv X_0 \xrightarrow{i_0} X_1 \xrightarrow{i_1} X_2 \xrightarrow{i_2} \cdots \big), \qquad \eta_n :\equiv \mathrm{cpr}_0 \,:\, X \to [X]_n
\tag{132}
$$

over $n+2$-cell attachments i_k that are indexed (122) by the functions $S^{n+1} \to X_k$ into the previous stage of the sequence[17]

$$\tag{133}$$

---

[17]In classical homotopy theory the construction (133) is expressed by explicitly attaching sets of cells of ever higher dimension $n+2+k$, thereby iteratively filling up the higher homotopy groups. Here in the type-theoretic formulation this increase in dimension is *implicit* in the fact that the cell attachments are indexed not just by a set, but by the higher homotopy type $S^{n+1} \to X_k$ (i.e. by "internalizing" the classical construction). For example, applying the construction for $n = -1$ to $X_0 = *$ yields in the first stage $X_1 = S^1$, whence the index type for the second stage is the 2-type $S^0 \to S^1 \simeq S^1 \times S^1$.

Proposition 5.2 (Universal property for *n*-truncation)**.** *The construction of* $\eta_n : X \to [X]_n$ *from Def. 5.1 has the universal property* (126) *of the n-truncation.*

Proof. This is a special case of the construction of the localization at a class of maps between compact types given in [Ri19, Thm. 7.2.10]. Specifically, the pushout (133) is a special case of the initial quasi-*F*-local extension ([Ri19, Defn. 7.2.6]) where the family F is taken to be the terminal map $S^{n+1} \to *$ to the singleton type (113). Then [Ri19, Thm. 7.2.10] applies since the spheres are sequentially compact by [Ri19, Cor. 7.1.12].

The universal property given in [Ri19, Thm. 7.2.10] is not the dependent one we gave in (126). However, a variety of similar universal properties which characterize *modalities* ([UFP13, §7.7]) are proven equivalent in [RSS20, §1]; these include the universal property (126) as [RSS20, Defn. 1.2] and the universal property appearing in [Ri19, Thm. 7.2.10] as [RSS20, Defn. 1.3] (noting that *n*-truncated types are closed under pair types, [UFP13, Thm. 7.1.8]). □

Quotient sets. The presence of higher inductive type formation (pp. 49) works wonders even when all types involved are sets (96) as opposed to higher homotopy types (100); namely it implies the existence of types encoding ordinary *quotient sets* of data. This solves a decade-old problem in type theory (review in [Li14, §1.1][Mu22, §4.3.2]) which goes back to the roots of the notion of *sets* in constructive mathematics [Bish67, p. 13][BB85, p. 15] as sets equipped with equivalence relations, also called "setoids" [Ho95, §1.3, §5.1][BCP03] or "Bishop sets" [CS07, §3.1].

Concretely, given a data set (96) equipped with a relation (109) which is an *equivalence relation* in that the following conditions are satisfied:

$$
\begin{aligned}
X &: \text{Set} & \text{isReflexive}(P) &:\equiv (x:X) \to P(x,x)\,, \\
P &: X \times X \to \text{Prop}, & \text{isSymmetric}(P) &:\equiv (x_0, x_1 : P) \times P(x_0, x_1) \to P(x_1, x_0)\,, \\
R &:\equiv (x_0, x_1 : X) \times P(x_0, x_1), & \text{isTransitive}(P) &:\equiv (x_0, x_1, x_2 : X) \times P(x_0, x_1) \times P(x_1, x_2) \to P(x_0, x_3)\,;
\end{aligned}
\tag{134}
$$

then we obtain the *quotient type* X/R of X by R, defined (cf. [UFP13, §6.10][RS15, §1.5, §2.4]) as the set truncation of the pushout (115) of the function which picks the pairs of data of type X that are in relation, along the function which projects out their distinction in $S^0 = \text{Bit}$ (119):

$$
\cdots \;\; \vdash \;\; X/R \;\; :\equiv \;\; \left[\text{po} \left(\begin{array}{c} ((x_0, x_1), i) \;\; \mapsto \;\; x_i \\ R \times S^0 \longrightarrow X \\ \downarrow{\scriptstyle \text{pr}_R} \\ R \end{array} \right) \right]_0 \;\; : \;\; \text{Set}
\tag{135}
$$

We could also define the quotient X/R as the set of *equivalence classes* of the equivalence relation R. An equivalence class is a property $E : X \to \text{Prop}$ so that there exists an $x : X$ with $E(y)$ if and only if $P(x,y)$ for all y:

$$
\text{isEquivalenceClass}(E) \;:\equiv\; \exists \Big((x:X) \times \big((y:X) \to (E(y) \xrightarrow{\sim} R(x,y)) \big) \Big)\,.
\tag{136}
$$

Then we may define

$$
\cdots \vdash X/R \;:\equiv\; \{ E : X \to \text{Prop} \mid \text{isEquivalenceClass}(E) \}.
\tag{137}
$$

A multitude of examples arises in the construction of the hierarchy of types of *number systems* which we come to in §6.

The idea of certified data structures. We had motivated, around (60), the notion of data typing by the promise of software verification; this now becomes nicely manifest:

Via iteration of dependent data pairings (65) (called *data telescopes* [Zu75][dB91], *data records* [CT08, p. 2] or type *classes* [GGMR09, §2]) of *data base* types B : Set (96) with B-dependent functions (64) constituting read/write/compute-operations on B data ("methods") and further with identification certificates (71) constituting data consistency statements, one obtains fully verifiable *data structures*, whose denotational semantics is just that of "mathematical structures" (see e.g. pointers in [Sak20]) in the orignal sense [Co04] of algebra (often referred to as: "the hierarchy of structures" or similar).

Data structure	Mathematical structure
Data base	Underlying set
Data access methods	Algebraic structure
Consistency specification	Laws/ properties

For example, data bases B equipped with methods to consistently read/write given D-data (known as well-behaved D-"lens"-structure [BPV06, §3]) are of the following form:

$$
\text{\textcolor{red}{\textbf{\textit{D}-lens structure on data base}}} \quad D\text{Lens} :\equiv \left\{
\begin{array}{l}
(B : \text{Set}) \;\}\; \text{\textcolor{red}{\textbf{data base}}} \\[4pt]
\times\; (\text{read}_D : B \to D) \\[4pt]
\times\; (\text{write}_D : D \times B \to B) \left.\vphantom{\begin{array}{l}a\\a\\a\end{array}}\right\} \text{\textcolor{red}{\textbf{access structure}}} \\[4pt]
\times\; \left(\text{rw} : \binom{b:B}{d:D} \to \text{Id}_D\Big(\text{read}_D\big(\text{write}_D(d,b)\big),\, d\Big) \right) \\[4pt]
\times\; \left(\text{wr} : (b : B) \to \text{Id}_B\Big(\text{write}_D\big(\text{read}_D(b),\, b\big),\, b\Big) \right) \left.\vphantom{\begin{array}{l}a\\a\end{array}}\right\} \text{\textcolor{red}{\textbf{consistency specification}}}
\end{array}
\right.
\tag{138}
$$

Notice that to give a D-lensed data structure (138) means, by the pair type introduction rule (65),

$$
(\underbrace{B}_{\text{\textcolor{red}{data base}}},\, \underbrace{\text{read}_D,\, \text{write}_D}_{\text{\textcolor{red}{methods}}},\, \underbrace{\text{rm},\, \text{rw}}_{\text{\textcolor{red}{certificates}}}) \;:\; D\text{Lens}
\tag{139}
$$

to instantiate the data base B equipped with its read/write methods *and* with certificates that these work as expected.

It is in this way that fully data-typed programs (60) are automatically *certified* and *verified*: To produce data of a given structured type necessarily involves supplying a certificate that correct data behaviour has been verified.

Group structure. A data structure of fundamental mathematical relevance is *group data structure* (e.g. [Ka09][Es19, §33.10][1lab, §"Group theory"]) in the sense of abstract group theory (e.g. [Mi72, §1.1][Ro95]):

$$
\text{\textcolor{red}{\textbf{group data structure}}} \quad \text{Grp} :\equiv \left\{
\begin{array}{l}
(G : \text{Set}) \;\}\; \text{\textcolor{red}{\textbf{data base}}} \\[4pt]
\times\; (\text{e} : G) \\[4pt]
\times\; (\cdot : G \times G \to G) \left.\vphantom{\begin{array}{l}a\\a\\a\end{array}}\right\} \text{\textcolor{red}{\textbf{group structure}}} \\[4pt]
\times\; ((-)^{-1} : G \to G) \\[4pt]
\times\; \left(\text{unt} : (g : G) \to \text{Id}_G\big(g\cdot\text{e},\, g\big) \times \text{Id}_G\big(\text{e}\cdot g,\, g\big) \right) \\[4pt]
\times\; \left(\text{asc} : \binom{g_1, g_2,}{g_3 : G} \to \text{Id}_G\big((g_1 \cdot g_2)\cdot g_3,\, g_1 \cdot (g_2 \cdot g_3)\big) \right) \left.\vphantom{\begin{array}{l}a\\a\\a\end{array}}\right\} \text{\textcolor{red}{\textbf{group laws}}} \\[4pt]
\times\; \left(\text{inv} : (g : G) \to \text{Id}_G\big(g\cdot g^{-1},\, \text{e}\big) \times \text{Id}_G\big(g^{-1}\cdot g,\, \text{e}\big) \right)
\end{array}
\right. .
\tag{140}
$$

Historically, the mathematical term "group" is short for *symmetry group* (e.g. [Mi72]) or *transformation group* (e.g. [tD87]) in the sense of: *group of transformational symmetries of some object* (cf. [tD87, (2.3)]). Curiously this meaning is natively brought out by the *magic of homotopy type theory* (cf. [BBCDG21]): From the discussion on p 41 it is clear that the self-identifications $d_0 \rightsquigarrow d_0$ (symmetries) of any datum $d_0 : D$ in a groupoid $D : \text{Type}_{\leq 1}$ (99) – hence the data in its loop type (88) at d_0 – form a group (140) under concatenation (77):

$$
D : \text{Type}_{\leq 1},\;\; d_0 : D \qquad \vdash \qquad \big(\underbrace{\Omega_{d_0} D},\;\; \underbrace{\text{e} :\equiv \text{id}_{d_0}},\;\; \underbrace{\cdot :\equiv \text{conc}},\;\; \underbrace{(-)^{-1} :\equiv \text{inv}},\;\; \underbrace{\cdots} \big) \quad:\quad \text{Grp}
\tag{141}
$$

Further in this vein: **subgroup data structure** $H \subset G$ (cf. e.g. [Mi72, §1.2]) of a given group structure G (140) may be formulated as the *propositions* (93) of the form $P_H : G \to \text{Prop}$ (which we may think of as) asserting: $P_H : g \mapsto$ "g is in $H \subset G$", equipped with certificates that these propositions do define subgroups (cf. [Es19, §33.12]):

$$
(G, \text{e}, \cdot, (-)^{-1}) : \text{Grp} \quad \vdash \quad \text{\textcolor{red}{\textbf{subgroup structure}}} \;\; \text{SubGrp}(G) :\equiv \left\{
\begin{array}{l}
(P : G \to \text{Prop}) \;\}\; \text{\textcolor{red}{structure}} \\[4pt]
\times\; (\text{ptd} : P(\text{e})) \\[4pt]
\times\; \left(\text{mld} : \big((g_1, g_2 : G) \times P(g_1) \times P(g_2)\big) \to P(g_1 \times g_2) \right) \left.\vphantom{\begin{array}{l}a\\a\\a\end{array}}\right\} \text{\textcolor{red}{properties}} \\[4pt]
\times\; \left(\text{ivd} : \big((g : G) \times P(g)\big) \to P\big(g^{-1}\big) \right)
\end{array}
\right.
\tag{142}
$$

From such $P : G \to \mathrm{Prop}$ the actual subgroup data is recovered (108) as the dependent pairings $H :\equiv (g : G) \times P_H(G)$ (elements of G paired with a certificate that they are in fact in the subgroup), which inherits group structure by using the group operations on G paired with the certificates that on H they do restrict to land again in H:

$$
\left(G, \mathrm{e}, \cdot, (-)^{-1}\right) : \mathrm{Grp} \;\vdash\; \mathrm{SubGrp}(G) \xrightarrow{\;\text{\textcolor{green}{underlying abstract group structure}}\;} \mathrm{Grp}
$$

$$
(P, \mathrm{ptd}, \mathrm{mld}, \mathrm{ivd}) \;\longmapsto\;
\begin{pmatrix}
H & :\equiv (g : G) \times P(g), \\
\mathrm{e}_H & :\equiv (\mathrm{e}, \mathrm{ptd}), \\
\cdot_H & :\equiv \big((g_1, p_1), (g_2, p_2) : H\big) \mapsto \big(g_1 \cdot g_2, \mathrm{mld}(g_1, g_2, p_1, p_2)\big), \\
(-)_H^{-1} & :\equiv \big((g, p) : H\big) \mapsto \big(g^{-1}, \mathrm{ivd}(g, p)\big)
\end{pmatrix}
\tag{143}
$$

The idea of data structure identification. With data structures (p. 53) defined as telescopes of dependently paired dependent functions and identification, we can make explicit their operational *equivalences* (103) or equivalently their identifications (105) as soon as we have an explicit handle on the identification of any dependent functions and dependent pairs.

Using the univalence axiom (105), these work component-wise as expected, via comparison functions readily defined by Id-induction (73), a statement known as *function extensionality* [Vo10, p. 8][UFP13, §4.9, Thm. 2.9.7] and its analogue for pairings [UFP13, Thm 2.7.2]):

$$
\textcolor{orange}{\substack{\text{function} \\ \text{extensionality}}} \quad
\begin{pmatrix} f, g \\ {:(d:D) \to C_d} \end{pmatrix} \to
\left(
\begin{array}{ccc}
\overset{\textcolor{blue}{\text{identifications of dependent functions}}}{\mathrm{Id}_{(d:D) \to C_d}(f, g)} & \xrightarrow{\textcolor{green}{\text{equivalent to}} \;\sim\;} & \overset{\textcolor{blue}{\text{dependent functions of identifications}}}{\Big((d : D) \to \mathrm{Id}_{C_d}\big(f(d), g(d)\big)\Big)} \\
\mathrm{id}_f & \longmapsto & \big(d \mapsto \mathrm{id}_{f(d)}\big)
\end{array}
\right)
\tag{144}
$$

$$
\textcolor{orange}{\substack{\text{pair} \\ \text{extensionality}}} \quad
\begin{pmatrix} (d,c), (d',c') \\ {:(d:D) \times C_d} \end{pmatrix} \to
\left(
\begin{array}{ccc}
\overset{\textcolor{blue}{\text{identifications of dependent pairs}}}{\mathrm{Id}_{(d:D) \times C_d}\big((d,c), (d',c')\big)} & \xrightarrow{\textcolor{green}{\text{equivalent to}} \;\sim\;} & \overset{\textcolor{blue}{\text{dependent pairs of identifications}}}{\big(p : \mathrm{Id}_D(d, d')\big) \times \mathrm{Id}_{C_{d'}}\big(p_* c, c'\big)} \\
\mathrm{id}_{(d,c)} & \longmapsto & \big(\mathrm{id}_d, \mathrm{id}_c\big)
\end{array}
\right)
\tag{145}
$$

By iteration of these two rules, the equivalences of telescope/record data structures (p. 53) are found to be those equivalences on the underlying base types which are compatible with all the given structure – hence which are *homo-morphisms*, thus recovering the original algebraic notion of *iso-morphism*.

For example, an equivalence of group data structures (140) comes out to be a *bijective group homo-morphism* (e.g. [Mi72, §1.3]), hence a *group iso-morphism* just as in the traditional algebraic sense, but here *emergent* from the principles of homotopy type theory:

$$
\overset{\textcolor{orange}{\substack{\text{identifications of} \\ \text{types with group structure}}}}{\mathrm{Id}_{\mathrm{Grp}}\Big((G, \mathrm{e}, \mathrm{cmp}, \mathrm{inv}), (G', \mathrm{e}', \mathrm{cmp}', \mathrm{inv}')\Big)} \;\xrightarrow{\;\sim\;}\;
\left\{
\begin{array}{l}
\big(p : \mathrm{Id}_{\mathrm{Set}}(G, G')\big) \;\; \textcolor{blue}{\substack{\text{underlying} \\ \text{bijection}}} \\[4pt]
\times\, \mathrm{Id}_G(p_* \mathrm{e}, \mathrm{e}') \\[4pt]
\times\, \mathrm{Id}_{G \times G \to G}(p_* \mathrm{cmp}, \mathrm{cmp}') \quad \textcolor{red}{\substack{\text{homomorphism} \\ \text{property}}} \\[4pt]
\times\, \mathrm{Id}_{G \to G}(p_* \mathrm{inv}, \mathrm{inv}')
\end{array}
\right.
\tag{146}
$$

Essentially this state of affairs (144) - (146) in univalent homotopy type theory was first made explicit in [CD13] and has come to be known as the "*structure identity principle*" (following [Ac11]), further discussed in [UFP13, §9.8] [Es19, §§3.33.1][ANST20] (see also in relation to *structuralism* in [Ts17, §5.1]). Notice that, thereby, dependent type theory pleasantly resolves a long historical quest (cf. [Co04]) for a good meta-theory of "mathematical structures", along the way unifying it with the notion of "data structures".

Type theory	**Homotopy theory**
types	Homotopy types
Data structures	Math. structures
Data bases	Carrier types

The idea of higher structures. The previous examples of data/mathematical structures (pp. 53) and many of those considered further below are based on (data-)*sets* (96), this being the classical situation traditionally considered in the literature. But in a homotopy-typed language we may just as well consider *higher homotopy types* (100) as base types, to obtain "higher structures" (in the now popular sense, see e.g. [CGX01][JSSW19][FSS19]).

Fundamental examples of higher structures are group *deloopings* (147) and higher deloopings (192); these we turn to in §5.2 below.

In summary, the remarkable insight of homotopy type theory as a statement in computer science may be expressed as follows:

Any programming language with truly thorough data typing
is natively a verification language for constructions
in algebraic topology and homotopy theory.

We next turn to show how this allows for a slick construction of homotopy types of Gauss-Manin monodromy in general (§5.2) and then of anyonic topological quantum gates in particular (§6).

5.2 Monodromy of cohomological data

We now describe the programming language construct (Def. 5.16 below) which, under the dictionary in §5.1, encodes the monodromy of Gauss-Manin connections on twisted cohomology groups as developed in §4, and hence (below in §6) specifically the operation of anyon braid gates via monodromy of KZ-connections.

As a type formation in itself, our main construction in Def. 5.16 below is rather immediate: It is nothing but the dependent 0-truncation of a doubly iterated dependent function type between (higher) delooping types — the reader with insider knowledge in homotopy type theory will not need any further introduction to parse Def. 5.16 and may want to jump ahead.

Recall that this type-theoretic simplicity is our main point: Under the dictionary of §5.1 the transport operation (74) in this readily constructed type of Def. 5.16 clearly interprets as what in §4 we showed is the parallel transport by Gauss-Manin connections — whose traditional construction however is rather less immediate (Lit. 2.23, Lit. 2.24).

However, to be self-contained to a broader audience and since the type-theoretic literature on the following issues remains thin, we first proceed now with laying out some type-theoretic foundations regarding what one might call the theory of *transformation groups* or *abstract Galois theory*. If nothing else, what follows may serve as an illustrative example for how to work concretely (albeit "informally" in the style of [UFP13]) with the homotopy type language of §5.1 (cf. Rem. 5.7 below).

Groups of self-identifications. In (141) we saw that a natural source of group structures (140) are the "loopings" (88) of pointed types, $\Omega_{d_0} D : \mathrm{Grp}$. Since this only depends on the single datum $d_0 : D$ on which the self-identifications of these loops are based, then given any $G : \mathrm{Grp}$ it makes sense to ask for a 1-type (99) – to be denoted $\mathbf{B}G$ and called a *delooping* of G – for which there is an essentially unique datum in the first place and whose self-identifications recover G in this way (141)

$$G \simeq \Omega_* \mathbf{B}G$$

Proposition 5.9 below asserts (in particular) that all groups arise this way, up to equivalence, This leads to an alternative slick definition of group types which is only available in homotopy-typed languages: the *pointed connected 1-types*:

$$\begin{array}{r} \text{\color{red}\textbf{groups as pointed}} \\ \text{\color{red}\textbf{connected 1-types}} \end{array} \quad \mathrm{Type}_{0 < \bullet \leq 1}^{*/} :\equiv \left\{ \begin{array}{ll} \mathbf{B}G : \mathrm{Type}_{\leq 1} & \} \text{ \color{red}\textbf{higher data}} \\ \times \ \mathrm{pt} : \mathbf{B}G & \} \text{ \color{red}\textbf{structure}} \\ \times \ (t : \mathbf{B}G) \to \exists \mathrm{Id}_{\mathbf{B}G}(\mathrm{pt}, t) & \} \text{ \color{red}\textbf{property}} \text{ (connectivity)} \end{array} \right. \tag{147}$$

Semantically, this alternative homotopy-theoretic conception of groups, and its equivalence (Prop. 5.9) to the algebraic definition is the content of the "May recognition theorem" for loop spaces [May72] generalized to groups internal to ∞-toposes [Lu09, Lem. 7.2.2.11] and as such much amplified in [NSS12a, p. 7][SS20-Orb, Prop.][SS21-Bun, Prop. 0.2.1], cf. (174) below. The perspective has been picked up by the type-theoretic literature in [BvDR18, p. 6][BBCDG21, §4] (see also [Wä]).

In order to prove this equivalence in type theoretic detail (Prop. 5.9 below), we step back and lay out all the ingredients:

G-Actions and torsors. For a given group structure G there is a classical notion of its *actions* on sets (e.g. [tD87, (1.1)]):

Definition 5.3 (G-Sets). For G a group (140), a *left G-action structure on a set* — or just *left G-set* for short — is data of the following type:

$$(G, \mathrm{e}, \cdot, (-)^{-1}) : \mathrm{Grp} \ \vdash \ \begin{array}{c} \text{\scriptsize left} \\ \text{\color{red}\scriptsize G-action} \\ \text{\color{red}\scriptsize / G-set} \end{array} \ G\,\mathrm{Act}_L :\equiv \left\{ \begin{array}{ll} (S : \mathrm{Set}) \ \} \text{ \color{red}\textbf{data}} \\ \times \ (\circlearrowleft : G \times S \to S) \ \} \text{ \color{red}\textbf{structure}} \\ \times \ \left(\mathrm{unt} : (s : S) \to \mathrm{Id}(\mathrm{e} \circlearrowleft s, s) \right) \\ \times \ \left(\mathrm{act} : (g_1, g_2 : G) \to \mathrm{Id}\Big((g_2 \cdot g_1) \circlearrowleft s, \ g_2 \circlearrowleft (g_1 \circlearrowleft s) \Big) \right) \end{array} \right\} \text{\color{red}\textbf{properties}} \tag{148}$$

Correspondingly there are *right G-sets*, a mild distinction which does and will matter in some applications:

$$(G, \mathrm{e}, \cdot, (-)^{-1}) : \mathrm{Grp} \ \vdash \ \begin{array}{c} \text{\scriptsize right} \\ \text{\color{red}\scriptsize G-action} \\ \text{\color{red}\scriptsize / G-set} \end{array} \ G\,\mathrm{Act}_R :\equiv \left\{ \begin{array}{ll} (S : \mathrm{Set}) \ \} \text{ \color{red}\textbf{data}} \\ \times \ (\circlearrowright : S \times G \to S) \ \} \text{ \color{red}\textbf{structure}} \\ \times \ \left(\mathrm{unt} : (s : S) \to \mathrm{Id}(s \circlearrowright \mathrm{e}, s) \right) \\ \times \ \left(\mathrm{act} : (g_1, g_2 : G) \to \mathrm{Id}\Big(s \circlearrowright (g_1 \cdot g_2), \ (s \circlearrowright g_1) \circlearrowright g_2 \Big) \right) \end{array} \right\} \text{\color{red}\textbf{properties}} \tag{149}$$

We will follow the usual convention of referring to a G-set by the name of its underlying set, often leaving the structure and properties implicit.

A function between G-sets (148) is *equivariant* if and only if it commutes with the action of G, and we denote the data set of equivariant maps by $\mathrm{Hom}_G(X,Y)$:

$$
\left.\begin{array}{l}
G : \mathrm{Grp}, \\
(S, \circlearrowleft), (T, \circlearrowleft) : G\,\mathrm{Act}_L
\end{array}\right\}
\begin{array}{l}
\phi : S \to T \;\vdash\; \mathrm{isEquivariant}(\varphi) \;:\equiv\; (g:G) \times (x:S) \to \mathrm{Id}_Y\big(\varphi(g \circlearrowleft s), g \circlearrowleft \varphi(s)\big) \\[6pt]
\qquad\qquad \vdash\; \mathrm{Hom}_G(X,Y) \;:\equiv\; \big\{\varphi : S \to T \mid \mathrm{isEquivariant}(\varphi)\big\}.
\end{array}
\tag{150}
$$

Via homotopy theory, the type of G-actions (5.3) has a slick reformulation (Prop. 5.10 below) in line with the delooping equivalence of Prop. 5.9.

The following is the evident type-theoretic formulation of the classical notion of G-torsors (e.g. [Miln80, §3.4], cf. [BBCDG21, Def. 4.8.1]):

Definition 5.4 (G-Torsors). For G a group (140), a *left G-torsor structure T* is an inhabited (128) left G-set $G \circlearrowleft T$ (148) whose action is regular (i.e., free and transitive) in that for any two elements $x, y : T$ there is a unique $g : G$ for which $g \circlearrowleft x = y$.

$$
\big(G, \mathrm{e}, \cdot, (-)^{-1}\big) : \mathrm{Grp} \;\vdash\; {\color{red}{G\text{-torsor}}}\;\; G\mathrm{Tors}_L \;:\equiv\;
\left\{
\begin{array}{ll}
(T, \circlearrowleft, \mathrm{unt}, \mathrm{act}) : G\mathrm{Act} & {\color{blue}{G\text{-action which}}} \\[4pt]
\times\, (x, y : T) \to \exists!\big\{g : G \mid \mathrm{Id}_T(g \circlearrowleft x, y)\big\} & {\color{blue}{\text{is regular}}} \\[4pt]
\times\, \exists T & {\color{blue}{\text{and inhabited.}}}
\end{array}
\right.
\tag{151}
$$

For example, the left/right multiplication action of any group on itself (i.e.: on its own underlying data set) makes a left/right G-torsor:

$$
\big(G\,\mathrm{e}, (-)^{-1}\big) : \mathrm{Grp}, \quad \big(T \equiv G, \circlearrowleft :\equiv \cdot, \mathrm{act} :\equiv \mathrm{asc}\big) : G\mathrm{Tor}_L
\tag{152}
$$

In fact, up to equivalence (101) this is the *only* example of a G-torsor: Every G-torsor is isomorphic to the canonical one (152), and the choice of isomorphism amounts to choosing which of its elements is identified with the neutral element of G. This standard fact is re-proven type-theoretically as Lem. 5.6 below; it is the main reason we care about G-torsors at this point, because it implies that the type of G-torsors (151) is (up to equivalence) the delooping (147) of G (Prop. 5.9).

To this end, first we need to see that every equivariant function between G torsors is an equivalence. The following proof of this statement is a standard argument, but to showcase how the type-theoretic rules surveyed in §5.1 are at work, we spell out this proof in more detail. The upshot however is that the formal rules allow reasoning just as one informally expects, which justifies leaving them more implicit as we proceed (cf. Rem. 5.7 below).

Lemma 5.5 (Equivariant maps between torsors are isomorphisms). *Let G be a group* (140) *and S and T be left G-torsors (definition 5.4). Then any equivariant map $\varphi : \mathrm{Hom}_G(S,T)$ (150) is an isomorphism, in that we have an equivalence* (101)

$$
\mathrm{Hom}_G(S,T) \;\simeq\; \mathrm{Id}_{G\mathrm{Tors}_L}(S,T)
$$

of equivariant maps (150) *with identifications* (71) *of G-torsors* (151), *under which composition of homomorphisms corresponds to concatenation* (77) *of identifications.*

Proof. We need to construct a dependent term of the following form

$$
G : \mathrm{Grp}, \;\; S, T : G\mathrm{Tor}_L, \;\; \phi : \mathrm{Hom}_G(S,T) \;\;\vdash\;\; \mathrm{proof}(\phi) : \mathrm{isBijection}(\phi).
$$

With that in hand the statement will follow fairly readily by the structure-identity-principle and using univalence.

Unwinding the definition of $\mathrm{isBijection}(-)$ (101), our goal is a term of this form:

$$
S, T : G\mathrm{Tor}_L, \;\; \phi : \mathrm{Hom}_G(S,T) \;\;\vdash\;\; \mathrm{proof}(\phi) : (t : T) \to \exists!\Big((s : S) \times \mathrm{Id}\big(\phi(s), t\big)\Big),
$$

where we are now notationally suppressing the assumption of the group G on the left, just for brevity.

But by the rules for dependent function types (68), such may be inferred from a term of this form:

$$
S, T : G\mathrm{Tor}_L, \;\; \phi : \mathrm{Hom}_G(S,T), \;\; t : T \;\;\vdash\;\; \mathrm{proof}(\phi) : \exists!\Big((s : S) \times \mathrm{Id}\big(\phi(s), t\big)\Big).
$$

So far this is closely analogous to classical reasoning. But in a subsequent step we will need to get our hands on an element of S, which type-theoretically is a little more subtle than classically: Namely what we nominally have in the assumptions on

the left – as part of the assumption that S is a torsor (151) – is only a term $p : \exists S$. Making this notationally explicit, we are really looking for

$$S, T : G\mathrm{Tor}_L, \ \phi : \mathrm{Hom}_G(S, T), \ t : T, \ p : \exists S \ \vdash \ \mathrm{proof}(\phi) : \exists! \Big((s : S) \times \mathrm{Id}\big(\phi(s), t\big) \Big).$$

By itself, the hypothesis $p : \exists S$ just says that "there is" such an element, but does not in generally allow us to actually put such an element into the context. What saves the day here is that the *right* hand side of the above dependent term is a proposition (91). Therefore the universal property (126) of propositional truncation applies to show that a term of the above form actually is equivalent to a term of this form:

$$S, T : G\mathrm{Tor}_L, \ \phi : \mathrm{Hom}_G(S, T), \ t : T, \ s : S \ \vdash \ \mathrm{proof}(\phi) : \exists! \Big((s : S) \times \mathrm{Id}\big(\phi(s), t\big) \Big).$$

Now unwinding the definition of $\exists!(-)$ on the right, our goal is finally in the following form (where on the right we are notationally suppressing, for readability, an iterated identification type, as that is contractible anyway in the present case since S is a set (96)):

$$S, T : G\mathrm{Tor}_L, \ \phi : \mathrm{Hom}_G(S, T), \ t : T, \ s : S$$
$$\vdash \ \mathrm{proof}(\phi) : \Big((s_t : S) \times \mathrm{Id}\big(\phi(s_t), t\big) \times \big((s'_t : S) \times \mathrm{Id}\big(\phi(s'_t), t\big) \to \mathrm{Id}(s_t, s'_t) \big) \Big). \tag{153}$$

Notice how this expression, when read out aloud, is pretty much the statement that a classical proof of the lemma would *start* with: It says that given a $t : T$ we need to show that there is an s_t in its pre-image under ϕ — where we are allowed to assume that we have *some* $s : S$ —, and that any other element s'_t in the preimage of t is identifiable with s_t. Accordingly, from here the classical proof idea applies essentially verbatim, using the torsor property of S and T (notationally hidden in the assumptions on the left):

First, by the regularity of the G-action on T there is $g_t : G$ such that $g_t \mathbin{\text{$\zeta$}} \varphi(s) = t$; and by equivariance of φ this implies $\varphi(g_t \mathbin{\text{ζ}} s) = t$. Hence we may take s_t in (153) to be

$$t : T, \ s : S \ \vdash \ s_t :\equiv g_t \mathbin{\text{ζ}} s \ : \ S.$$

Finally, if $s'_t : S$ with $\varphi(s'_t) = t$ is any other pre-image of t, then by the same reasoning we have $g'_t : G$ with $g'_t \mathbin{\text{$\zeta$}} s = s'_t$, and hence $g'_t \mathbin{\text{$\zeta$}} \varphi(s) = t$. But since an equation of this form also defined g_t, regularity of the G-action implies that $g'_t = g_t$; and therefore that $s'_t = g_t \mathbin{\text{ζ}} s = s_t$. $\qquad\square$

Delooping of groups and classification of principal bundles. One of the crown jewels of classical homotopy theory is the proof (good review is in [RS17, Thm. 3.5.1]) that, under mild conditions on a topological group G there exists a topological space BG such that homotopy classes of maps $X \to BG$ are in bijection to isomorphism classes of "principal G-bundles" over X, meaning: fiber bundles of G-torsors (Def. 5.4) — whence one speaks of the *classifying space BG*.
A modern way to re-prove this classical theorem is (see [SS21-Bun, p. 7]) to observe:

(i) **(a)** that before passage to their isomorphism classes, the *groupoid* (99) of G-principal bundles is equivalent to that of maps of "topological stacks" $X \to \mathbf{B}G$, where $\mathbf{B}G$ is the "delooping stack" of G,

    (b) which is thereby identified with the "moduli stack" of principal G-bundles, hence of G-torsors: $\mathbf{B}G \simeq G\mathrm{Tors}$;

(ii) that the classifying space BG is the underlying *cohesive shape* (cf. p. 1) of $\mathbf{B}G$.[18]

This modernized re-proof of classifying space theory makes it essentially a formality if only one has a good abstract grasp on (cohesive) homotopy theory — as is provided by (cohesive) homotopy type theory. (In particular, the local triviality condition on fiber bundles does not need to be declared as an explicit condition, but is implied by the topos-theoretic semantics of univalent HoTT; this is discussed in [SS21-Bun, §4.2]).

Type theory	**Geometric homotopy theory**			
$x : X \ \vdash \ P_x : G\mathrm{Tors}_L$	$G \mathbin{\curvearrowright} P \longrightarrow G\mathrm{Tors}^{G/} \quad \simeq \quad \mathbf{E}G \quad \simeq \quad *$ $\downarrow \qquad\qquad \text{(pb)} \qquad \downarrow \qquad\qquad \downarrow$ $X \longrightarrow G\mathrm{Tors} \quad \simeq \quad \mathbf{B}G$			

(where the column headers in blue read: **principal G-bundle aka: G-torsor over X**, **bundle of pointed G-torsors**, **universal principal G-bundle**, **moduli stack of G-torsors**, **delooping of group stack**) (154)

[18]It is for this reason that we use the boldface $\mathbf{B}(-)$ for "delooping" (147), following [Sc13, p. 74] (cf. [SS21-Bun, p. 7]), to indicate that it is a topological/cohesive *enrichment* of what classically is denoted BG – which notation, in turn, is a historical memory of its first construction, known as the "bar construction".

In this spirit we now indicate the formal type-theoretic proof of the first item above (the cohesive aspect in the second item is beyond the scope of the present text, but see the outlook on p. 4). This is the content of Prop. 5.9 below, for which we first establish Lemmas 5.6 and 5.8. While these should not be surprising to the experts, the result is of profound importance and currently not citable in this form from the literature (though see [BBCDG21, §4.13] for a less structural proof of Prop. 5.9).

Lemma 5.6 (The type of torsors deloops a group)**.** *For G a group type (140), the type $G\mathrm{Tors}_L$ of G-torsors (Def. 5.4) is a pointed connected 1-type (141), pointed by $G : G\mathrm{Tors}$ (152)*

$$G : \mathrm{Grp} \quad \vdash \quad \big(G\mathrm{Tors}_L, \mathrm{pt} :\equiv G\big) : \mathrm{Type}^{*/}_{0 < \bullet \leq 1}, \tag{155}$$

whose loop type (88) is G:

$$G : \mathrm{Grp} \quad \vdash \quad p : \mathrm{Id}_{\mathrm{Grp}}\big(\Omega_G(G\mathrm{Tors}), G\big). \tag{156}$$

Proof. First to see that the type of torsors is 0-connected, namely to provide a certificate of the form required in the last line in (147):

$$T : G\mathrm{Tors}_L \quad \vdash \quad \exists \mathrm{Id}_{G\mathrm{Tors}_L}(G, T).$$

Applying the type equivalence of Lem. 5.5, it suffices to show that there exists an equivariant map $\varphi_T : \mathrm{Hom}_G(G, T)$:

$$T : G\mathrm{Tors}_L \quad \vdash \quad \exists \mathrm{Hom}_G(G, T).$$

Making here explicit the inhabitation certificate (151) provided by a G-torsor structure on the left, this is

$$T : G\mathrm{Tors}_L, \ p : \exists T \quad \vdash \quad \exists \mathrm{Hom}_G(G, T).$$

But since on the right we need a proof of the proposition $\exists(-)$ which is given by (-1)-truncation (128), its universal property (126) implies that such dependent terms are equivalent to those where we assume a specific element $t : T$ on the left:

$$T : G\mathrm{Tors}_L, \ t : T \quad \vdash \quad \exists \mathrm{Hom}_G(G, T).$$

Moreover, applying the rules of function types (68) to the (-1)-truncation unit $\eta : \mathrm{Hom}_G(G, T) \to \exists \mathrm{Hom}_G(G, T)$ (127) means that we will infer such an existence proof from constructing one example:

$$T : G\mathrm{Tors}_L, \ t : T \quad \vdash \quad \phi_t : \mathrm{Hom}_G(G, T).$$

But such a term we obtained by setting

$$\varphi_t(g) \ :\equiv \ g \,\zeta\, t \tag{157}$$

and providing the following certificate that this function is equivariant (150), where we use concatenation of equalities (98):

$$g', g : G \quad \vdash \quad \phi_t(g' \cdot g) \underset{(157)}{=} (g' \cdot g) \,\zeta\, t \overset{\mathrm{act}_T}{\underset{(148)}{=}} g' \,\zeta\, (g \,\zeta\, t) \underset{(157)}{=} g' \,\zeta\, \phi_t(g) \ : \ \mathrm{Id}_T\Big(\phi_t(g' \cdot g), g' \,\zeta\, \phi_t(g)\Big).$$

In summary, this provides the connectivity certificate required in (147). The 1-truncation certificate also required there will immediately follow from the proof of the second claim (156), to which we now turn (since G is 0-truncated and by the recursive definition of n-truncation (99)).

The proof of (156), after unwinding the definition of looping (88), is to be a dependent term of the form

$$G : \mathrm{Grp} \quad \vdash \quad \mathrm{proof}_G : \mathrm{Id}_{\mathrm{Grp}}\big(\mathrm{Id}_{G\mathrm{Tors}}(G, G), G\big).$$

By applying the type equivalence of Lem. 5.5 our goal is equivalently a term of the following type:

$$G : \mathrm{Grp} \quad \vdash \quad \mathrm{proof}_G : \mathrm{Id}_{\mathrm{Grp}}\big(\mathrm{Hom}_G(G, G), G\big).$$

Moreover, by the type equivalence of univalence (105), we equivalently need to produce an inverted function between group types

$$G : \mathrm{Grp} \quad \vdash \quad \mathrm{proof}_G :\equiv \big(f_G, f_G^{-1}, \cdots\big) : \mathrm{Hom}_G(G, G) \overset{\sim}{\longrightarrow} G.$$

Now take this function f_G to be given by evaluation at the neutral element:

$$\phi : \mathrm{Hom}_G(G, G) \quad \vdash \quad f_G(\phi) :\equiv \phi(\mathrm{e}) \ : \ G \tag{158}$$

and take its reverse f_G^{-1} to be

$$h : G \;\;\vdash\;\; f_G^{-1}(h) \;:\equiv\; (h \mapsto h \cdot g) \;:\; G \longrightarrow \mathrm{Hom}_G(G, G).$$ (159)

The certificates that this does constitute a pair of inverse functions follow, via the composition rule for equality certificates (98), just the way one would prove this classically:

$$f \circ f^{-1}(g) \underset{(159)}{=} f(h \mapsto h \cdot g) \underset{(158)}{=} \mathrm{e} \cdot g \overset{\mathrm{unt}}{\underset{(148)}{=\!=}} g \;:\; \mathrm{Id}(f \circ f^{-1}(g), g)$$

$$f^{-1} \circ f(\phi) \underset{(158)}{=} f^{-1}\big(\phi(\mathrm{e})\big) \underset{(159)}{=} \big(h \mapsto h \,\zeta\, \phi(\mathrm{e})\big) \underset{(150)}{=} \big(h \mapsto \phi(h \,\zeta\, \mathrm{e})\big) \overset{\mathrm{unt}}{\underset{(148)}{=\!=}} \big(h \mapsto \phi(h)\big) \underset{(68)}{=} \phi \;:\; \mathrm{Id}\big(f^{-1} \circ f(\phi), \phi\big).$$

This completes the proof of (156) and with it that of (155). □

Remark 5.7 (Perspective on proof). The above proofs of Lem. 5.5 and Lem. 5.6 should serve to illustrate how type-theoretic constructions proceed (of course, many more illustrative examples may be found in the literature listed at Lit. 2.27). Therefore, in the following proofs we shall omit the fine-grained manipulation of dependent terms and just indicate enough of the proof strategy that obtaining a fully formal type-theoretic proof is straightforward, if maybe tedious.

Of course, this is exactly the style in which all rigorous math has been communicated during the last century, it being understood that in principle all rigorous human-readable proofs could be transformed (straightforwardly, if tediously) into fully formal proofs in a classical logical foundation like ZFC set theory, if desired. The big difference however is that there is little reason to desire a formal proof in ZFC set theory, while here in the new foundations of homotopy type theory, proofs are computer programs whose construction may be highly desirable, such as the construction of a program evaluating topological quantum gate execution that we are headed towards.

Lemma 5.8 (Any delooping is equivalent to the type of torsors). *For G a group (140) and $(\mathbf{B}G, \mathrm{pt})$ any delooping (147), then the function which sends data t of type $\mathbf{B}G$ to its identification type with pt extends to a type equivalence (101) between $\mathbf{B}G$ and the type of G-torsors (Def. 5.4):*

$$
\begin{array}{l}
G : \mathrm{Grp}, \;\; \mathbf{B}G : \mathrm{Type}^{*/}_{0 < \bullet \leq 1}, \\
\ell : \big(\Omega_{\mathrm{pt}}\mathbf{B}G, \mathrm{id}_{\mathrm{pt}}, \mathrm{conc}\big) \overset{\sim}{\to} (G, \mathrm{e}, \cdot)
\end{array}
\;\;\vdash\;\;
\begin{array}{ccc}
\mathbf{B}G & \overset{\sim}{\longrightarrow} & G\mathrm{Tors}_L \\
t & \longmapsto & \big(\mathrm{Id}_{\mathbf{B}G}(\mathrm{pt}, t), \mathrm{conc}\big),
\end{array}
$$ (160)

where the G-action on $\mathrm{Id}(\mathrm{pt}, t)$ is given by concatenation of identifications, under the given looping equivalence ℓ:

$$
\begin{array}{ccccc}
G \times \mathrm{Id}_{\mathbf{B}G}(\mathrm{pt}, t) & \overset{\sim}{\longrightarrow} & \mathrm{Id}_{\mathbf{B}G}\big(\mathrm{pt}, \mathrm{pt}\big) \times \mathrm{Id}_{\mathbf{B}G}(\mathrm{pt}, t) & \overset{\zeta}{\longrightarrow} & \mathrm{Id}_{\mathbf{B}G}(\mathrm{pt}, t) \\
\big(g, \mathrm{pt} \overset{p}{\rightsquigarrow} t\big) & \longmapsto & \big(\mathrm{pt} \overset{\ell_g}{\rightsquigarrow} \mathrm{pt}, \mathrm{pt} \overset{p}{\rightsquigarrow} t\big) & \longmapsto & \mathrm{pt} \overset{\mathrm{conc}(\ell_g, p)}{\rightsquigarrow} t
\end{array}
$$

Proof. First to see that the construction (160) really produces torsors:

The inhabitation condition, $\exists \mathrm{Id}(\mathrm{pt}, t)$, is part of the connectivity certificate that comes with a pointed connected type (147). Moreover, given a pair of torsor elements, then using their inversion (76) and concatenation (77) produces a group element taking one to the other, under the above action:

$$
p, q : \mathrm{Id}(\mathrm{pt}, t) \;\vdash\; g \;:\longmapsto\; \mathrm{pt} \overset{\ell_g \,:\equiv\, \mathrm{conc}(p, \mathrm{inv}(q))}{\rightsquigarrow} \mathrm{pt} \;:\; \mathrm{Id}(\mathrm{pt}, \mathrm{pt})
$$
(161)

$$
p, q : \mathrm{Id}(\mathrm{pt}, t) \;\vdash\; g \,\zeta\, q = p \;:\; \mathrm{Id}\big(g \,\zeta\, q, p\big).
$$

Finally, this group element is unique, because for another $g' : G$ with $g' \,\zeta\, q = p$, we have

$$\mathrm{conc}(\ell_g, q) = p = \mathrm{conc}(\ell_{g'}, q) \;:\; \mathrm{Id}\big(\mathrm{pt}, t\big)$$ (162)

and hence

$$\ell_g \underset{(78)}{=} \mathrm{conc}\big(\mathrm{conc}(\ell_g, q), \mathrm{inv}(q)\big) \underset{(162)}{=} \mathrm{conc}\big(\mathrm{conc}(\ell_{g'}, q), \mathrm{inv}(q)\big) \underset{(78)}{=} \ell_{g'} \;:\; \mathrm{Id}\big(\mathrm{pt}, \mathrm{pt}\big).$$

Second, that this function (160) is an equivalence follows from the fundamental lemma of pointed connected types [My22, Thm. 2.1.1]: If a function between pointed connected types induces an equivalence on loop types, then it is itself an equivalence. □

We now have the tools in hand to state the inverse equivalence to the looping operation (141):

Proposition 5.9 (The looping-delooping equivalence). *There is an equivalence* (101) *between the type* Grp *of algebraically defined groups* (140) *and the type* $\mathrm{Type}_{1\bullet}^{>0}$ (147) *of pointed connected 1-types:*

$$\mathrm{Type}_{0<\bullet\leq 1}^{*/} \;\;\xleftarrow[\underset{\Omega_{\mathrm{pt}}(-)}{\xrightarrow{\;\sim\;}}]{\mathbf{B}(-)\,:\equiv\,(-)\mathrm{Tors}_L}\;\; \mathrm{Grp} \tag{163}$$
$$(D,\mathrm{pt}) \qquad\qquad \longmapsto \qquad\qquad \Omega_{\mathrm{pt}}D$$

In one direction this equivalence is given by the looping construction Ω_{pt} (141)*; in the other direction by sending G to its type* $G\mathrm{Tors}_L$ *of left G-torsors* (5.4)*.*

Another type-theoretic proof of this statement was earlier given in [LF14], using higher inductive types (groupoid quotients [VW21]). We may now offer a more structural proof, using delooping by the type of torsors:

Proof. It is sufficient to show that the two functions are inverses of each other, up to re-identification (101), which is now immediate from the previous lemmas: In one direction we have, by lemma 5.6:

$$\big(\Omega_G \circ (-)\mathrm{Tors}_L\big)(G) \;\equiv\; \Omega_G\big(G\mathrm{Tors}_L\big) \underset{(156)}{\;\tilde\simeq\;} G\,.$$

In the other direction we have, using lemma 5.6 and lemma 5.8

$$\big((-)\mathrm{Tors}_L \circ \Omega_{\mathrm{pt}}\big)(\mathbf{B}G) \;\equiv\; \big(\Omega_{\mathrm{pt}}\mathbf{B}G\big)\mathrm{Tors}_L \underset{(156)}{\;\tilde\simeq\;} G\mathrm{Tors}_L \underset{(160)}{\;\tilde\simeq\;} \mathbf{B}G\,. \qquad\square$$

We can now reframe the type of G-actions (Def. 5.3) simply as types dependent on a delooping of G – this is where the right G-actions show up.

Proposition 5.10 (Functions of $\mathbf{B}G$ are G-Actions). *Given* G : Grp (140)*, the type of right G-sets (Def. 5.3) is equivalent* (101) *to the type of functions from a delooping* $\mathbf{B}G$ *(Prop. 5.9) to* Set (96)*:*

$$(\mathbf{B}G \to \mathrm{Set}) \xrightarrow{\quad\sim\quad} G\mathrm{Set}_R \tag{164}$$
$$f \qquad\longmapsto\qquad \big(S :\equiv f(\mathrm{pt}),\; s \circlearrowright g :\equiv g_*(s)\big)$$

Proof. By the type equivalence $\mathbf{B}G \simeq G\mathrm{Tors}_L$ from lemma 5.8, and recalling that this identifies pt : $\mathbf{B}G$ with G : $G\mathrm{Tors}_L$, we may equivalently show that the following function is an equivalence

$$F \;:\; (G\mathrm{Tors}_L \to \mathrm{Set}) \xrightarrow{\qquad\qquad} G\mathrm{Set}_R \tag{165}$$
$$f \qquad\longmapsto\qquad \big(S :\equiv f(G),\; s \circlearrowright g :\equiv (h \mapsto h \cdot g)_*(s)\big)\,.$$

We will now do so by constructing an explicit inverse to this function and then showing that it is an inverse:

$$F^{-1} \;:\; G\mathrm{Set}_R \xrightarrow{\qquad\qquad} (G\mathrm{Tors}_L \to \mathrm{Set}) \tag{166}$$
$$(X, \circlearrowright) \qquad\longmapsto\qquad \big((T, \circlearrowleft) \mapsto X \otimes_G T\big)\,,$$

where the tensor product "$\otimes_G$" (classically denoted "$\times_G$", cf. [SS21-Bun, (1.17)] and references given there) denotes the following quotient set (135):

$$X : G\mathrm{Set}_R,\;\; T : G\mathrm{Set}_L \;\;\vdash\;\; X \otimes_G T :\equiv \frac{X \times T}{(x \circlearrowright g, t) \sim (x, g \circlearrowleft t)}\,. \tag{167}$$

As usual, we will denote also the images under the quotient map by the tensor product symbol

$$\mathrm{cpr} \;:\; X \times Y \xrightarrow{\qquad\qquad} X \otimes_G T \tag{168}$$
$$(x, t) \qquad\longmapsto\qquad x \otimes t$$

62

Notice that/how this implies the expected equality certificates (97) of this form:

$$x:X,\; t:T,\; g:G \quad\vdash\quad (x \mathbin{⤸} g)\otimes t \underset{(168)}{=} \mathrm{cpr}\big((x \mathbin{⤸} g),t\big) \underset{(167)}{=} \mathrm{cpr}\big(x,\, g \mathbin{Ċ} t\big) \underset{(168)}{=} x\otimes(g \mathbin{Ċ} t) \;:\; \mathrm{Id}\big(\cdots,\cdots\big) \tag{169}$$

Now to prove the claim, we construct homotopies (101) which exibit F^{-1} (166) as a homotopy-inverse to F (165) from both sides. From one side, we need dependent equivalences of this form:

$$(X,⤸):G\mathrm{Tors}_R \quad\vdash\quad F\circ F^{-1}(X,⤸) \equiv F\big((T,Ċ)\mapsto X\otimes_G T\big) \equiv \big(X\otimes_G G,⤸\big) \simeq (X,⤸). \tag{170}$$

By the structure identity principle, it will suffice to construct a comparison function which is G-equivariant (150) and prove that it is an isomorphism. Consider this one:

$$(X,⤸):G\mathrm{Tors}_R \quad\vdash\quad
\begin{array}{ccc}
X & \longrightarrow & X\otimes_G G\\[4pt]
x & \mapsto & x\otimes e
\end{array} \tag{171}$$

To see that this is indeed equivariant, we first need to describe the G-action on the left: Since G is identified with $\mathrm{Hom}_G(G,G)$ via right multiplication, the induced action on $X\otimes_G G$ is also by right multiplication:

$$(x\otimes h)\mathbin{⤸} g = x\otimes(h\cdot g). \tag{172}$$

This can be checked by Id-induction (73), considering how the type family $X\otimes_G(-)$ acts on an identification $\varphi:\mathrm{Id}_{G\mathrm{Tors}_L}(G,T)$ and seeing that it holds for $\mathrm{id}_G:\mathrm{Id}_{G\mathrm{Tors}_L}(G,G)$. From this we find the certificate that (171) is indeed equivariant, by composing equality certificates, as follows (98):

$$(x\mathbin{⤸}g)\otimes e \underset{(169)}{=} x\otimes(g\mathbin{Ċ}e) \underset{(152)}{=} x\otimes(g\cdot e) \underset{(140)}{=} x\otimes(e\cdot g) \underset{(169)}{=} (x\otimes e)\mathbin{⤸}g.$$

This establishes that (171) is equivariant and hence invertible (Lem. 5.5). We may also readily define the inverse: it is given by $x\otimes g\mapsto x\mathbin{⤸}g$, which is well defined since $x\mathbin{⤸}(g\cdot h)=(x\mathbin{⤸}g)\mathbin{⤸}h$.

From the other side, we need to construct dependent equivalences of this form:

$$f:G\mathrm{Tors}_L\to\mathrm{Set} \quad\vdash\quad F^{-1}\circ F(f) \equiv F^{-1}\big(f(G)\big) \equiv \big((T,Ċ)\mapsto f(G)\otimes_G T\big) \simeq \big((T,Ċ)\mapsto f(T,Ċ)\big) \equiv f.$$

By (univalence and) function extensionality (144), this is obtained by constructing T-dependent bijections between the arguments of the functions on the right — consider this one:

$$\begin{array}{ccc}
f(G)\otimes_G T & \longrightarrow & f(T)\\[4pt]
x\otimes t & \mapsto & (\varphi_t)_*(x),
\end{array}$$

where $\varphi_t:\mathrm{Hom}_G(G,T)$ is the equivariant isomorphism $\varphi_t(h):\equiv h\mathbin{Ċ}t$ and $(\varphi_t)_*:f(G)\to f(T)$ is the transport along this identification in f. This map is well defined on the tensor product quotient since $(\varphi_{g\mathbin{Ċ}t})_*(x)=(\varphi_t\circ(h\mapsto hg))_*(x)=(\varphi_t)_*(h\mapsto hg)_*(x)=:(\varphi_t)_*(x\mathbin{⤸}g)$, recalling that the right action on $f(G)$ is given by $x\mathbin{⤸}g:\equiv(h\mapsto hg)_*(x)$ and making use of the functoriality of transport (83). It remains, then, to show that this map is a bijection.

Let $y:f(T)$, seeking to show that it has a unique inverse image in $f(G)\otimes_G T$. Since we are trying to prove the proposition $\exists!\mathrm{fib}_{x\otimes t\mapsto(\varphi_t)_*(x)}(y)$, we may assume we have an element $t:T$ (since by assumption we have $p:\exists T$). We then have that $y=(\varphi_t)_*((\varphi_t^{-1})_*(y))$, so that y has an inverse image given by $(\varphi_t^{-1})_*(y)\otimes t$. Now suppose that $y=(\varphi_s)_*(x)$, seeking to show that $x\otimes s=(\varphi_t^{-1})_*(y)\otimes t$. Since T is a torsor, there is a unique $g:G$ for which $g\mathbin{Ċ}t=s$; therefore, $\varphi_s=\varphi_t\circ(h\mapsto hg)$, which rearranges to give us that $(h\mapsto hg)=\varphi_t^{-1}\circ\varphi_s$. This means that $x\mathbin{⤸}g=:(h\mapsto hg)_*(x)=(\varphi_t^{-1}\circ\varphi_s)_*(x)=(\varphi_t^{-1})_*(y)$, so that $x\otimes s=x\otimes(g\mathbin{Ċ}t)=(x\mathbin{⤸}g)\otimes t=(\varphi_t^{-1})_*(y)\otimes t$. $\qquad\square$

Notation 5.11. It will be useful to have generic, concise, and suggestive notation for the inverse construction to the equivalence (164). We will denote this as follows:

$$\begin{array}{ccc}
G\mathrm{Set} & \longrightarrow & (\mathbf{B}G\to\mathrm{Set})\\[4pt]
(S,\,Ċ) & \longmapsto & \big((G\mathbin{Ċ}S):t\mapsto Ċ_t S\big)
\end{array} \tag{173}$$

Remark 5.12 (Choice of delooping). Since (by Prop. 5.3) an action of G is equivalently a function $\mathbf{B}G \to \mathrm{Set}$, and since (by Prop. 5.9) all deloopings of G are equivalent, it often pays to tailor the choice of delooping to the sorts of actions relevant for a given construction. We will see this in action when we construct the delooping $\mathbf{B}R^\times$ of the group of units (188) of a ring R as the type of R-modules (186) identifiable with R – in (190) below.

G-∞-Actions. The equivalent characterization of group actions from Prop. 5.10 has the striking advantage that it immediately generalizes to a notion of G-actions on any (higher) homotopy type $\mathscr{A}$: simply as those functions $\mathbf{B}G \to \mathrm{Type}$ (61) which take $\mathrm{pt} : \mathbf{B}G$ to $\mathscr{A}$. Under the pertinent identifications (69) this are nothing but the $\mathbf{B}G$-dependent types (cf. [BvDR18, §4.2]):

(174)

What is essentially a syntactic triviality on the left translates semantically, as indicated on the right, to the rather profound notion of "infinitely homotopy coherent actions" or "∞-actions" of (sheaves of) groups on ∞-stacks, see [NSS12a, §4][SS20-Orb, §2.2][SS21-Bun, §3.23] for details and further references. In particular, this shows that the type of dependent pairs which constitutes the "total space" (106) of a $\mathbf{B}G$-dependent type is semantically the *homotopy quotient* of the corresponding ∞-action, while the corresponding dependent function type is semantically the *homotopy fixed locus*.

Concretely, the canonical model for the G-homotopy quotient $(-) /\!\!/ G$ in topological spaces (4) is given (under mild conditions, certainly when G is discrete) by the *Borel construction* $(-) \times_G EG$, which manifestly forms the topological $\mathscr{A}$-fiber bundle that is *associated* to the universal G-principal bundle:

Homotopy type theory	**Homotopy theory**
$\vdash \ (t : \mathbf{B}G) \times (\circlearrowleft_t \mathscr{A}) \ : \ \mathrm{Type}$	$\mathscr{A} \times_G EG$ **Borel construction**

(175)

Delooping of group homomorphisms. A similar construction as in the proof of Prop. 5.10 shows that group homomorphisms

$$\big(G, \mathrm{e}_G, \cdot_G, (-)_G^{-1}\big), \ \big(H, \mathrm{e}_H, \cdot_H, (-)_H^{-1}\big) : \mathrm{Grp} \ \vdash$$

$$
\underset{\substack{\text{Group} \\ \text{homomorphisms}}}{} \big(G \underset{\mathrm{hom}}{\to} H\big) \ :\equiv \
\left\{
\begin{array}{l}
\big(\phi : G \to H\big) \quad \substack{\text{map of} \\ \text{underlying data}} \\[4pt]
\times \ \big(\mathrm{unt} : \mathrm{Id}\big(\phi(\mathrm{e}_G), \mathrm{e}_H\big)\big) \\[4pt]
\times \ \big(\mathrm{hom} : (g_1, g_2 : G) \to \mathrm{Id}\big(\phi(g_1 \cdot_G g_2), \ \phi(g_1) \cdot_H \phi(g_2)\big)\big)
\end{array}
\right\}
\substack{\text{respecting} \\ \text{group structure}}
$$

(176)

are equivalently pointed maps between the deloopings:

$$
\underset{\text{group homomorphisms}}{\Big(\phi \ : \ G \underset{\mathrm{hom}}{\longrightarrow} H\Big)} \xrightarrow{\ \sim\ } \underset{\text{maps of deloopings}}{\big(\mathbf{B}\phi \ : \ \mathbf{B}G \longrightarrow \mathbf{B}H\big)} \times \mathrm{Id}\big((\mathbf{B}\phi)(\mathrm{pt}), \ \mathrm{pt}\big) \quad \substack{\text{preserving the base point}}
$$
$$
g \ \longmapsto \ \phi(g) \qquad\qquad T \ \longmapsto \ H \otimes_G T \quad (h \otimes g) \mapsto h\phi(g)
$$

(177)

where the tensor product $H \otimes_G T$ is defined as the following quotient set (135) with the evident left action by H:

$$H \otimes_G T \ :\equiv \ \frac{H \times T}{(h\phi(g), t) \sim (h, g \circlearrowright t)}.$$

(178)

64

Free groups. The perspective on groups as loop group of pointed connected 1-types (147) is suggestive of a slick construction of the *free group* on a set of generators (e.g. [Jo90, §1]) as being simply the loop group (141) of the suspension (117) of the generating set together with a freely adjoined base element e (cf. [BBCDG21, §6.2]):

$$\text{Generators} : \text{Set} \quad \vdash \quad \text{FreeGrp}(\text{Generators}) \; :\equiv \; \underbrace{\Omega_{\text{nth}}}_{\text{looping}} \Big(\big[\underbrace{\text{S}(\{e\} \sqcup \text{Generators})}_{\text{suspension}} \big]_1 \Big) \; : \; \text{Grp} \tag{179}$$

Here each g : Generators may be identified with the loop obtained by concatenating (77) the corresponding meridian (117) with the reverse (76) of the merididan indexed by the base generator e:

$$\text{in the suspension} \quad \text{S(Generators)} \tag{180}$$

Hence in a free group (179) we may compose any finite sequence of generators by forming the concatenation (77) of the corresponding self-equivalences (180):

$$\overbrace{\text{Generators} \times \cdots \times \text{Generators}}^{n \text{ factors}} \longrightarrow \text{FreeGrp}(S)$$

$$(g_1, \cdots, g_n) \longmapsto \text{conc}\big(\text{mer}_{g_1}, \text{inv}(\text{mer}_e), \cdots, \text{mer}_{g_n}, \text{inv}(\text{mer}_e) \big).$$

For example, the free group on a single non-trivial generator, namely the loop type of the suspension of the pointed set with a single non-base element Generators $:\equiv \{1\}$, is the loop type of the circle (118), which is equivalently the group of integers [UFP13, §8.1] (these integers being the "winding numbers" of loops around the circle):

$$\text{FreeGrp}\big(\{e\} \sqcup \{1\}\big) \; \simeq \; \Omega\,[\text{S}(\text{Bit})]_1 \; \simeq \; \Omega S^1 \; \simeq \; \mathbb{Z}. \tag{181}$$

Presentations of groups. Further, the perspective on groups as pointed connected 1-types (147) allows for a slick construction of groups by *presentations* in terms of *generators and relations* (e.g. [MKS66][Jo90]): Given Generators : Set and a set Relations : Set of relations between the elements of the corresponding free group (179)

$$p_0, p_1 \; : \; \text{Relations} \to \text{FreeGrp}(\text{Generators})$$

(where we interpret r : Relations as stating that $p_0(r)$ should equal $p_1(r)$), then the loop group (141) of the 1-truncation $[-]_1$ (127) of the CW-complex (123) whose 1-cells are the Generators and whose 2-cell attachment (122) are the pairs of free group elements which are in Relations, is the group *presented* by these Generators and Relations:

$$\text{Gen} : \text{Set}, \; \text{Rel} : \text{Set}, \; p_1, p_2 : \text{Rel} \to \text{FreeGroup}(\text{Gen}) \quad \vdash \quad \mathbf{B}G(\text{Gen}, \text{Rel}) : \text{Grp}$$

$$\mathbf{B}G(\text{Gen}, \text{Rel}) \; :\equiv \; \left[\text{po} \left(\begin{array}{ccc} \text{Rel} \times S^1 & \xrightarrow{(r, \text{mer}_i) \mapsto \text{mer}_{p_i(r)}} & \text{S}\big(\{e\} \sqcup \text{Gen}\big) \\ \downarrow & & \\ \text{Rel} & & \end{array} \right) \right]_1 \tag{182}$$

The resulting group $G(\text{Gen}, \text{Rel})$ is this $\Omega_{\text{nth}}\mathbf{B}G(\text{Gen}, \text{Rel})$, and by the uniqueness of deloopings (Lem. 5.8), $\mathbf{B}G(\text{Gen}, \text{Rel})$ is equivalent to the type of left $G(\text{Gen}, \text{Rel})$-torsors.

Remark 5.13 (Comparison to the literature). A version of this construction (182) is briefly indicated in [UFP13, Ex. 8.7.17]. Alternatively, it is popular in homotopy type theory (following [LF14, Def. 3.1]) to write the intent of (182) as a single-step higher-inductive type whose generators include second-order-identifications (a "2-HIT") instead of an iterated pushout as above. But a general theory and semantics of n-HITs for $n \geq 2$ does not seem to be available in the literature — except via reformulation as iterated 1-HITs [vD18, §3.2], which is what we are using here.

Last but not least, it is in the form of (fundamental groups of loops in) CW-complexes (182) that group presentations are known in classical algebraic topology, see [Ro88, Thm. 7.34][MM10].

Algebra. Besides this basic group theory, we need some most basic concepts of algebra (e.g. [Kn06], cf. [1lab, §algebra]):

A group structure (140) is *abelian* (e.g. [Fu15]) if the group operation is *commutative* (then written additively as "+"), i.e. invariant under permutation of the arguments (cf. [BBCDG21, §4.12]).

$$
\underset{\substack{\text{abelian}\\\text{group}}}{\text{AbGrp}} \;\;:\equiv\;\; \left\{ \begin{array}{l} ((A, 0, +, -) : \text{Grp}) \;\left.\right\}\;\substack{\text{underlying}\\\text{group}} \\ \times\; (a_1, a_2 : A) \to \text{Id}_A(a_1 + a_2,\, a_2 + a_1) \Big\} \\ \qquad\qquad\qquad\qquad \underset{\text{abelian property}}{} \end{array} \right.
\tag{183}
$$

Given $G : \text{Grp}$ we may think — in view of Prop. 5.10 and the structure identity principle (146) — of functions from $\mathbf{B}G$ to this type (183) of abelian groups as *linear actions* of G (i.e., respecting the zero-element and the additive operation):

$$
G\text{LinAction} \;:\equiv\; (\mathbf{B}G \to \text{AbGrp})
\tag{184}
$$

A (commutative) *ring structure* (e.g. [AF92, §1][Bl11, Def. 1.1.1]) is an abelian group structure (183) equipped with a further product operation "distributing" over the abelian addition (cf. [Es19, §33.13] [1lab, §ring-theory], examples in §6):

$$
\underset{\substack{\text{unital}\\\text{ring}}}{\text{Ring}} \;:\equiv\; \left\{ \begin{array}{l}
R : \text{Set} \quad\}\;\text{data base} \\[4pt]
\text{(abelian group)} \left\{ \begin{array}{l}
\left\{ \begin{array}{l}
\times\; (0 : R) \\
\times\; (- : R \to R) \\
\times\; (+ : R \times R \to R)
\end{array} \right\}\;\text{structure} \\[10pt]
.\times\; (r : R) \to \text{Id}_R(0 + r,\, r) \\
\times\; (r : R) \to \text{Id}_R(-r + r,\, 0) \\
\times\; (r_1, r_2 : R) \to \text{Id}_R(r_1 + r_2,\, r_2 + r_1) \\
\times\; (r_1, r_2, r_3 : R) \to \text{Id}_R((r_1 + r_2) + r_3,\, r_1 + (r_2 + r_3))
\end{array} \right\}\;\text{properties} \\[30pt]
\text{(abelian monoid)} \left\{ \begin{array}{l}
\left\{ \begin{array}{l}
\times\; (1 : R) \\
\times\; (\cdot : R \times R \to R)
\end{array} \right\}\;\text{structure} \\[6pt]
\times\; (r : R) \to \text{Id}_R(1 \cdot r,\, r) \\
\times\; (r_1, r_2 : R) \to \text{Id}_R(r_1 \cdot r_2,\, r_2 \cdot r_1) \\
\times\; (r_1, r_2, r_3 : R) \to \text{Id}_R(r_1 \cdot (r_2 \cdot r_3),\, (r_1 \cdot r_2) \cdot r_3)
\end{array} \right\}\;\text{properties} \\[24pt]
\times\; (r, r_1, r_2 : R) \to \text{Id}_R(r \cdot (r_1 + r_2),\, r \cdot r_1 + r \cdot r_2) \;\left.\right\}\;\substack{\text{distributivity}\\\text{property}}
\end{array} \right.
\tag{185}
$$

Given $R : \text{Ring}$ (185), then *R-module structure* (e.g. [AF92, §2][Bl11, Def. 1.4.1]) is given by the following type:

$$
R : \text{Ring} \;\vdash\; \underset{R\text{-module}}{\text{Mod}_R} \;:\equiv\; \left\{ \begin{array}{l}
(N : \text{Set}) \qquad \}\;\text{data} \\[4pt]
\text{(abelian group)} \left\{ \begin{array}{l}
\left\{ \begin{array}{l}
\times\; (0 : N) \\
\times\; (- : N \to N) \\
\times\; (+ : N \times N \to N)
\end{array} \right\}\;\text{structure} \\[10pt]
\times\; (n : N) \to \text{Id}_N(0 + n,\, n) \\
\times\; (n : N) \to \text{Id}_N(-n + n,\, 0) \\
\times\; (n_1, n_2 : N) \to \text{Id}_N(n_1 + n_2,\, n_2 + n_1) \\
\times\; (n_1, n_2, n_3 : N) \to \text{Id}_N((n_1 + n_2) + n_3,\, n_1 + (n_2 + n_3))
\end{array} \right\}\;\text{properties} \\[30pt]
\text{(linear } R\text{-action)} \left\{ \begin{array}{l}
\times\; (\cdot : R \times N \to N) \qquad \}\;\text{structure} \\[6pt]
\times\; (n : N) \to \text{Id}_N(0 \cdot n,\, 0) \\
\times\; (n : N) \to \text{Id}_N(1 \cdot n,\, n) \\
\times\; (c_1, c_2 : R, n : N) \to \text{Id}_N(c_1 \cdot (c_2 \cdot n),\, (c_1 \cdot c_2) \cdot n) \\
\times\; (c : R, n_1, n_2 : N) \to \text{Id}_N(c \cdot (n_1 + n_2),\, c \cdot n_1 + c \cdot n_2) \\
\times\; (c_1, c_2 : R, n : N) \to \text{Id}_N((c_1 + c_2)n,\, c_1 \cdot n + c_2 \cdot n)
\end{array} \right\}\;\text{properties}
\end{array} \right.
\tag{186}
$$

Here we will be concerned with those R-modules that are R-lines (189):

Linear action of groups of units. Given a ring (185) we obtain:
- the **underlying abelian group** (183) whose group operation is the addition operation of the ring and which the remaining monoid structure makes into an R-module (186), which we will denote by the same symbols:

$$(-)_{\mathrm{udl}} : \quad \mathrm{Ring} \xrightarrow{\hspace{4cm}} R\mathrm{Mod} \xrightarrow{\hspace{4cm}} \mathrm{AbGrp}$$
$$\left(R, 0, -, +, 1, \cdot\right) \longmapsto \left(R, \left(0, -, +, \cdots\right), \left(\cdot, \cdots\right)\right) \longmapsto \left(R, 0, -, +\right), \tag{187}$$

- the **group of units**[19], namely of multiplicatively invertible elements in the ring (e.g. [Kn06, p. 143]) whose group operation is the multiplication operation of the ring:

$$(-)^{\times} : \quad \mathrm{Ring} \xrightarrow{\hspace{5cm}} \mathrm{Grp}$$
$$\left(R, 0, -, +, 1, \cdot\right) \longmapsto \left(\begin{matrix} (r, r^{-1} : R) \\ \times \mathrm{Id}_R(r \cdot r^{-1}, 1) \end{matrix}, (1,1), (\cdot, \cdot), (-)^{-1}\right). \tag{188}$$

- the **R-lines** over a ring R (e.g. [ABGHR14]), are those R-modules (186) which are isomorphic to $R_{\mathrm{udl}} : R\mathrm{Mod}$ (187) itself (also called the "free cyclic R-modules"):

$$R : \mathrm{Ring} \quad \vdash \quad R\mathrm{Line} :\equiv \left(\left\{N : R\mathrm{Mod} \,\middle|\, \exists (N \simeq R_{\mathrm{udl}})\right\}, \mathrm{pt} :\equiv R\right) \; : \; \mathrm{Type}^{*/}_{0 < \bullet \leq 1}. \tag{189}$$

Noticing the close analogy of the lines over a ring to torsors over a group, essentially the same proof as of Lem. 5.6 shows that the type of R-lines deloops the group of units (188). By Thm. 5.8, we therefore have a pointed equivalence:

$$R : \mathrm{Ring} \quad \vdash \quad R\mathrm{Line} \xrightarrow{\;\sim\;} R^{\times}\mathrm{Tors}_L$$
$$L \longmapsto \mathrm{Id}_{R\mathrm{Line}}(R, L). \tag{190}$$

For this reason, we are justified in defining $\mathbf{B}R^{\times} :\equiv R\mathrm{Line}$ to be the type of R-lines.

- the **action of the group of units on the underlying abelian group**: Via the ring multiplication (185), the group of units (188) acts (148) on the underlying abelian group (187) and the distributivity property makes this a linear action (184):

$$R : \mathrm{Ring} \quad \vdash \quad R^{\times} \mathbin{\reflectbox{$\circlearrowright$}} R_{\mathrm{udl}} :\equiv \left(R_{\mathrm{udl}}, \mathrm{act} :\equiv \cdot, \cdots\right) : R^{\times}\mathrm{LinAct}_L.$$

But under Prop. 5.10 and with (190) we may equivalently express this more elegantly under delooping, simply as:

$$R : \mathrm{Ring} \quad \vdash \quad R^{\times} \mathbin{\reflectbox{$\circlearrowright$}} R_{\mathrm{udl}} : \quad \mathbf{B}R^{\times} \xrightarrow{\hspace{3cm}} \mathrm{AbGrp}$$
$$\left(N, (0, +, -), \cdot\right) \longmapsto \left(N, (0, +, -)\right). \tag{191}$$

Iterated delooping and Eilenberg-MacLane types. In evident generalization of the notion of delooping of groups (Prop. 5.9) the *delooping of a pointed type* is (cf. [Wä, §2][BCTFR23]) another pointed *and connected* type whose loop type (88) recovers the given pointed type, up to specified equivalence (101):

$$\mathscr{A} : \mathrm{Type}, a : \mathscr{A} \quad \vdash \quad \overset{\text{\color{purple}\small delooping of }(\mathscr{A}, a)}{\mathrm{DeLpg}(\mathscr{A}, a)} :\equiv \begin{cases} \left(\mathbf{B}\mathscr{A} : \mathrm{Type}\right) & \text{\color{blue}\small higher data base} \\[4pt] \times \; (\mathrm{pt} : \mathbf{B}\mathscr{A}) \\[4pt] \times \; \left(\mathrm{equ} : \Omega_{\mathrm{pt}}\mathbf{B}\mathscr{A} \xrightarrow{\sim} \mathscr{A}\right) & \text{\color{blue}\small higher structure} \\[4pt] \times \; \left(\mathrm{cnd} : \exists! \left[\mathbf{B}\mathscr{A}\right]_0\right) \\[4pt] \times \; \left(\mathrm{ptd} : \mathrm{Id}\left(\mathrm{equ}(\mathrm{id}_{\mathrm{pt}}), a\right)\right) & \text{\color{blue}\small properties} \end{cases} \tag{192}$$

---

[19] A group of units need not be abelian if the corresponding ring is non-commutative. However, below in §6 we specialize to the ring of complex numbers, whose group of units is, of course, abelian.

Since the delooping of a pointed type has itself an underlying pointed type

$$\mathscr{A} : \text{Type}, a : \mathscr{A} \quad \vdash \quad \text{DeLpg}(\mathscr{A}, a) \longrightarrow (\mathbf{B}\mathscr{A} : \text{Type}) \times (\text{pt} : \mathbf{B}\mathscr{A})$$
$$(\mathbf{B}\mathscr{A}, \text{pt}, \text{equ}, \text{cnd}, \text{ptd}) \longmapsto (\mathbf{B}\mathscr{A}, \text{pt})$$

it makes sense to ask for iterated deloopings:

$$\mathscr{A} : \text{Type}, a : \mathscr{A}, n : \mathbb{N} \quad \vdash \quad \overset{\substack{n\text{-fold} \\ \textbf{delooping of } (\mathscr{A}, a)}}{n\text{DeLpg}(\mathscr{A}, a)} := \begin{cases} (\mathbf{B}^n \mathscr{A} : \text{Type}) \} \text{ higher data base} \\ \times (\text{pt} : \mathbf{B}^n \mathscr{A}) \\ \times (\text{equ} : \Omega^n_{\text{pt}} \mathbf{B}^n \mathscr{A} \overset{\sim}{\to} \mathscr{A}) \\ \times (\text{cnt} : \exists! [\mathbf{B}^n \mathscr{A}]_0) \\ \times (\text{ptd} : \text{Id}(\text{equ}(\text{id}_{\text{pt}}), a)) \end{cases} \begin{array}{l} \\ \} \text{ higher structure} \\ \\ \} \text{ properties} \end{array} \quad (193)$$

Here Ω^n_{pt} denotes the n-fold iteration of the looping operation (88) regarded as an endo-function on pointed types.

In the denotational semantics of homotopy theory one refers to *iterated loop space* structure:[20] [May72][Se73][Lu17, §5.2.6]. If a tower of n-fold deloopings are compatibly given for all values of $n : \mathbb{N}$ one speaks of a connective *spectrum* of spaces exhibiting *infinite loop space* structure [May77][Ad78][Lu17, §1.4] (type-theoretic discussion includes [Ca15, §3.2][vD18, §5.3]). This is the structure that leads over to the notion of *linear* homotopy types, which we turn to in the companion articles [SS23-QM][SS23-EoS]. For example:

Proposition 5.14 ([LF14, p. 3 & Thm. 5.4])**.** *The* $(n+1)$*-fold iterated delooping* (193) *of* A : AbGrp (183) *can be constructed as the* $n+1$*-truncation (Def. 5.1) of the* n*-fold iterated suspension* (117) *of the first delooping* (163)*:*

$$A : \text{AbGrp}, \ n : \mathbb{N} \quad \vdash \quad (\mathbf{B}^{n+1} A := [\Sigma^n \mathbf{B} A]_{n+1}, \text{nth}, \text{equ}) : n\text{DeLpg}(A, e) \quad (194)$$

In the denotational semantics of homotopy theory, these iterated deloopings of abelian groups are known as *Eilenberg-MacLane spaces* (Lit. 2.14) which are the classifying types/spaces for ordinary cohomology — see p. 69.

Twisted higher deloopings and associated bundles of EM-Types. A key point for the definition of twisted cohomology below (198) is that type-theoretic constructions such as the higher delooping in (194) apply in the generality where the data involved may *depend* on any given context: Specifically, placing a delooping type $\mathbf{B}G$ (163) into the context immediatley gives a notion of higher delooping types that is coherently acted on (174) by a given group G:

Definition 5.15 (*G*-Equivariant higher deloopings)**.** Given R : Ring (185) — considered with the action $R^\times \circlearrowright R_{\text{udl}}$ (191) of $R^\times$ (188) on its underlying abelian group R_{udl} (187) —, we obtain twisted higher deloopings of the underlying abelian groups of rings by applying (194) in the context of $\mathbf{B}R^\times$ (using Ntn. 5.11 on the right):

$$R : \text{Ring}, \ n : \mathbb{N} \quad \vdash \quad R^\times \circlearrowright \mathbf{B}^n R_{\text{udl}} \ : \ \mathbf{B}R^\times \longrightarrow \text{Type}$$
$$t \longmapsto \underset{(194) \quad (173)}{\mathbf{B}^n(\circlearrowright_t R_{\text{udl}})} \quad (195)$$

The semantics of this $\mathbf{B}R^\times$-dependent type (195) is given, as a special case of (174), by bundles of Eilenberg-MacLane spaces associated to the universal $R^\times$-principal bundle:

$\mathbf{B}R^\times$-**dependent higher** R-**delooping**	**Universal** $R^\times$-**Associated** $K(R_{\text{udl}}, n)$-**Bundle**
$t : \mathbf{B}R^\times \ \vdash \ \circlearrowright_t \mathbf{B}^n R_{\text{udl}} : \text{Type}$	

$$(196)$$

With these preliminaries in hand, we come to the main type construction of this section, in Def. 5.16 below.

---

[20]Here we present iterated (de/)looping as extra structure on $\mathbf{B}^n \mathscr{A}$; alternatively one could rewrite these definitions to regard it as extra structure on $\mathscr{A}$. The distinction, which traditional literature glosses over anyway, is a matter of convention rather than of practical content.

Twisted cohomology and Gauss-Manin connections. Given R : Ring (185) and $n : \mathbb{N}$ (111), the **ordinary cohomology** with coefficients R in degree n is (cf. [Ca15, §3. 2][Wä, §4.1], implemented in Agda: [BLM22]) the 0-truncation $[-]_0$ (127) of the function type (64) into the n-fold delooping $\mathbf{B}^n(-)$ (194) of the underlying abelian group (187).

$$\left.\begin{array}{c} \overset{\text{coefficients} \quad \text{degree}}{R : \text{Ring}, \ n : \mathbb{N},} \\ X : \text{Type} \\ \underset{\text{domain}}{} \end{array}\right\} \quad \vdash \quad \overset{\text{ordinary}}{\underset{\text{cohomology}}{H^n(X; R)}} :\equiv \left[X \to \mathbf{B}^n R_{\text{udl}}\right]_0 \ : \ \text{Type}$$

Under the dictionary of §5.1, this type construction clearly interprets as the traditional notion of ordinary cohomology (Lit. 2.15, cf. discussion and references in [FSS20-Cha, Ex. 1.0.2]).

Given furthermore a *twist* $\tau : X \to \mathbf{B}R^\times$ (188), we say (for the second version (198) cf. [vD18, Def. 5.4.2]) that the τ-**twisted ordinary cohomology** of X with coefficients in R is:

$$\left.\begin{array}{c} \overset{\text{coefficients} \quad \text{degree}}{R : \text{Ring}, \ n : \mathbb{N},} \\ \underset{\text{domain} \qquad \text{twist}}{X : \text{Type}, \ \tau : X \to \mathbf{B}R^\times} \end{array}\right\} \quad \vdash \quad \overset{\text{twisted cohomology}}{H^{n+\tau}(X; R)} :\equiv \left[(t : \mathbf{B}R^\times) \to \left(\overset{(87)}{\text{fib}_t(\tau)} \to \overset{(195)}{\mathbf{B}^n(\circlearrowright_t R_{\text{udl}})}\right)\right]_0 \ : \ \text{Type} \qquad (197)$$

$$\simeq \left[\underbrace{(t : \mathbf{B}R^\times) \times (x_t : \text{fib}_t(\tau))}_{(x:X) \quad (106)} \to \underbrace{\mathbf{B}^n(\circlearrowright_t R_{\text{udl}})}_{(\tau^* \mathbf{B}(R^\times \circlearrowright R_{\text{udl}}))(x)}\right]_0 \ : \ \text{Type} \qquad (198)$$

Under the dictionary of §5.1, this type construction interprets as the traditional notion of twisted ordinary cohomology (Lit. 2.15), see the discussion and references in [NSS12a, Rem. 4.22] and [SS20-Orb, Rem. 2.94][FSS20-Cha, Ex. 2.0.5]:

Twisted ordinary cohomology $H^{n+\tau}(X; R)$	
Homotopy type theory	**Homotopy theory**
$\left[(t : \mathbf{B}R^\times) \to \left(\text{fib}_t(\tau) \to \mathbf{B}^n(\circlearrowright_t R_{\text{udl}})\right)\right]_0$ (197)	$\left\{\begin{array}{c} \quad\quad\quad\quad \mathbf{B}^n R_{\text{udl}} /\!/ R^\times \\ \quad\quad\quad\quad\quad \downarrow \\ X \xrightarrow{\quad\tau\quad} \mathbf{B}R^\times \end{array}\right\}_{/\sim_{\text{hmt}}}$ (199)
$\left[(x : X) \to \left(\tau^* \mathbf{B}(R^\times \circlearrowright R_{\text{udl}})\right)(x)\right]_0$ (198)	$\left\{\begin{array}{c} \tau^*\left(\mathbf{B}^n R_{\text{udl}} /\!/ R^\times\right) \to \mathbf{B}^n R_{\text{udl}} /\!/ R^\times \\ \quad \downarrow \quad\quad\quad \downarrow \\ X = X \xrightarrow{\ \tau\ } \mathbf{B}R^\times \end{array}\right\}_{/\sim_{\text{hmt}}}$

For the construction of Gauss-Manin connections from §4.2, we are to furthermore consider B-indexed families of such twisted cohomology types (199). Remarkably, in the syntax of homotopy types this is a triviality that amounts to introducing a type B into the context and having the terms X and τ depend on it:

Definition 5.16 (Gauss-Manin transport on fibrations of twisted cohomology groups)**.** We say that the type of *fibrations of twisted ordinary cohomology sets* is:

$$\left.\begin{array}{c} \overset{\text{coefficients} \quad \text{degree} \quad \text{parameter base}}{R : \text{Ring}, \ n : \mathbb{N}, \ B : \text{Type},} \\ \underset{\text{fibration of domains} \qquad\qquad \text{family of twists}}{X_{(-)} : B \to \text{Type}, \ \tau_{(-)} : (b : B) \to (X_b \to \mathbf{B}R^\times)} \end{array}\right\} \quad \vdash$$

$$\overset{\text{twisted cohomology}}{H^{n+\tau_{(-)}}(X_{(-)}; R)} :\equiv$$

$$\left[(t : \mathbf{B}R^\times) \to \left(\underbrace{\text{fib}_t(\tau_{(-)})}_{\text{fib}_{(-,t)}(\text{pr}_X, \tau) \ (204)} \to \mathbf{B}^n(\circlearrowright_t R_{\text{udl}})\right)\right]_0 \ : \ \overset{\text{fibered}}{\underset{}{B \to \text{Type}}} \qquad (200)$$

Given such, its *Gauss-Manin monodromy* is the corresponding transport (74) over the base type B:

$$\text{GMTransport} \ : \ \prod_{b_1, b_2 : B} \left(\text{Id}_B(b_1, b_2) \longrightarrow \left(H^{n+\tau_{b_1}}(X_{b_1}; R) \xrightarrow{\sim} H^{n+\tau_{b_2}}(X_{b_2}; R)\right)\right).$$

$$\left(b_1 \overset{p_{12}}{\rightsquigarrow} b_2\right) \longmapsto (p_{12})_* \qquad (201)$$

By Thm. 4.13 and under the dictionary in §5.1, this is the type-theoretic construction whose denotational semantics is the parallel transport/monodromy of Gauss-Manin connections on fibrations of twisted ordinary cohomology sets.

(These cohomology sets inherit abelian group structure from that of the coefficients R_{udl}, making them into *cohomology groups*, as expected, in fact they become modules over the un-twisted cohomology ring; and the Gauss-Manin mondromy (201) is by *linear maps* with respect to this structure. In applications, such as to quantum computation below, it is crucial to add certificates of this linear structure to the data type of the Gauss-Manin connection, but here we do not further dwell on this point.)

Next, in §6, we turn to the specialization of this general construction to the case corresponding to Knizhnik-Zamolodchikov connections and hence to the operation of topological quantum gates (Thm. 6.8 below).

In closing this subsection, we comment on the equivalence under the brace in (200):

Remark 5.17 (Equivalence between fiberwise and global twisting). Given a family of domain types equipped with a fiberwise system of twists as assumed in (200),

$$X_{(-)} : B \longrightarrow \text{Type}, \qquad \tau_{(-)} : B \longrightarrow \left(X_b \to \mathbf{B}R^{\times}\right)$$

consider the corresponding total space (106)

$$
X :\equiv (b : B) \times X_b, \qquad
\begin{array}{rccccc}
\text{pr}_X : & X & \longrightarrow & (b : X) \times X_b & \longrightarrow & B \\
& x & \longmapsto & (b, x_b) & \longmapsto & b
\end{array}
\tag{202}
$$

equipped with the corresponding "global twist":

$$
\begin{array}{rccccc}
\tau : & X & = & (b : B) \times X_b & \longrightarrow & \mathbf{B}R^{\times} \\
& x & \longmapsto & (b, x_b) & \longmapsto & \tau_b(x_b)
\end{array}
\tag{203}
$$

Then the fiber type (87) of the pairing

$$\left(\text{pr}_X, \tau\right) : X \longrightarrow B \times \mathbf{B}R^{\times}$$

at any $(b, t) : B \times \mathbf{B}R^{\times}$ is equivalently the fiber type of τ_b at t:

$$
\begin{array}{rll}
\text{fib}_{(b,t)}\left(\text{pr}_X, \tau\right) & = (x : X) \times \text{Id}\left(\left(\text{pr}_X(x), \tau(x)\right), (b, t)\right) & \text{by (87)} \\[4pt]
& \simeq (x : X) \times \text{Id}\left(\text{pr}_X(x), b\right) \times \text{Id}(\tau(x), t) & \text{by (144)} \\[4pt]
& \simeq (x_b : X_b) \times \text{Id}\left(\tau_b(x_b), t\right) & \text{by (87) \& (203)} \\[4pt]
& = \text{fib}_t\left(\tau_b\right) & \text{by (87).}
\end{array}
\tag{204}
$$

6 The homotopy type of anyon braid gates

Finally, we discuss the specialization of the type construction of Gauss-Manin monodromy in Def. 5.16 to that of KZ-monodromy on $\widehat{\mathfrak{su}_2}^{\kappa-2}$-conformal blocks, along the lines of Ex. 4.14, concluding with Def. 6.7 and Thm. 6.8 below.

This requires encoding the three specializations (54) - (56) invoked in Ex. 4.14:

(i) The domain fibration is to be specialized to that of delooped **braid groups** $\mathbf{B}\,\mathrm{PBr}(n+N) \to \mathbf{B}\,\mathrm{PBr}(N)$;

(ii) the local coefficients are specialized to the EM-type $\mathbb{C}^\times \wr K(\mathbb{C}, n)$ of the **complex numbers**;

(iii) the twist is specialized to what on pure Artin generators is a list of **complex exponentials** $\exp(2\pi\mathrm{i}-) : \mathbb{Q} \to \mathbb{C}^\times$.

All three of these are, of course, classical mathematical constructions, and their encodings in typed programming languages are in principle well-understood. Nonetheless, at the time of this writing, implementations specifically for a univalently computing and inductively homotopy-typed programming languages as needed here (such as cubical `Agda`, Lit. 2.28) is not readily available for import from standard libraries. Therefore we dwell a little on how to get one's hands on these data structures.

The fibration of delooped pure braid groups. The construction of the delooping type (cf. p. 59) $\mathbf{B}\,\mathrm{PBr}(N+1)$ of the pure braid group (7) is a straightforward consequence of our previous discussion: Given that the pure braid group has a finite presentation (15) by pure braid generators (13), we may use the construction formula (182) (or, in `Agda`, the 2-HIT equivalent to it): First form the suspension type of the set of pure braid generators (13) and then pushout-out (122) each loop formed by a pair of free group elements in relation:

$$\mathbf{B}\mathrm{PBr}(N+1) :\equiv \left[\mathrm{po} \left(\begin{array}{c} \overset{(16)\,(17)}{(\text{Artin-Lee relations})} \times S^1 \longrightarrow \mathrm{S}\left(\{\mathrm{e}\} \sqcup \left\{ b_{ij} = \left[\!\!\left| \cdots \left|\,\right|\,\underset{i}{\Big|}\,\genfrac{}{}{0pt}{}{}{}\,\underset{j}{\Big|}\,\right|\,\right| \cdots \left|\,\right| \right]\!\!\right\}_{1 \le i < j \le N+1}\right) \\ \downarrow \\ (\text{Artin-Lee relations}) \end{array} \right) \right]_1$$

By the classical homotopy equivalence (5) we may regard this as our type-theoretic model of the homotopy type of ordered configuration spaces: In *cohesive homotopy type theory* (which we do not discuss here, but see the Outlook on p. 4) we would (or will) have a type equivalence of the form $\int \underset{\{1,\cdots,N+1\}}{\mathrm{Conf}}(\mathbb{R}^2) \simeq \mathbf{B}\mathrm{PBr}(N+1)$ where "$\int$" denotes the cohesive *shape modality*.

Now observe, by the nature of the pure braid generators b_{ij} (15), that the canonical fibrations of ordered configuration spaces (6) which forget the last point(s) in a configuration clearly induce on pure braid groups the homomorphisms which trivialize those generators that carry the label of a discarded strand as an index:

$$
\begin{array}{ccccccc}
\underset{\{1,\cdots,N+2\}}{\mathrm{Conf}}(\mathbb{R}^2) & \underset{\mathrm{whe}}{\simeq} & B\mathrm{PBr}(N+2) & b_{I,J} & b_{I,N+1} & b_{I,N+2} & \\
\downarrow & & \downarrow & \updownarrow & \updownarrow & \updownarrow & \\
\underset{\{1,\cdots,N+1\}}{\mathrm{Conf}}(\mathbb{R}^2) & \underset{\mathrm{whe}}{\simeq} & B\mathrm{PBr}(N+1) & b_{I,J} & b_{I,N+1} & \mathrm{e} & \text{for } 1 \le I < J \le N \quad (205) \\
\downarrow & & \downarrow & \updownarrow & \updownarrow & \updownarrow & \\
\underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{R}^2) & \underset{\mathrm{whe}}{\simeq} & B\mathrm{PBr}(N) & b_{I,J} & \mathrm{e} & \mathrm{e} &
\end{array}
$$

The delooping (177) of this group homomorphism yields the desired fibration:

$$
\mathrm{pr}: \quad \mathbf{B}\mathrm{PBr}(N+n) \longrightarrow \mathbf{B}\mathrm{PBr}(N)
$$

$$
\begin{array}{ccc}
\underset{b_{IJ}}{\{\mathrm{pt}\}} & \mapsto & \underset{b_{IJ}}{\{\mathrm{pt}\}} \\
\underset{b_{Ii}}{\{\mathrm{pt}\}} & \mapsto & \underset{\mathrm{e}}{\{\mathrm{pt}\}} \qquad \text{for } \begin{array}{l} 1 \le I < J \le N \\ N < i < j \le N+n \end{array} \quad (206) \\
\underset{b_{ij}}{\{\mathrm{pt}\}} & \mapsto & \underset{\mathrm{e}}{\{\mathrm{pt}\}}
\end{array}
$$

Constructing the continuum. For encoding physical reality in general (and in particular for encoding topological quantum gates, via Def. 6.7 below) we need the data structure of *real numbers* ("*the continuum*" $\mathbb{R}$, e.g. [Ru64]). Elementary as this appears to any practicing mathematician today, when reconsidered from the bare logical foundations of typed programming languages one is reminded that there is a fair bit of work and some subtleties involved in constructing the real numbers and verifying their expected properties starting from just the type of natural numbers. This invokes non-trivial insights fully developed only in the 19th century (cf. [Co15]), fully understood in its "constructive" refinement only late in the 20th century ([Bish67][BB85][Br99]), which only now in the 21st century is being appreciated as the theory of real numbers pertinent to certified computing (e.g. [O'C07][GNSW07], cf. Lit. 2.29).

This may serve to explain that an actual library for real number arithmetic in the programming language Agda (Lit. 2.26) has been started only recently [Mu22] (see also [Lu15]) and is still under development. However, its basic principles of constructions and proofs are those originally developed already, in full detail, for the *constructive analysis* of [Bish67][BB85] and are well-understood. Better yet, implemented in homotopy type theory this seminal historical program of constructive analysis arguably finds its conceptual conclusion, in that here the required quotient sets (135) actually exist, thus solving the remaining problems (cf. [Li14, §1.1][Mu22, §4.3.2]) with previous "setoid" models. (A yet better but currently more hypothetical approach may be that of [UFP13, §11], see Rem. 6.3 below).

We now briefly list the incremental ring data structure constructions that produce the type of complex numbers, in this fashion, starting from the type of natural numbers (111) and proceeding through the types of integer numbers (207), rational numbers (211) and – which requires the most care: — the type of real numbers (214).

Remark 6.1 (Conventions for displaying ring data structures)**.** In doing so in the following, beware that:
 (i) Throughout we omit displaying the construction of the properties-certificates (such as for associativity etc.): These are all either straightforward or, in the case of the convergence-certificates for the operations on real numbers (214), the required ideas are all given in [Bish67][BB85] and have been coded into Agda in [Lu15][Mu22].
 (ii) Consequently, we display functions on quotient types X/R (135) as functions on the quotient*ed* types X, leaving implicit the proof certificates that these functions respect the pertinent equivalence relations and hence descend to the quotient, much as spelled out in the proof of Prop. 5.10.
 (iii) We systematically "overload" notation for operations on number systems, as usual: For instance in the definition of the addition operation on integer numbers (207)

$$(n_1, m_1) + (n_2, m_2) :\equiv (n_1 + n_2, m_1 + m_2)$$

it is understood that the operation on the left goes $+ : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$ (as per the previous item) and is defined by the operation on the right which instead goes $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ and was constructed previously (112).
 (iv) On our use of quotient types of real and complex numbers in the following, see Rem. 6.3 below.

The integer numbers. The archetypical (namely: initial) example of ring data structure (185) is the type of *integers*: Its underlying set may be given (cf. [UFP13, Rem. 6.10.7][1lab, §Data.Int]) as the quotient type (135) of that of pairs (62) of natural numbers (111) by the equivalence relation (134) which identifies those pairs of pairs (n, m) whose cross-sums (112) are equal, hence which serve as stand-ins for their difference $n - m$:

$$
\begin{aligned}
\vdash \quad & \left(\mathbb{Z}, 0, +, -, 1, \cdot, \right) : \text{Ring} \\
& \text{where} \\
& \mathbb{Z} :\equiv \mathbb{N} \times \mathbb{N} \,\big/\, \left((n_1, m_1), (n_2, m_2) : \mathbb{N} \times \mathbb{N}\right) \times \text{Id}_{\mathbb{N}}\left(n_1 + m_2, n_2 + m_1\right) \\
& 0 :\equiv (0,0) \\
& + \;:\; \left((n_1, m_1), (n_2, m_2)\right) \;\mapsto\; \left(n_1 + n_2, m_1 + m_2\right) \\
& - \;:\; (n, m) \;\mapsto\; (m, n) \\
& 1 :\equiv \left(\text{succ}(0), 0\right) \\
& \cdot \;:\; \left((n_1, m_1), (n_1, m_2)\right) \;\mapsto\; \left(n_1 \cdot n_2 + m_1 \cdot m_2, \; m_1 \cdot n_2 + n_1 \cdot m_2\right)
\end{aligned}
\tag{207}
$$

Using the canonical inclusion of the underlying types[21] of natural numbers (111)

$$
\begin{aligned}
\iota \;:\; \mathbb{N} &\longrightarrow \mathbb{Z} \\
n &\longmapsto (n, 0)
\end{aligned}
\tag{208}
$$

---

[21] The inclusion (208) is of course a homomorphism of monoid (semi-group) data structures, but here we do not dwell on monoid structure, for brevity.

we obtain the *ordering relation* on the integer numbers

$$\leq \;:\quad \mathbb{Z} \times \mathbb{Z} \xrightarrow{\hspace{3cm}} \text{Prop}$$
$$(n_1, n_2) \;\longmapsto\; (k : \mathbb{N}) \times \text{Id}(n_2, n_1 + \iota(k)) \tag{209}$$

and hence the sub-type of *positive* integers (equivalent to that of poisitive natural numbers)

$$\mathbb{Z}_+ \;:\equiv\; (n : \mathbb{Z}) \times (1 \leq n). \tag{210}$$

The rational numbers. The ring data structure (185) of *rational numbers* may be given by the quotient set (134) of pairs consisting of a *numerator* $p : \mathbb{Z}$ (207) and a *denominator* $q : \mathbb{Z}_+$ (210) subject to the usual identification of fractional arithmetic which makes the pair (p, q) be a stand-in for the fraction p/q:

$$\vdash \quad (\mathbb{Q}, 0, +, -, 1, \cdot) \;:\; \text{Ring}$$
$$\text{where}$$
$$\mathbb{Q} :\equiv \mathbb{Z} \times \mathbb{Z}_+ \,\big/\, \big((p_1, q_1), (p_1, q_2) : \mathbb{Z} \times \mathbb{Z}_+\big) \times \text{Id}_{\mathbb{Z}}\big(p_1 \cdot q_2,\, q_1 \cdot p_2\big)$$
$$0 :\equiv (0, 1)$$
$$+ \;:\; \big((p_1, q_1), (p_2, q_2)\big) \;\mapsto\; \big(p_1 \cdot q_2 + p_2 \cdot q_1,\; q_1 \cdot q_2\big) \tag{211}$$
$$- \;:\; (p, q) \mapsto (-p, q)$$
$$1 :\equiv (1, 1)$$
$$\cdot \;:\; \big((p_1, q_1), (p, q_2)\big) \;\mapsto\; \big(p_1 \cdot p_2,\; q_1 \cdot q_2\big)$$

The ordering relation (209) on the integer numbers induces the ordering on the rational numbers (remembering that we are forcing the denominators q to be positive):

$$\leq \;:\quad \mathbb{Q} \times \mathbb{Q} \xrightarrow{\hspace{3cm}} \text{Prop}$$
$$\big((q_1, p_1), (q_2, p_2)\big) \;\longmapsto\; q_1 \cdot p_2 \leq q_2 \cdot p_1. \tag{212}$$

For the following construction of the type of real numbers below in (214) we introduce common notational abbreviations for the multiplicative *inverse* of a positive number and for the *square* of any rational number

$$\frac{1}{(-)} \;:\; \mathbb{Z}_+ \longrightarrow \mathbb{Q} \qquad\qquad (-)^2 \;:\; \mathbb{Q} \longrightarrow \mathbb{Q}$$
$$n \;\longmapsto\; (1, n) \qquad\qquad r \;\longmapsto\; r \cdot r \tag{213}$$

Notice that bounds (212) on a square r^2 (213) equivalently serve as bounds on the *absolute value* $|r|$ (which we do not introduce separately).

The real numbers. The following is the construction of the ring data structure (185) of *real numbers* as *regular sequences* $x_{(-)} : \mathbb{Z}_+ \to \mathbb{Q}$ of rational numbers (211) indexed by positive integers (210), serving as stand-ins for the real number to which these converge. This definition and the verification of its intended properties is due to [Bish67, pp. 15][BB85, pp. 18] and has been implemented in Agda by [Lu15][Mu22, Def. 3.3.1] (also in Coq [KS13] following a monadic re-formulation due to [O'C07]) — there as data sets equipped with equivalence relations (134), while we now display the corresponding quotient type (135), bewaring of Rem. 6.3 below:

$$\vdash \quad (\mathbb{R}, 0, +, -, 1, \cdot) \;:\; \text{Ring}$$
$$\text{where}$$

$$\mathbb{R} :\equiv \underbrace{\big(x_{(-)} : \mathbb{Z}_+ \to \mathbb{Q}\big)}_{\substack{\text{sequences of} \\ \text{rational numbers (211)}}} \times \underbrace{\Big((n, m : \mathbb{Z}_+) \to \big((x_n - x_m)^2 \leq (\tfrac{1}{n} + \tfrac{1}{m})^2\big)\Big)}_{\text{which converge regularly}} \Big/ \underbrace{\big(x_{(-)}, y_{(-)}\big)}_{\text{modulo}} \underbrace{\vphantom{\big(}}_{\text{those}} \times \underbrace{\Big((n : \mathbb{Z}_+) \to \big((x_n - y_b)^2 \leq (\tfrac{2}{n})^2\big)\Big)}_{\text{that converge to zero}}$$

$$0 \;:\; n \mapsto 0$$
$$+ \;:\; \big(x_{(-)}, y_{(-)}\big) \;\mapsto\; \big(n \mapsto x_n + y_n\big)$$
$$- \;:\; x_{(-)} \mapsto \big(n \mapsto -x_n\big)$$
$$1 \;:\; n \mapsto 1$$
$$\cdot \;:\; \big(x_{(-)}, y_{(-)}\big) \;\mapsto\; \big(n \mapsto x_n \cdot y_n\big)$$

$$\tag{214}$$

Remark 6.2 (**Computational content of real numbers**). In the type (214) any real number is *represented* by a sequence of rational numbers; in practice these representing sequences

$$\frac{\vdash \quad x_{(-)} : \mathbb{Z}_+ \to \mathbb{Q}}{n : \mathbb{Z}_+ \quad \vdash \quad x_n : \mathbb{Q}} \tag{215}$$

are little programs (60) which for any prescribed but finite precision $\sim \frac{1}{n}$ compute the intended number to within that precision. That this is really what it means to *know* a real number was the conviction of the founding fathers of *constructive analysis* [Bish67][BB85]; in any case this is what it means for a finite computing machine to store an exact real number (as opposed to some finite floating-point approximation) in its memory (see [Vui88, §3]).

For example, the circle number π may be encoded by the BBP-formula [BBP97], for which the required proof certificates of convergence are evident:

$$\pi :\equiv \left(n \mapsto \sum_{k=0}^{n} \frac{1}{16^k} \left(\frac{4}{6k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right) \right) \ : \ \mathbb{R} \tag{216}$$

Remark 6.3 (**Cauchy completeness and/or its constructive failure for Bishop reals**). A desirable or even constituting property of the real numbers is their *Cauchy completeness* (e.g. [Ru64, Def. 3.12]), meaning that any Cauchy sequence of real numbers converges to a real number: Intuitively this is the requirement that real numbers really form a *continuum* in that no "gaps" remain on the real line. But there is a difference in whether one asks for convergence of *constructible* sequences of real numbers, or more generally:

On the one hand, given a Cauchy sequence of *operational encodings* of real numbers (215), namely of, in turn, Cauchy sequences of rational numbers, hence given a Cauchy sequence of Cauchy sequences

$$x^{(-)}_{(-)} : \mathbb{Z}_+ \longrightarrow \left(\mathbb{Z}_+ \to \mathbb{Q} \right), \tag{217}$$

then its convergence is indeed provable ([Bish67, p. 27, Thm. 2][BB85, p. 29, Thm. 3.3], cf. [Mu22, Thm. 3.4.4]): this is the fully constructive notion of Cauchy completeness of the real numbers.

On the other hand, such Cauchy sequences of Cauchy sequences of rational numbers (217) represent single sequences of real numbers

$$x^{(-)} : \mathbb{Z}_+ \longrightarrow \mathbb{R} \tag{218}$$

when we regard the latter inside the quotient type (214). But given *just* such quotiented Cauchy sequence data of type (218), there is no proof that it converges to an element of $\mathbb{R}$ [Lu07]. By the constructive proof of [Bish67, p. 27][BB85, p. 29] it *would* be possible if one could *lift* the sequence (218) back to one of the "constructive" form (217) — but this is in turn not possible in general. (The relevant "axiom of countable choice" does not hold syntactically in the type theory. While it does hold semantically in the classical model topos (107), it does not hold in the more general model toposes that we are eventually interested in, as per p. 4.)

Luckily, it looks like for our intended application (in Def. 6.7 below) we never (need to) care about sequences of real number data (218) for which we do not have constructive lifts (217), whence the operational Cauchy completeness proven by [Bish67, p. 27][BB85, p. 29][Mu22, Thm. 3.4.4] ensures that real number data of type (214) is satisfactory in applications.

Remark 6.4 (Other descriptions of the reals). There are other possible constructions of the real numbers which do not share the Cauchy completeness deficiency of the Bishop real numbers.
(i) The most straightforward is the type of Dedekind real numbers, or two-sided Dedekind cuts ([UFP13, Defn. 11.2.1]):

$$\mathbb{R}_D :\equiv \left\{ \begin{array}{l} (L, U : \mathbb{Q} \to \mathrm{Prop}) \ \} \ \text{\small structure} \\[4pt] \times \ \Big(\exists\{q : \mathbb{Q} \mid L(q)\} \times \exists\{r : \mathbb{Q} \mid U(r)\} \Big) \\[4pt] \times \ \Big((q : \mathbb{Q}) \to L(q) \simeq \exists\{r : \mathbb{Q} \mid (q < r) \times L(r)\} \Big) \\[4pt] \times \ \Big((r : \mathbb{Q}) \to R(r) \simeq \exists\{q : \mathbb{Q} \mid (q < r) \times U(q)\} \Big) \ \} \ \text{\small properties} \\[4pt] \times \ \Big((q : \mathbb{Q}) \to (L(q) \times U(q)) \to \varnothing \Big) \\[4pt] \times \ \Big((q, r : \mathbb{Q}) \to \exists(L(q) + R(r)) \Big) \end{array} \right. \tag{219}$$

The Dedekind real numbers have the benefit of being provably Cauchy complete without any non-constructive principles. Since they evidently contain the rationals by sending $x : \mathbb{Q}$ to the cut $(q \mapsto (q < x), r \mapsto (x < r))$, this implies that the Bishop reals admit a map to the Dedekind reals which respects the quotient relation on the regular sequences.

(ii) Finally, another construction principle for the type of Cauchy real numbers as a more intricate "higher inductive-inductive type" (further generalizing the notion of higher inductive types, pp. 49) has been laid out in [UFP13, §11.3] and proven to be genuinely Cauchy complete ([UFP13, Thm. 11.3.49]). This novel construction principle has attracted some type-theoretic attention and may well be the way to go forward in the future; but at the time of this writing, it is not available in practice.

The complex numbers. With a type (214) of real numbers in hand, it is straightforward to construct the corresponding ring data structure (185) of complex numbers (applying the classical formulas; see e.g. [Ru64, Def. 1.24]):

$$\vdash \quad (\mathbb{C}, +, -, 0, \cdot, 1) : \text{Ring}$$

where

$$
\begin{aligned}
(z : \mathbb{C}) &:\equiv \big(\text{Re}(z), \text{Im}(z)\big) : \mathbb{R} \times \mathbb{R} \\
0 \quad &:\equiv (0, 0) \\
+ \quad &: \Big(\big(\text{Re}(z_1), \text{Im}(z_1)\big), \big(\text{Re}(z_2), \text{Im}(z_2)\big)\Big) \mapsto \big(\text{Re}(z_1) + \text{Re}(z_2), \text{Re}(z_1) + \text{Re}(z_2)\big) \\
- \quad &: \big(\text{Re}(z), \text{Im}(z)\big) \mapsto \big(-\text{Re}(z), -\text{Im}(z)\big) \\
\cdot \quad &: \Big(\big(\text{Re}(z_1), \text{Im}(z_1)\big), \big(\text{Re}(z_2), \text{Im}(z_2)\big)\Big) \mapsto \Big(\text{Re}(z_1) \cdot \text{Re}(z_2) - \text{Im}(z_1) \cdot \text{Im}(z_2), \ \text{Re}(z_1) \cdot \text{Im}(z_2) + \text{Im}(z_1) \cdot \text{Re}(z_2)\Big) \\
1 \quad &:\equiv (1, 0)
\end{aligned}
$$

(220)

So the imaginary unit is

$$\text{i} :\equiv (0, 1) : \mathbb{C}.$$

(221)

We now obtain the complex exponential function on rational arguments by composing its classical series expansion $n \mapsto \sum_{k=0}^{n} \frac{1}{k!}(-)^k$ (e.g. [Ru64, p. 178]) with the series representation of π (216); and by classical computations, the result is readily certified to be a group homomorphism (176) from the underlying abelian group (187) of the ring of rational numbers (211) to the group of units (188) of the ring of complex numbers (220):

$$\exp\big(2\pi\text{i} \cdot (-)\big) \ : \ \mathbb{Q}_{\text{udl}} \xrightarrow[\text{hom}]{} \mathbb{C}^{\times}.$$

(222)

Similarly, we readily equip $\mathbb{C}^{\times}$ with a certificate that it is abelian, $\mathbb{C}^{\times} : \text{AbGrp}$ (183).

Lemma 6.5 (Assigning phases to pure Artin generators). *Any list of rational numbers, one for each pure braid generator, defines a group homomorphism:*

$$
N : \mathbb{N}_+, \ n : \mathbb{N} \quad \vdash \quad \left(r_{(--)} : \begin{array}{c} (k_1, k_2 : \mathbb{Z}_+) \\ \times (k_1 + 1 \leq k_2) \\ \times (k_2 \leq N + n) \end{array} \to \mathbb{Q} \right) \longrightarrow \Big(\mathbf{B}\,\text{PBr}(N+n) \longrightarrow \mathbf{B}\,\mathbb{C}^{\times} \Big)
$$

(223)

$$
r_{(--)} \quad \longmapsto \quad \Big(\ \underset{\substack{b_{k_1 k_2} \\ \textbf{pure braid} \\ \textbf{generators}}}{\text{pt}} \ \mapsto \ \underset{\exp\left(2\pi\text{i}r_{(k_1 k_2)}\right)}{\text{pt}} \ \Big)
$$

Proof. Since the relations (17) on the pure Artin generators (13) are all group commutator relations, and since in the abelian target group every group commutator is canonically witnessed as an identity. $\square$

Remark 6.6 (Efficiency of implementations). There is much room to replace this general construction with optimized special-purpose constructions in special cases. For example, if in applications we are to focus on rational numbers with a numerator equal to $q = 4$ (which, under the next Thm. 6.8, is the case of Majorana anyons!, Lit. 2.19), then the corresponding exponential is an integer complex number and may be defined directly:

$$\exp\big(2\pi\text{i}\,p/4\big) \ = \ \text{i}^p.$$

The homotopy data structure of topological quantum gates. With all these data structures in hand, we may conclude.

Definition 6.7 (Homotopy data structure of conformal blocks)**.** In specialization of Def. 5.16, we obtain this type:

$$\left.\begin{array}{l} \color{blue}{\text{punctures}}\ \ \color{blue}{\text{degree}}\ \ \ \color{blue}{\text{shifted level}} \\ N : \mathbb{N}_+, \ \ n : \mathbb{N}, \ \ \kappa : \mathbb{N}_{\geq 2} \\[4pt] \mathrm{w}_{(-)} : N \to \{0, \cdots, \kappa-2\} \\ \color{blue}{\text{weights}} \end{array}\right\} \vdash \left(\vec{z} \mapsto \left[(t : \mathbf{B}\mathbb{C}^\times) \to \left(\mathrm{fib}_{(t,\vec{z})}\big(\mathrm{pr}_N^{N+n}, \tau_{(\kappa,\mathrm{w}_\bullet)}\big) \to \mathbf{B}^n(\zeta_t\,\mathbb{C}_{\mathrm{udl}}) \right) \right]_0 \right) : \mathbf{B}\,\mathrm{PBr}(N) \to \mathrm{Type}$$

$$(224)$$

where

$$(206)\quad \mathrm{pr}_N^{N+n} : \quad \mathbf{B}\,\mathrm{PBr}(N+n) \longrightarrow \mathbf{B}\,\mathrm{PBr}(N)$$

$$\tau_{(\kappa,\mathrm{w}_\bullet)} : \quad \mathbf{B}\,\mathrm{PBr}(N+n) \longrightarrow \mathbf{B}\,\mathbb{C}^\times$$

$$(53)\,(223)$$

Theorem 6.8 (**Topological quantum gates as homotopy data structure**)**.** *The semantics in the classical model topos* (107) *of the transport operation* (201) *in this type* (224) *is given by the monodromy of the Knizhnik-Zamolodchikov connection, on* $\widehat{\mathfrak{su}}_2^{\kappa-2}$-*conformal blocks (on the Riemann sphere with $N+1$ punctures weighted by $(\mathrm{w}_I)_{I=1}^N$ and $\mathrm{w}_{N+1} = n + \sum_I \mathrm{w}_I$).*

Proof. By Example 4.14 of Theorem 4.13, we are reduced to showing that the semantics of the type formation (224) equals the topological construction expressed by the formula (58). This follows by applying the syntax/semantics dictionary §5.1 iteratively to the sub-terms of (224), as shown in the following steps:

Syntax	$\xleftrightarrow{\S 5.1}$	Semantics
$\vec{z} : \mathbf{B}\,\mathrm{PBr}(N),\ t : \mathbf{B}\mathbb{C}^\times\ \vdash$ $\quad \mathrm{fib}_{(t,\vec{z})}(\mathrm{pr}_N^{N+n}, \tau_{(\kappa,\mathrm{w}_\bullet)}) : \mathrm{Type}$	(106) (5)	$\underset{\{1,\cdots,N+n\}}{\mathrm{Conf}}(\mathbb{C})$ $\big\downarrow {\scriptstyle (\mathrm{pr}_N^{N+n}, \tau_{(\kappa,\mathrm{w}_\bullet)})}$ $\underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C}) \times B\mathbb{C}^\times$
$\vec{z} : \mathbf{B}\,\mathrm{PBr}(N),\ t : \mathbf{B}\mathbb{C}^\times\ \vdash$ $\quad \mathbf{B}^n(\zeta_t\,\mathbb{C}_{\mathrm{udl}}) : \mathrm{Type}$	(67) (175)	$p^*_{\underset{\{1,\cdots,N\}}{\mathrm{Conf}}}\left(\begin{array}{c} K(\mathbb{C},n) \times_{\mathbb{C}^\times} E\mathbb{C}^\times \\ \big\downarrow \\ B\mathbb{C}^\times \end{array} \right)$
$\vec{z} : \mathbf{B}\,\mathrm{PBr}(N),\ t : \mathbf{B}\mathbb{C}^\times\ \vdash$ $\quad \mathrm{fib}_{(t,\vec{z})}(\mathrm{pr}_N^{N+n}, \tau_{(\kappa,\mathrm{w}_\bullet)}) \to \mathbf{B}^n(\zeta_t\,\mathbb{C}_{\mathrm{udl}}) : \mathrm{Type}$	(66)	$\mathrm{Map}\left(\begin{array}{c} \underset{\{1,\cdots,N+n\}}{\mathrm{Conf}}(\mathbb{C}) \\ \big\downarrow {\scriptstyle (\mathrm{pr}_N^{N+n}, \tau_{(\kappa,\mathrm{w}_\bullet)})} \\ \underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C}) \times B\mathbb{C}^\times \end{array} ,\ p^*_{\underset{\{1,\cdots,N\}}{\mathrm{Conf}}}\left(\begin{array}{c} K(\mathbb{C},n) \times_{\mathbb{C}^\times} E\mathbb{C}^\times \\ \big\downarrow \\ B\mathbb{C}^\times \end{array} \right) \right)$
$\vec{z} : \mathbf{B}\,\mathrm{PBr}(N)\ \vdash$ $\quad (t : \mathbf{B}\mathbb{C}^\times) \to T_{\vec{z},t}\ :\ \mathrm{Type}$	(67) (33)	$\big(\mathrm{id}_{\underset{\{1,\cdots,N\}}{\mathrm{Conf}}} \times P_{B\mathbb{C}^\times}\big)_* \begin{array}{c} T \\ \big\downarrow \\ \underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C}) \times B\mathbb{C}^\times \end{array}$
$\vec{z} : \mathbf{B}\,\mathrm{PBr}(N),\ t : \mathbf{B}\mathbb{C}^\times\ \vdash$ $\quad \big[(t : \mathbf{B}\mathbb{C}^\times) \to T_{\vec{z},t}\ :\ \mathrm{Type}\big]_0$	(129)	$\pi_{0/\underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C})}\left(\big(\mathrm{id}_{\underset{\{1,\cdots,N\}}{\mathrm{Conf}}} \times P_{B\mathbb{C}^\times}\big)_* \begin{array}{c} T \\ \big\downarrow \\ \underset{\{1,\cdots,N\}}{\mathrm{Conf}}(\mathbb{C}) \times B\mathbb{C}^\times \end{array} \right)$
Homotopy type structure of Def. 6.7 specializing Def. 5.16	$\leftrightarrow$	Fibration of conformal blocks (58) via Thm. 4.13 & Ex. 4.14

$\square$

References

[Ab15] C. A. Abad, *Introduction to representations of braid groups*, Rev. Colomb. Mat. **49** 1 (2015) 1, [`arXiv:1404.0724`], [`doi:10.15446/recolma.v49n1.54160`].

[Ac11] P. Aczel, *On Voevodsky's Univalence Axiom*, talk at the *Third European Set Theory Conference* (2011), [`ncatlab.org/nlab/files/Aczel-Univalence.pdf`]

[AHS90] J. Adamek, H. Herrlich, and G. Strecker, *Abstract and Concrete Categories*, John Wiley and Sons, New York (1990); reprinted as: Reprints in Th. Appl. Categ. **17** (2006), 1-507, [tac:tr17].

[Ad74] J. F. Adams, *Stable homotopy and generalized homology*, Chicago Lectures in Math., University of Chicago Press, 1974, [`ucp:bo21302708`].

[Ad78] J. F. Adams, *Infinite loop spaces*, Annals of Mathematics Studies **90**, Princeton University Press (1978), [`doi:10.1515/9781400821259`].

[AGP02] M. Aguilar, S. Gitler, and C. Prieto, *Algebraic topology from a homotopical viewpoint*, Springer, Berlin (2002), [`doi:10.1007/b97586`].

[AvDKLLR07] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, *Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation*, SIAM J. Computing **37** 1 (2007), 166-194, [`jstor:20454175`], [`arXiv:quant-ph/0405098`].

[AGM00] O. Aharony, S. Gubser, J. Maldacena, H. Ooguri, and Y. Oz, *Large N Field Theories, String Theory and Gravity*, Phys. Rep. **323** (2000), 183-386, [`doi:10.1016/S0370-1573(99)00083-6`], [`arXiv:hep-th/9905111`].

[AN19] B. Ahrens and P. R. North, *Univalent foundations and the equivalence principle*, in: *Reflections on the Foundations of Mathematics*, Synthese Library **407**, Springer, Berlin (2019), [`doi:10.1007/978-3-030-15655-8`], [`arXiv:2202.01892`].

[ANST20] B. Ahrens, P. R. North, M. Shulman, and D. Tsementzis, *A Higher Structure Identity Principle*, LICS '20 (2020), 53–66, [`doi:10.1145/3373718.3394755`], [`arXiv:2004.06572`].

[Am+20] B. Aman, G. Ciobanu, R. Glück, R. Kaarsgaard, J. Kari, M. Kutrib, I. Lanese, C. Antares Mezzina, Ł. Mikulski, R. Nagarajan, I. Phillips, G. M. Pinna, L. Prigioniero, I. Ulidowski, and G. Vidal, *Foundations of Reversible Computation*, in: *Reversible Computation: Extending Horizons of Computing. RC 2020*, Lec. Notes Comp. Sci. **12070**, Springer, New York (2020), [`doi:10.1007/978-3-030-47361-7_1`].

[AF92] F. W. Anderson and K. R. Fuller, *Rings and Categories of Modules*, Graduate Texts in Mathematics **13** Springer, Berlin (1992), [`doi:10.1007/978-1-4612-4418-9`].

[ABGHR14] M. Ando, A. Blumberg, D. Gepner, M. Hopkins, and C. Rezk, *An ∞-categorical approach to R-line bundles, RR-module Thom spectra, and twisted R-homology*, J. Topology **7** 3 (2014), 869–893, [`arXiv:1403.4325`], [`doi:10.1112/jtopol/jtt035`].

[Ar11] M. Arkowitz, *Introduction to Homotopy Theory*, Springer, Berlin (2011), [`doi:10.1007/978-1-4419-7329-0`].

[Ar25] E. Artin, *Theorie der Zöpfe*, Abh. Math. Semin. Univ. Hambg. **4** (1925), 47–72, [`doi:10.1007/BF02950718`].

[ASY87] J. E. Avron, R. Seiler, and L. G. Yaffe, *Adiabatic theorems and applications to the quantum Hall effect*, Commun. Math. Phys. **110** (1987), 33–49, [`doi:10.1007/BF01209015`].

[Aw12] S. Awodey, *Type theory and homotopy*, in: *Epistemology versus Ontology*, Springer, Berlin (2012), 183-201, [`doi:10.1007/978-94-007-4435-6_9`], [`arXiv:1010.1810`].

[Aw13] S. Awodey (2013). *Natural Models of Homotopy Type Theory (Abstract)*. in: *Logic, Language, Information, and Computation*. WoLLIC 2013. Lecture Notes in Computer Science **8071**, Springer, Berlin, Heidelberg. []doi:10.1007/978-3-642-39992-3_2 [`arXiv:1406.3219`].

[AB04] S. Awodey and A. Bauer, *Propositions as [Types]*, J. Logic and Computation **14** (2004), 447-471, [`doi:10.1093/logcom/14.4.447`].

[AGS12] S. Awodey, N. Gambino, and K. Sojakova, *Inductive types in homotopy type theory*, LICS'12 (2012), 95–104, [`doi:10.1109/LICS.2012.21`], [`arXiv:1201.3898`].

[AGS15] S. Awodey, N. Gambino and K. Sojakova, *Homotopy-initial algebras in type theory*, J. Assoc. Comput. Machin. **63**6 (2017), 1–45, [`doi:10.1145/3006383`], [`arXiv:1504.05531`].

[AW09] S. Awodey and M. Warren, *Homotopy theoretic models of identity type*, Math. Proc. Cambridge Philosophical Society **146** (2009), 45-55, [`doi:10.1017/S0305004108001783`], [`arXiv:0709.0248`].

[BDF20] S. Bachmann, W. De Roeck, and M. Fraas, *The adiabatic theorem in a quantum many-body setting*, in: Analytic Trends in Mathematical Physics, Cont. Math. **741** (2020), 43-58, [`ams:conm-741`], [`arXiv:1808.09985`].

[BBP97] D. Bailey, P. Borwein and S. Plouffe, *On the Rapid Computation of Various Polylogarithmic Constants*, Math. Comp. **66** (1997), 903-913, [`doi:1997-66-218/S0025-5718-97-00856-9`].

[Ba15] P. Balmer, *Stacks of group representations*, J. Eur. Math. Soc. **17** 1 (2015), 189-228, [`doi:10.4171/jems/501`], [`arXiv:1302.6290`].

[BCP03] G. Barthe, V. Capretta, and O. Pons, *Setoids in type theory*, J. Funct. Progr. **13** 2 (2003), 261-293, [`doi:10.1017/S0956796802004501`].

[BP01] G. Barthe and O. Pons, *Type Isomorphisms and Proof Reuse in Dependent Type Theory*, in: *Foundations of Software Science and Computation Structures. FoSSaCS 2001*, Lecture Notes in Computer Science **2030**, Springer, Berlin (2001), [`doi:10.1007/3-540-45315-6_4`].

[Bau95] H. J. Baues, *Homotopy Types* in *Handbook of Algebraic Topology*, North Holland (1995), 1-72, [`doi:10.1016/B978-0-444-81779-2.X5000-7`], [`ncatlab.org/nlab/files/Baues-HomotopyTypes.pdf`]

[BCR18] G. Benenti, G. Casati, and D. Rossini, *Principles of Quantum Computation and Information*, World Scientific, Singapore (2004, 2018), [`doi:10.1142/10909`].

[Be80] P. Benioff, *The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines*, J. Stat. Phys. **22** (1980), 563–591, [`doi:10.1007/BF01011339`].

[Be03] C. H. Bennett, *Notes on Landauer's principle, reversible computation, and Maxwell's Demon*, Studies in History and Philosophy of Modern Physics **34** 3 (2003), 501-510, [`doi:10.1016/S1355-2198(03)00039-X`].

[Be84] M. V. Berry, *Quantal phase factors accompanying adiabatic changes*, Proc. R. Soc. Lond. A **392** (1984), 45–57, [`doi:10.1098/rspa.1984.0023`].

[Be89] M. Berry, *The quantum phase, five years after*, in: *Geometric phases in physics*, Adv. Ser. Math. Phys. **5**, World Scientific (1989), 7–28, [`doi:10.1142/0613`].

[BBCDG21] M. Bezem, U. Buchholtz, P. Cagne, B. I. Dundas, and D. R. Grayson: *Symmetry* (2021), [`unimath.github.io/SymmetryBook/book.pdf`], [`github.com/UniMath/agda-unimath/tree/master/src/group-theory`].

[BvN36] G. Birkhoff and J. von Neumann, *The logic of quantum mechanics*, Ann. Math. **37** (1936), 823-843, [`doi:10.2307/1968621`].

[Bir75] J. S. Birman, *Braids, links, and mapping class groups*, Princeton Univ Press (1975) [`ISBN:9780691081496`]

[Bish67] E. Bishop, *Foundations of Constructive Analysis*, McGraw-Hill (1967), [`archive.org/details/foundationsofcon0000bish`].

[BB85] E. Bishop and D. Bridges, *Constructive Analysis*, Grund. math. Wiss. **279** Springer, Berlin (1985), [`doi:10.1007/978-3-642-61667-9`].

[Bl11] P. E. Bland, *Rings and Their Modules*, De Gruyter, Berlin (2011), [`doi:10.1515/9783110250237`].

[BPV06] A. Bohannon, B. C. Pierce and J. A. Vaughan, *Relational lenses: a language for updatable views*, Proceedings of Principles of Database Systems (2006), 338-347, [`doi:10.1145/1142351.1142399`].

[BHZS05] N. E. Bonesteel, L. Hormozi, G. Zikos, and S. H. Simon, *Braid Topologies for Quantum Computation*, Phys. Rev. Lett. **95** (2005) 140503, [`doi:10.1103/PhysRevLett.95.140503`], [`arXiv:quant-ph/0505065`].

[Bo70] P. I. Booth, *The Exponential Law of Maps I*, Proc. London Math. Soc. **s3-20** (1970), 179-192, [`doi:10.1112/plms/s3-20.1.179`].

[BB78] P. I. Booth and R. Brown, *Spaces of partial maps, fibred mapping spaces and the compact-open topology*, Gen. Topol. Appl. **8** (1978), 181-195, [`doi:10.1016/0016-660X(78)90049-1`].

[Bo94II] F. Borceux, *Categories and Structures*, Vol. 2 of: *Handbook of Categorical Algebra*, Enc. Math. Appl. **50** Cambridge Univ. Press (1994), [`doi:10.1017/CBO9780511525865`].

[Bo94III] F. Borceux, *Categories of Sheaves*, Vol. 3 of: *Handbook of Categorical Algebra*, Enc. Math. Appl. **50** Cambridge Univ. Press (1994), [`doi:10.1017/CBO9780511525872`].

[BEZ20] D. Bouwmeester, A. Ekert, and A. Zeilinger, *The Physics of Quantum Information – Quantum Cryptography, Quantum Teleportation, Quantum Computation*, Springer, New York (2020), [`doi:10.1007/978-3-662-04209-0`].

[BM21] V. Braunack-Mayer, *Combinatorial parametrised spectra*, Algebr. Geom. Topol. **21** (2021), 801-891, [`doi:10.2140/agt.2021.21.801`], [`arXiv:1907.08496`].

[Br93] G. Bredon, *Topology and Geometry*, Graduate Texts in Math. **139**, Springer, Berlin (1993), [`doi:10.1007/978-1-4757-6848-0`].

[BP08] G. K. Brennen and J. K. Pachos, *Why should anyone care about computing with anyons?*, Proc. R. Soc. A **464** (2008), 1-24, [`doi:10.1098/rspa.2007.0026`], [`arXiv:0704.2241`].

[Br99] D. Bridges, *Constructive mathematics: a foundation for computable analysis*, Theor. Comp Sci. **219** 1–2 (1999), 95-109, [`doi:10.1016/S0304-3975(98)00285-0`].

[Br65] E. H. Brown, Jr., *Abstract homotopy theory*, Trans. Amer. Math. Soc. **119** (1965), 79–85, [`doi:10.1090/S0002-9947-1965-0182970-6`].

[Br73] K. S. Brown, *Abstract Homotopy Theory and Generalized Sheaf Cohomology*, Trans. Amer. Math. Soc., **186** (1973), 419-458, [`jstor:1996573`].

[Br14] J. W. Brunekreef, *Topological Quantum Computation and Quantum Compilation*, thesis, Utrecht (2014), [`studentheses:20.500.12932/17738`].

[BLL13] G. Brunerie, D. Licata, and P. Lumsdaine, *Homotopy theory in type theory*, lecture notes (2013), [`dlicata.wescreates.wesleyan.edu/pubs/bll13homotopy/bll13homotopy.pdf`]

[BLM22] G. Brunerie, A. Ljungström, and A. Mörtberg, *Synthetic Integral Cohomology in Cubical Agda*, 30th EACSL Annual Conference on Computer Science Logic (CSL 2022) **216** (2022), [`doi:10.4230/LIPIcs.CSL.2022.11`].

[BCTFR23] U. Buchholtz, J. D. Christensen, J. Taxerás Flaten, and E. Rijke, *Central H-spaces and banded types*, [`arXiv:2301.02636`].

[BvDR18] U. Buchholtz, F. van Doorn, and E. Rijke, *Higher Groups in Homotopy Type Theory*, LICS **33** (2018), 205-214, [`doi:10.1145/3209108.3209150`], [`arXiv:1802.04315`].

[Bun76] M. Bunge, *Possibility and Probability*, pp. 17-34 in: *Foundations of Probability Theory, Statistical Interference, and Statistical Theories of Science*, Reidel Publishing (1976), [`doi:10.1007/978-94-010-1438-0_2`].

[BNV16] U. Bunke, T. Nikolaus, and M. Völkl, *Differential cohomology theories as sheaves of spectra*, J. Homotopy Rel. Struct. **11** (2016), 1–66, [`doi:10.1007/s40062-014-0092-5`], [`arXiv:1311.3188`].

[CR97] D. C. Cabra and G. L. Rossini, *Explicit connection between conformal field theory and 2+1 Chern-Simons theory*, Mod. Phys. Lett. A **12** (1997), 1687-1697, [`doi:10.1142/S0217732397001722`], [`arXiv:hep-th/9506054`].

[CMV95] F. Cagliari, S. Mantovani, and E. Vitale, *Regularity of the category of Kelley spaces*, Appli. Categ. Struc. **3** (1995), 357–361, [`doi:10.1007/BF00872904`].

[CGX01] A. Cattaneo, A. Giaquinto and P. Xu, *Higher Structures in Geometry and Physics – In Honor of Murray Gerstenhaber and Jim Stasheff*, Progress in Mathematics **287**, Birkhäuser, (2001), [`doi:10.1007/978-0-8176-4735-3`].

[Ca15] E. Cavallo, *Synthetic Cohomology in Homotopy Type Theory.*, PhD Thesis, Carnegie Mellon University (2015), [`staff.math.su.se/evan.cavallo/works/thesis15.pdf`]

[CLBFN15] C. Cesare, A. J. Landahl, D. Bacon, S. T. Flammia, and A. Neels, *Adiabatic topological quantum computing*, Phys. Rev. A **92** (2015) 012336, [`doi:10.1103/PhysRevA.92.012336`], [`arXiv:1406.2690`].

[CGD11] M. Cheng, V. Galitski, and S. Das Sarma, *Non-adiabatic Effects in the Braiding of Non-Abelian Anyons in Topological Superconductors*, Phys. Rev. B **84** (2011) 104529, [`doi:10.1103/PhysRevB.84.104529`], [`arXiv:1106.2549`].

[CR21] F. Cherubini and E. Rijke, *Modal Descent*, Math. Struc. Comp. Sci. **31** 4 (2021), 363-391, [`arXiv:2003.09713`], [`doi:10.1017/S0960129520000201`].

[CFP02] A. Childs, E. Farhi and J. Preskill, *Robustness of adiabatic quantum computation*, Phys.Rev. A **65** (2002) 012322 [`arXiv:quant-ph/0108048`] [`doi:10.1103/PhysRevA.65.012322`]

[Ch07] A. Chlipala, *Implementing Certified Programming Language Tools in Dependent Type Theory*, PhD thesis, U. California at Berkeley (2007), [`UCB/EECS-2007-113`].

[Ch13] A. Chlipala, *Certified programming with dependent types*, MIT Press, Boston (2013), [`ISBN:9780262026659`].

[Ch40] A. Church, *A Formulation of the Simple Theory of Types*, J. Symbolic Logic **5** 2 (1940), 56-68, [`doi:doi:10.2307/2266170`].

[Ci19] D.-C. Cisinski, *Higher Categories and Homotopical Algebra*, Cambridge University Press (2019), [`doi:10.1017/9781108588737`].

[CW08] J. Clarke and F. K. Wilhelm, *Superconducting quantum bits*, Nature **453** (2008), 1031–1042, [`doi:10.1038/nature07128`].

[CT08] C. S. Coen and E. Tassi, *Working with Mathematical Structures in Type Theory*, in: *Types for Proofs and Programs. TYPES 2007*, Lecture Notes in Computer Science **4941** Springer (2008), [`doi:10.1007/978-3-540-68103-8_11`].

[Co09] F. R. Cohen, *Introduction to configuration spaces and their applications*, in *Braids*, Lecture Notes Series, Institute for Mathematical Sciences, **19** (2009), 183-261, World Scientific, Singapore, [`doi:10.1142/9789814291415_0003`].

[CCHM15] C. Cohen, T. Coquand, S. Huber, A. Mörtberg, *Cubical Type Theory: a constructive interpretation of the univalence axiom*, 21st International Conference on Types for Proofs and Programs (TYPES 2015), 5.1-5.34, Leibniz International Proceedings in Informatics (LIPIcs), Dagstuhl, Germany [`doi:10.48550/arXiv.1611.02108`].

[Co11] T. Coquand, *Equality and dependent type theory*, lecture notes (2011), [`ncatlab.org/nlab/files/Coquand-EqualityAndDependentTypeTheory.pdf`]

[CD13] T. Coquand and N. A. Danielsson, *Isomorphism is equality*, Indagationes Mathematicae **24** 4 (2013), 1105-1120, [`doi:10.1016/j.indag.2013.09.002`].

[CHM18] Thierry Coquand, Simon Huber, and Anders Mörtberg, *On Higher Inductive Types in Cubical Type Theory*, in Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS '18) (2018), Association for Computing Machinery, New York, NY, USA, 255–264. [`doi:10.1145/3209108.3209197`].

[CP90] T. Coquand and C. Paulin, *Inductively defined types*, COLOG-88 Lecture Notes in Computer Science **417**, Springer, Berlin (1990), 50-66, [`doi:10.1007/3-540-52335-9_47`].

[CS07] T. Coquand and A. Spiwack, *Towards constructive homological algebra in type theory*, in: *Towards Mechanized Mathematical Assistants. MKM Calculemus 2007*, Lecture Notes in Computer Science **4573** Springer, Berlin (2007), [`doi:10.1007/978-3-540-73086-6_4`].

[O'C07] R. O'Connor, *A Monadic, Functional Implementation of Real Numbers*, Math. Struc. Comp. Sci. **17** 1 (2007), 129-159, [`doi:10.1017/S0960129506005871`], [`arXiv:cs/0605058`].

[Co11] R. Constable, *The Triumph of Types: Creating a Logic of Computational Reality*, lecture at: *Types, Semantics and Verification*, Oregon (2011),
[`www.cs.uoregon.edu/research/summerschool/summer11/lectures/Triumph-of-Types-Extended.pdf`]

[CSS21] D. Corfield, H. Sati, and U. Schreiber, *Fundamental weight systems are quantum states*, [`arXiv:2105.02871`].

[Co04] L. Corry, *Modern Algebra and the Rise of Mathematical Structures*, Springer, Berlin (2004), [`doi:10.1007/978-3-0348-7917-0`].

[Co15] L. Corry, *A Brief History of Numbers*, Oxford University Press (2015), [`ISBN:9780198702597`].

[Co00] D. G. Cory, R. Laflamme, E. Knill, L. Viola, T. F. Havel, N. Boulant, G. Boutis, E. Fortunato, S. Lloyd, R. Martinez, C. Negrevergne, M. Pravia, Y. Sharf, G. Teklemariam, Y.S. Weinstein, and W. H. Zurek, *NMR Based Quantum Information Processing: Achievements and Prospects*, Fortsch. Phys. **48** (2000), 875-907, [`arXiv:quant-ph/0004104`], [`doi:10.1002/1521-3978(200009)48:9/11<875::AID-PROP875>3.0.CO;2-V`].

[CJ98] M. C. Crabb and I. M. James, *Fiberwise homotopy theory*, Springer (1998), [`doi:10.1007/978-1-4471-1265-5`].

[CGH14] P.-L. Curien, R. Garner, and M. Hofmann, *Revisiting the categorical interpretation of dependent type theory*, Theor. Comp. Sci. **546** 21 (2014), 99-119, [`doi:10.1016/j.tcs.2014.03.003`].

[DS22a] S. Das Sarma, *Quantum computing has a hype problem*, MIT Technology Review (March 2022),
[`www.technologyreview.com/2022/03/28/1048355/quantum-computing-has-a-hype-problem`].

[DS22b] S. Das Sarma, *In search of Majorana*, [`arXiv:2210.17365`].

[DSP21] S. Das Sarma and H. Pan, *Disorder-induced zero-bias peaks in Majorana nanowires*, Phys. Rev. B **103** (2021) 195158, [`doi:10.1103/PhysRevB.103.195158`], [`arXiv:2103.05628`].

[DJMM90] E. Date, M. Jimbo, A. Matsuo, and T. Miwa, *Hypergeometric-type integrals and the $\mathfrak{sl}(2,\mathbb{C})$ Knizhnik-Zamolodchikov equation*, Int. J. Mod. Phys. B **4** 5 (1990), 1049-1057, [`doi:10.1142/S0217979290000528`].

[DN06] C. M. Dawson and M. A. Nielsen, *The Solovay-Kitaev algorithm*, Quantum Info. Comput. **6** 1 (2006), 81–95, [`doi:10.5555/2011679.2011685`], [`arXiv:quant-ph/0505030`].

[dB91] N. de Bruijn, *Telescopic mappings in typed lambda calculus*, Information and Computation **91** 2 (1991), 189-204, [`doi:10.1016/0890-5401(91)90066-B`].

[De70] P. Deligne, *Equations différentielles à points singuliers règuliers*, Lect. Notes Math. **163** Springer (1970), [`ias:355`].

[De89] D. E. Deutsch, *Quantum computational networks*, Proc. of the Royal Society A **425** 1868 (1989), 73-90, [`doi:10.1098/rspa.1989.0099`].

[DiC95] R. Di Cosmo, *Isomorphisms of Types – from λ-calculus to information retrieval and language design*, Progress in Theoretical Computer Science, Birkhäuser (1995), [`doi:10.1007/978-1-4612-2572-0`].

[FMS97] P. Di Francesco, P. Mathieu and D. Sénéchal, *Conformal field theory*, Springer, Berlin (1997), [`doi:10.1007/978-1-4612-2256-9`].

[Di04] A. Dimca, *Sheaves in Topology*, Springer, Berlin (2004), [`doi:10.1007/978-3-642-18868-8`].

[DRS22] M. Ding, C. D. Roberts, and S. M. Schmidt, *Emergence of Hadron Mass and Structure*, [`arXiv:2211.07763`].

[DA18] P. Downen and Z. M. Ariola, *A tutorial on computational classical logic and the sequent calculus*, J. Funct. Progr. **28** (2018) E3, [`doi:10.1017/S0956796818000023`].

[DN08] M. R. Dowling and M. A. Nielsen, *The geometry of quantum computation*, Quantum Info. Comput. **8** 10 (2008), 861–899, [`doi:10.5555/2016985.2016986`], [`arXiv:quant-ph/0701004`].

[Du99] M. Duff, *The World in Eleven Dimensions: Supergravity, Supermembranes and M-theory*, IoP, Bristol (1999), [`ISBN:9780750306720`].

[DS95] W. Dwyer and J. Spalinski, *Homotopy theories and model categories* in: I. M. James (ed.), *Handbook of Algebraic Topology*, North Holland (1995), [`doi:10.1016/B978-0-444-81779-2.X5000-7`].

[Dy94] P. Dybjer, *Inductive families*, Formal Aspects of Computing **6** (1994), 440–465, [`doi:10.1007/BF01211308`].

[Dy96] P. Dybjer, *Internal Type Theory*, Types for Proofs and Programs. TYPES 1995. Lecture Notes in Computer Science **1158** Springer (1995) [`doi:10.1007/3-540-61780-9_66`].

[Dy97] P. Dybjer, *Representing inductively defined sets by wellorderings in Martin-Löf's type theory*, Theor. Comp. Sci. **176** 1–2 (1997), 329-335, [`doi:10.1016/S0304-3975(96)00145-4`].

[Eq22] A. Equbal, *Molecular spin qubits for future quantum technology*, Quantum Colloquium at CQTS (Nov 2022), [`ncatlab.org/nlab/show/CQTS#EqubalNov22`].

[Er15] J. Erlich, *An Introduction to Holographic QCD for Nonspecialists*, Contemp. Phys. **56** (2015) 2, [`arXiv:1407.5002`].

[Es19] M. H. Escardó, *Introduction to Univalent Foundations of Mathematics with Agda* (2019), [`arXiv:1911.00580`], [`cs.bham.ac.uk/~mhe/HoTT-UF-in-Agda-Lecture-Notes`].

[EFK98] P. I. Etingof, I. Frenkel, and A. A Kirillov, *Lectures on Representation Theory and Knizhnik-Zamolodchikov Equations*, Math. Surv. monogr. **58**, Amer. Math. Soc., Providence, RI (1998), [`ams.org/surv-58`].

[FaN62] E. Fadell and L. Neuwirth, *Configuration spaces*, Math. Scand. **10** (1962), 111-118, [`doi:10.7146/math.scand.a-10517`].

[FH01] E. Fadell and S. Husseini, *Geometry and topology of configuration spaces*, Springer Monographs in Mathematics, Berlin (2001), [`doi:10.1007/978-3-642-56446-8`].

[FGGS00] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, *Quantum Computation by Adiabatic Evolution*, [`arXiv:quant-ph/0001106`].

[Fe82] R. Feynman, *Simulating physics with computers*, Int J Theor Phys **21** (1982) 467–488 [`doi:10.1007/BF02650179`]

[FSV94] B. Feigin, V. Schechtman, A. Varchenko, *On algebraic equations satisfied by hypergeometric correlators in WZW models. I*, Commun. Math. Phys. **163** (1994), 173–184, [`doi:10.1007/BF02101739`].

[FSS19] D. Fiorenza, H. Sati and U. Schreiber, *The rational higher structure of M-theory*, Fortsch. Phys. **67** (2019) 1910017, [`doi:10.1002/prop.201910017`], [`arXiv:1903.02834`].

[FSS20-Cha] D. Fiorenza, H. Sati, and U. Schreiber, *The Character Map in Non-Abelian Cohomology – Twisted, Differential and Generalized*,
World Scientific (2023) [`10.1142/13422`] [`arXiv:2009.11909`].

[FoN62] R. H. Fox and L. Neuwirth, *The braid groups*, Math. Scand. **10** (1962), 119-126, [`doi:10.7146/math.scand.a-10518`].

[FM12] D. S. Freed and G. W. Moore, *Twisted equivariant matter*, Ann. Henri Poincaré **14** (2013), 1927–2023, [`doi:10.1007/s00023-013-0236-x`], [`arXiv:1208.5055`].

[Fr98] M. Freedman, *P/NP, and the quantum field-computer*, Proc. Nat. Acad. Sci. **95** 1 (1998), 98-101, [`doi:10.1073/pnas.95.1.9`].

[FLW02b] M. H. Freedman, M. Larsen, and Z. Wang, *A modular functor which is universal for quantum computation*, Commun. Math. Phys. **227** (2002), 605-622, [`doi:10.1007/s002200200645`], [`arXiv:quant-ph/0001108`].

[FKLW03] M. Freedman, A. Kitaev, M. Larsen, and Z. Wang, *Topological Quantum Computation*, Bull. Am. Math. Soc. **40** (2003) 31, [`doi:10.1090/S0273-0979-02-00964-3`], [`arXiv:quant-ph/0101025`].

[Fr1879] G. Frege, *Begriffsschrift:eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*, Verlag von Louis Nebert (1879), [`ISBN:9783487006239`]; English translation: J. Corcoran and D. Levin, *Gottlob Frege: Conceptual notation and related articles*, Oxford University Press (1972), [`doi:10.1086/288549`].

[Fu15] L. Fuchs, *Abelian Groups*, Springer, Berlin (2015), [`doi:10.1007/978-3-319-19422-6`].

[GL17] D. Gaitsgory and J. Lurie, *Weil's conjecture for function fields* (2014-2017), [`www.math.ias.edu/~lurie/papers/tamagawa-abridged.pdf`]

[GGMR09] F. Garillot, G. Gonthier, A. Mahboubi, and L. Rideau, *Packaging Mathematical Structures*, in: *Theorem Proving in Higher Order Logics. TPHOLs 2009*, Lecture Notes in Computer Science **5674** Springer, Berlin (2009), [`doi:10.1007/978-3-642-03359-9_23`].

[Ga00] K. Gawedzki, *Conformal field theory: a case study*, in *Conformal Field Theory – New Non-perturbative Methods In String And Field Theory*, CRC Press (2000), [`doi:10.1201/9780429502873`], [`arXiv:hep-th/9904145`].

[GM56] M. Gell-Mann, *The interpretation of the new particles as displaced charge multiplets*, Nuovo Cim **4** 2 (1956), 848–866, [`doi:10.1007/BF02748000`].

[Ge34] G. Gentzen, *Untersuchungen über das logische Schließen*, Math. Zeitschrift **39** (1935), 176–210, [`doi:10.1007/BF01201353`].

[Ge69] G. Gentzen, *Investigations into Logical Deduction*, translated by M. E. Szabo (ed.) in: *The Collected Papers of Gerhard Gentzen*, Studies in Logic and the Foundations of Mathematics **55**, Springer, Berlin (1969), 68-131, [`ISBN:9780444534194`].

[GNSW07] H. Geuvers, M. Niqui, B. Spitters, and F. Wiedijk, *Constructive analysis, types and exact real numbers*, Math. Struc. Comp. Sci. **17** 01 (2007), 3-36, [`doi:10.1017/S0960129506005834`].

[GJ99] P. Goerss and J. F. Jardine, *Simplicial homotopy theory*, Progress in Mathematics, Birkhäuser, Boston (2009), [`doi:10.1007/978-3-0346-0189-4`].

[Gö18] L. Götz, *Martin-Löf's J-rule*, Bachelor's Thesis, Ludwig Maximili an Univ., Munich, 2018, [`math.lmu.de/~petrakis/Goetz.pdf`]

[GIB12] D. Golze, M. Icker, and S. Berger, *Implementation of two-qubit and three-qubit quantum computers using liquid-state nuclear magnetic resonance*, Concepts in Magnetic Resonance **40**A 1 (2012), 25-37, [`doi:10.1002/cmr.a.21222`].

[GPP21] A. Govzmann, D. Pištalo, and N. Poncin, *Indeterminacies and models of homotopy limits*, [`arXiv:2109.12395`].

[GS19] D. Grady and H. Sati, *Twisted differential* KO-*theory*, [`arXiv:1905.09085`].

[GH20] E. K. Grant and T. S. Humble, *Adiabatic Quantum Computing and Quantum Annealing*, Oxford Research Encyclopedia (2020), [`doi:10.1093/acrefore/9780190871994.013.32`].

[Gri70] P. A. Griffiths, *Periods of integrals on algebraic manifolds: Summary of main results and discussion of open problems*, Bull. Amer. Math. Soc. **76** (1970), 228-296, [`doi:10.1090/S0002-9904-1970-12444-2`].

[Gro66] A. Grothendieck, *On the de Rham cohomology of algebraic varieties*, Inst. Hautes Études Sci. Publ. Math. **29** (1966), 95-103, [`http://www.numdam.org/item/PMIHES_1966__29__95_0`].

[GHL22] X. Gu, B. Haghighat, and Y. Liu, *Ising- and Fibonacci-Anyons from KZ-equations*, J. High Energy Phys. **2022** (2022) 15, [`arXiv:2112.07195`][`doi:10.1007/JHEP09(2022)015`].

[Gu95] C. A. Gunter, *The Semantics of Types in Programming Languages*, in: *Handbook of Logic in Computer Science*, Vol 3: *Semantic structures*, Oxford University Press (1995), [`ISBN:9780198537625`].

[HKK02] K. A. Hardie, K. H. Kamps, and R. Kieboom, *A Homotopy Bigroupoid of a Topological Space*, Appl. Categ. Struct. **9** (2001), 311-327, [`doi:10.1023/A:1011270417127`].

[Ha16] R. Harper, *Practical Foundations for Programming Languages*, Cambridge University Press (2016), [`ISBN:9781107150300`].

[HLS18] S. Hartnoll, A. Lucas, and S. Sachdev, *Holographic quantum matter*, MIT Press, Boston, (2018), [`ISBN:9780262348010`], [`arXiv:1612.07324`].

[Ha02] A. Hatcher, *Algebraic Topology*, Cambridge University Press (2002) [`ISBN:9780521795401`], [`https://pi.math.cornell.edu/~hatcher/AT/ATpage.html`].

[Hi88] P. Hilton, *Subjective History of Homology and Homotopy Theory*, Mathematics Magazine **61** 5 (1988), 282-291, [`doi:10.2307/2689545`].

[Ho95] M. Hofmann, *Extensional concepts in intensional type theory*, Ph.D. thesis, Edinburgh (1995), Distinguished Dissertations, Springer (1997), [`ECS-LFCS-95-327`],[`doi:10.1007/978-1-4471-0963-1`].

[Ho97] M. Hofmann, *Syntax and semantics of dependent types*, in: *Semantics and logics of computation*, Publ. Newton Inst. **14**, Cambridge Univ. Press (1997), 79-130, [`doi:10.1017/CBO9780511526619.004`].

[HS98] M. Hofmann and T. Streicher, *The groupoid interpretation of type theory*, in: *Twenty-five years of constructive type theory*, Oxf. Logic Guides. **36**, Clarendon Press (1998) 83-111, [`ISBN:9780198501275`].

[HBS09] L. Hormozi, N. E. Bonesteel and S. H. Simon, *Topological Quantum Computing with Read-Rezayi States*, Phys. Rev. Lett. **103** (2009) 160501, [`doi:10.1103/PhysRevLett.103.160501`], [`arXiv:0903.2239`].

[HZBS07] L. Hormozi, G. Zikos, N. E. Bonesteel, and S. H. Simon, *Topological Quantum Compiling*, Phys. Rev. B **75** (2007) 165310, [`doi:10.1103/PhysRevB.75.165310`], [`arXiv:quant-ph/0610111`].

[HFLL1616] K.-B. Hou, E. Finster, D. Licata, and P. Lumsdaine, *A mechanization of the Blakers-Massey connectivity theorem in Homotopy Type Theory*, LICS '16 (2016), 565–574, [`doi:10.1145/2933575.2934545`], [`arXiv:1605.03227`].

[HWFZ20] H.-L. Huang, D. Wu, D. Fan, and X. Zhu, *Superconducting Quantum Computing: A Review*, Science China Information Sciences **63** 8 (2020), 1-32, [`doi:10.1007/s11432-020-2881-9`], [`arXiv:2006.10433`].

[Ja93] B. Jacobs, *Comprehension categories and the semantics of type dependency*, Theor. Comp. Sci. **107** 2 (1993) 169-207 [`doi:10.1016/0304-3975(93)90169-T`].

[Ja98] B. Jacobs, *Categorical Logic and Type Theory*, Studies in Logic and the Foundations of Mathematics **141**, Elsevier (1998), [`ISBN:9780444501707`].

[Ja84] I. M. James, *General Topology and Homotopy Theory*, Springer, Berlin (1984), [`doi:10.1007/978-1-4613-8283-6`].

[Ja84] K. Jänich, *Topology*, Undergraduate Texts in Mathematics, Springer, Berlin (1984), [`ISBN:9780387908922`].

[JS21] E. G. Johansen and T. Simula, *Fibonacci anyons versus Majorana fermions – A Monte Carlo Approach to the Compilation of Braid Circuits in* $SU(2)_k$ *Anyon Models*, PRX Quantum **2** (2021) 010334, [`arXiv:2008.10790`], [`doi:10.1103/PRXQuantum.2.010334`].

[Jo90] D. L. Johnson, *Presentations of Groups*, Cambridge Univ. Press (1990), [`doi:10.1017/CBO9781139168410`].

[Jo17] A. Joyal, *Notes on Clans and Tribes*, [`arXiv:1710.10238`].

[JSSW19] B. Jurčo, C. Saemann, U. Schreiber, and M. Wolf, *Higher Structures in M-Theory*, Fortsch. Phys. **67** (2019), 8-9, [`doi:10.1002/prop.201910001`], [`arXiv:1903.02807`].

[Ka09] F. Kachapova, *Formalizing groups in type theory*, [`arXiv:2102.09125`].

[KN98] T. Kadowaki and H. Nishimori, *Quantum annealing in the transverse Ising model*, Phys. Rev. **E. 58** (1998), 5355-5363, [`doi:10.1103/PhysRevE.58.5355`], [`arXiv:cond-mat/9804280`].

[KP97] K. H. Kamps and T. Porter, *Abstract Homotopy and Simple Homotopy Theory*, World Scientific, Singapore (1997), [`doi:10.1142/2215`].

[KLV12] C. Kapulkin, P. LeFanu Lumsdaine, and V. Voevodsky, *Univalence in simplicial sets*, [arXiv:1203.2553].

[KL21] C. Kapulkin and P. LeFanu Lumsdaine, *The Simplicial Model of Univalent Foundations (after Voevodsky)*, J. Eur. Math. Soc. **23** (2021), 2071–2126, [doi:10.4171/jems/1050], [arXiv:1211.2851].

[Ka68] N. M. Katz, *On the differential equations satisfied by period matrices*, Inst. Hautes Études Sci. Publ. Math. **35** (1968), 223–258, [http://www.numdam.org/item/PMIHES_1968__35__71_0/].

[KO68] N. M. Katz and T. Oda, *On the differentiation of de Rham cohomology classes with respect to parameters*, J. Math. Kyoto Univ. **8** (1968), 199-213, [doi:10.1215/kjm/1250524135].

[KL04] L. H. Kauffman and S. J. Lomonaco Jr., *Braiding Operators are Universal Quantum Gates*, New J. Phys. **6** (2004) 134, [doi:10.1088/1367-2630/6/1/134], [arXiv:quant-ph/0401090].

[Ki97] A. Kitaev, *Quantum computations: algorithms and error correction*, Russian Math. Surv. **52** (1997), 1191-1249, [doi:10.1070/rm1997v052n06abeh002155].

[Ki01] A. Kitaev, *Unpaired Majorana fermions in quantum wires*, Physics-Uspekhi **44** 131 (2001), 131-136, [doi:10.1070/1063-7869/44/10S/S29], [arXiv:cond-mat/0010440].

[Ki03] A. Kitaev, *Fault-tolerant quantum computation by anyons*, Annals Phys. **303** (2003), 2-30, [doi:10.1016/S0003-4916(02)00018-0], [arXiv:quant-ph/9707021].

[Ki06] A. Kitaev, *Anyons in an exactly solved model and beyond*, Ann. Phys. **321** 1 (2006) 2-111, [doi:10.1016/j.aop.2005.10.005], [arXiv:cond-mat/0506438].

[Ki09] A. Kitaev, *Periodic table for topological insulators and superconductors*, AIP Conference Proceedings **1134** (2009) 22, [doi:10.1063/1.3149495], [arXiv:0901.2686].

[KBS14] V. Kliuchnikov, A. Bocharov and K. M. Svore, *Asymptotically Optimal Topological Quantum Compiling*, Phys. Rev. Lett. **112** (2014) 140504 [arXiv:1310.4150][doi:10.1103/PhysRevLett.112.140504]

[Kn06] A. Knapp, *Basic Algebra*, Springer, Berlin (2006), [doi:10.1007/978-0-8176-4529-8].

[Kn96] E. Knill, *Conventions for quantum pseudocode*, Los Alamos Technical Report LA-UR-96-2724 (1996), [doi:10.2172/366453].

[Ko61] S. Kochen, *Ultraproducts in the Theory of Models*, Ann. Math. **74** 2 (1961), 221-261, [doi:10.2307/1970235].

[Ko02] T. Kohno, *Conformal field theory and topology*, Transl Math. Monogr. **210**, Amer. Math. Soc., Providence, RI (2002), [ams:mmono-210].

[Ko12] T. Kohno, *Homological representations of braid groups and KZ connections*, J. Singularities **5** (2012), 94-108, [doi:10.5427/jsing.2012.5g].

[KMM23] N. Kolganov, S. Mironov and A. Morozov, *Large k topological quantum computer*, Nucl. Phys. B **987** (2023) 116072 [arXiv:2105.03980] [doi:10.1016/j.nuclphysb.2023.116072]

[Ko32] A. Kolmogorov, *Zur Deutung der intuitionistischen Logik*, Math. Z. **35** (1932), 58-65, [doi:10.1007/BF01186549].

[Kr15] N. Kraus, *The General Universal Property of the Propositional Truncation*, in: *TYPES 2014*, Leibniz International Proceedings in Informatics (LIPIcs) **39** (2015) [arXiv:1411.2682] [doi:10.4230/LIPIcs.TYPES.2014.111]

[KS13] R. Krebbers and B. Spitters, *Type classes for efficient exact real arithmetic in Coq*, Logical Methods in Computer Science **9** 1 (2013) 958, [doi:10.2168/LMCS-9(1:1)2013], [arXiv:1106.3448].

[Kr07] R. Krömer, *Tool and object: A history and philosophy of category theory*, Springer, Berlin (2007), [doi:10.1007/978-3-7643-7524-9].

[LP15] J. Ladyman and S. Presnell, *Identity in Homotopy Type Theory, Part I: The Justification of Path Induction*, Philosophia Mathematica **23** 3 (2015), 386–406, [doi:10.1093/philmat/nkv014].

[La70] W. Lawvere, *Quantifiers and sheaves*, Actes Congrès intern. Math. **1** (1970), 329-334, [ncatlab.org/nlab/files/Lawvere-QuantifiersAndSheaves.pdf]

[La94] W. Lawvere, *Cohesive Toposes and Cantor's "lauter Einsen"*, Philosophia Mathematica **2** 1 (1994), 5-15, [doi:10.1093/philmat/2.1.5].

[La07] W. Lawvere, *Axiomatic cohesion*, Theory & Appl. of Categ., **19** 3 (2007), 41–49, [tac:19-3].

[Le10] E.-K. Lee, *A positive presentation of the pure braid group*, J. Chungcheong Math. Soc. **23** 3 (2010), 555-561, [JAKO201007648745187].

[LW17] J. van Leeuwen and J. Wiedermann: *Knowledge, Representation and the Dynamics of Computation*, Studies in Applied Philosophy **28** Springer, Berlin (2017), 69-89, [doi:10.1007/978-3-319-43784-2_5].

[LM77] J. M. Leinaas and J. Myrheim, *On the theory of identical particles*, Nuovo Cim B **37** (1977), 1–23, [doi:10.1007/BF02727953].

[Le92] A. Lerda, *Anyons – Quantum Mechanics of Particles with Fractional Statistics*, Lect. Notes Phys. **14**, Springer, Berlin (1992), [doi:10.1007/978-3-540-47466-1].

[Le18] C. I. Lewis, *A Survey of Symbolic Logic*, Univ. of California Press, Berkeley, (1918), [archive.org/details/surveyofsymbolic00lewiiala].

[LB20] F. Leymann and J. Barzen, *The bitter truth about gate-based quantum algorithms in the NISQ era*, Quantum Sci. Technol. **5** (2020) 044007, [`doi:10.1088/2058-9565/abae7d`].

[Li14] N. Li, *Quotient Types in Type Theory*, PhD Thesis, Nottingham (2014), [`eprints.nottingham.ac.uk:28941`], [`ncatlab.org/nlab/files/Li-QuotientTypes.pdf`]

[1lab] A. Liao, J. Coates, and R. Mullanix, *1lab*, [`https://1lab.dev`].

[LF14] D. Licata and E. Finster, *Eilenberg-MacLane spaces in homotopy type theory*, CSL-LICS '14 **66** (2014), 1-9, [`doi:10.1145/2603088.2603153`].

[Lu07] R. Lubarsky, *On the Cauchy Completeness of the Constructive Cauchy Reals*, Electronic Notes in Theoretical Computer Science **167** (2007), 225-254, [`doi:10.1016/j.entcs.2006.09.012`].

[Lü51] G. Lüders, *Über die Zustandsänderung durch den Meßprozeß*, Ann. Phys. **8** (1951), 322–328, [`doi:10.1002/andp.19504430510`]; translated as: *Concerning the state-change due to the measurement process*, Ann. Phys. **15** 9 (2006), 663-670, [`ncatlab.org/nlab/files/Lueders-StateChange.pdf`]

[Lu15] M. Lundfall, *Formalizing real numbers in Agda*, preprint (2015), [`ncatlab.org/nlab/files/Lundfall-RealNumbersInAgda.pdf`]

[Lu94] Z. Luo, *Computation and Reasoning – A Type Theory for Computer Science*, Clarendon Press, Oxford (1994), [`ISBN:9780198538356`].

[Lu09] J. Lurie, *Higher Topos Theory*, Ann. Math. Stud. **170**, Princeton University Press (2009), [`pup:8957`], [`arXiv:math/0608040`].

[Lu17] J. Lurie, *Higher Algebra*, [`https://www.math.ias.edu/~lurie/papers/HA.pdf`]

[MCMC19] E. Macaluso, T. Comparin, L. Mazza, and I. Carusotto, *Fusion Channels of Non-Abelian Anyons from Angular-Momentum and Density-Profile Measurements*, Phys. Rev. Lett. **123** (2019) 266801, [`arXiv:1903.03011`], [`doi:10.1103/PhysRevLett.123.266801`].

[MKS66] W. Magnus, A. Karras and D. Solitar, *Combinatorial group theory: Presentation of groups in terms of generators and relations*, Dover Publications (2004), [`ISBN-13:9780486438306`].

[Ma58] Y. I. Manin, *Algebraic curves over fields with differentiation* (Russian), Izv. Akad. Nauk SSSR. Ser. Mast. **22** (1958), 737–756, [`bookstore.ams.org/trans2-37`], [`books.google.com/books?id=fZ7ms3db_cMC`].

[Ma80] Y. I. Manin, *Computable and Uncomputable*, Sov. Radio (1980), published in: *Mathematics as Metaphor: Selected essays of Yuri I. Manin*, Collected Works **20**, AMS (2007), 69-77, [`ISBN:978-0-8218-4331-4`].

[Ma00] Y. I. Manin, *Classical computing, quantum computing, and Shor's factoring algorithm*, Séminaire Bourbaki exp. 862, Astérisque **266** (2000), 375-404, [`numdam:SB_1998-1999__41__375_0`], [`arXiv:quant-ph/9903008`].

[Ma22] P. Marra, *Majorana nanowires for topological quantum computation: A tutorial*, J. of Applied Physics **132** (2022) 231101, [`doi:10.1063/5.0102999`], [`arXiv:2206.14828`].

[ML71] P. Martin-Löf, *A Theory of Types*, unpublished note (1971), [`ncatlab.org/nlab/files/MartinLoef1971-ATheoryOfTypes.pdf`]

[ML75] P. Martin-Löf, *An intuitionistic theory of types: predicative part*, in: *Logic Colloquium '73*, Studies in Logic and the Foundations of Mathematics **80** (1975), 73-118, [`doi:10.1016/S0049-237X(08)71945-1`].

[ML82] P. Martin-Löf, *Constructive Mathematics and Computer Programming*, Studies in Logic and the Foundations of Mathematics **104** (1982), 153-175, [`doi:10.1016/S0049-237X(09)70189-2`].

[ML84] P. Martin-Löf (notes by G. Sambin of a series of lectures given in Padua in 1980), *Intuitionistic type theory*, Bibliopolis, Naples (1984), [`ncatlab.org/nlab/files/MartinLofIntuitionisticTypeTheory.pdf`]

[ML96] P. Martin-Löf, *On the Meanings of the Logical Constants and the Justifications of the Logical Laws*, Nordic J. Philosophical Logic **1** (1996), 11-60, [`docenti.lett.unisi.it/files/4/1/1/6/martinlof4.pdf`]

[MMN23] Y. Masaki, T. Mizushima, and M. Nitta, *Non-Abelian Anyons and Non-Abelian Vortices in Topological Superconductors*, [`arXiv:2301.11614`].

[MM10] B. Mashayekhy and H. Mirebrahimi, *Some Properties of Finitely Presented Groups with Topological Viewpoints*, Int. J. Math., Game Theory, and Algebra **18** 6 (2010), 511-515, [`arXiv:1012.1744`].

[MFL22] E. S. Matekole, Y.-L L. Fang, and M. Lin, *Methods and Results for Quantum Optimal Pulse Control on Superconducting Qubit Systems*, 2022 IEEE International Parallel and Distributed Processing Symposium Workshops (2022), [`doi:10.1109/IPDPSW55747.2022.00102`], [`arXiv:2202.03260`].

[MPSS19] T. Mawson, T. Petersen, J. Slingerland, and T. Simula, *Braiding and fusion of non-Abelian vortex anyons*, Phys. Rev. Lett. **123** (2019) 140404, [`doi:10.1103/PhysRevLett.123.140404`], [`arXiv:1805.10009`].

[May72] P. May, *The geometry of iterated loop spaces*, Springer, Berlin (1972), [`doi:10.1007/BFb0067491`].

[May77] P. May, *Infinite loop space theory*, Bull. Amer. Math. Soc. **83** 4 (1977), 456-494, [`doi:10.1090/S0002-9904-1977-14318-8`], [`http://www.math.uchicago.edu/~may/PAPERS/18.pdf`]

[MS06] P. May and J. Sigurdsson, *Parametrized Homotopy Theory*, Math. Surv. Monogr. **132** Amer. Math. Soc. (2006), [`ISBN:9781470413590`], [`arXiv:math/0411656`].

[Mi72] W. Miller, *Symmetry Groups and Their Applications*, Pure and Applied Mathematics **50** (1972), pp. 16-60, Elsevier, [`ISBN:9780080873657`].

[Mill19] H. Miller (ed.), *Handbook of Homotopy Theory*, Chapman and Hall/CRC Press (2019), [`ISBN:9780815369707`], [`doi:10.1201/9781351251624`].

[Miln80] J. Milne, *Étale Cohomology*, Mathematical Series **33**, Princeton University Press (1980), [`ISBN:9780691082387`], [`jstor:j.ctt1bpmbk1`].

[Mo03] C. Mochon, *Anyons from non-solvable finite groups are sufficient for universal quantum computation*, Phys. Rev. A **67** (2003) 022315, [`doi:10.1103/PhysRevA.67.022315`], [`arXiv:quant-ph/0206128`].

[Mo04] C. Mochon, *Anyon computers with smaller groups*, Phys. Rev. A **69** (2004) 032306, [`arXiv:quant-ph/0306063`], [`doi:10.1103/PhysRevA.69.032306`].

[Mø11] J. Møller, *The fundamental group and covering spaces*, lecture notes, [`arXiv:1106.5650`].

[Mu13] J. Munkres, *Topology*, Pearson (2013), [`ISBN:1292023627`].

[Mu10] F. Muro, *Representability of Cohomology Theories*, talk at Joint Math. Conf, CSASC, Prague (2010), [`ncatlab.org/nlab/files/Muro-Representability.pdf`]

[Mu22] Z. Murray, *Constructive Analysis in the Agda Proof Assistant*, [`arXiv:2205.08354`], [`github.com/z-murray/honours-project-constructive-analysis-in-agda`].

[My21] D. J. Myers, *Modal Fracture of Higher Groups*, Diff. Geom. Appl. (2024, in print) [`arXiv:2106.15390`].

[My22] D. J. Myers, *Orbifolds as microlinear types in synthetic differential cohesive homotopy type theory*, [`arXiv:2205.15887`].

[MR23] D. J. Myers and M. Riley, *Commuting cohesions*, [`arXiv:2301.13780`].

[NSSFD08] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, *Non-Abelian Anyons and Topological Quantum Computation*, Rev. Mod. Phys. **80** (2008), 1083-1159, [`doi:10.1103/RevModPhys.80.1083`], [`arXiv:0707.1889`].

[Ne80] G. Nenciu, *On the adiabatic theorem of quantum mechanics*, J. Phys. A: Math. Gen. **13** (1980) L15, [`doi:10.1088/0305-4470/13/2/002`].

[NC10] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, (2010), [`ISBN:9780511976667`].

[NDGD06] M. A. Nielsen, M. R. Dowling, M. Gu, and A. C. Doherty, *Quantum Computation as Geometry*, Science **311** 5764 (2006), 1133-1135, [`doi:10.1126/science.1121541`], [`arXiv:quant-ph/0603161`].

[NSS12a] T. Nikolaus, U. Schreiber, and D. Stevenson, *Principal ∞-bundles – General theory*, J. Homotopy Rel. Struc. **10** (2015), 749–801, [`doi:10.1007/s40062-014-0083-6`], [`arXiv:1207.0248`].

[NPS90] B. Nordström, K. Petersson, and J. M. Smith, *Programming in Martin-Löf's Type Theory*, Oxford University Press (1990), [`www.cse.chalmers.se/research/group/logic/book`].

[No09] U. Norell, *Dependently Typed Programming in Agda*, in *Advanced Functional Programming AFP 2008*, Lecture Notes in Computer Science 5832 (2009), 230-266, [`doi:10.1007/978-3-642-04652-0_5`].

[OP99] R. W. Ogburn and J. Preskill, *Topological Quantum Computation*, In: Williams, C.P. (Ed.), Quantum Computing and Quantum Communications, Lect. Notes Comput. Sci. **1509**, 341, Springer, Berlin (1999), [`doi:10.1007/3-540-49208-9-31`].

[Pa12] J. K. Pachos, *Introduction to Topological Quantum Computation*, Cambridge Univ. Press (2012), [`doi:10.1017/CBO9780511792908`].

[Pa98] E. Palmgren, *On Universes in Type Theory*, in: *Twenty-Five Years of Constructive Type Theory*, Oxford Univ. Press (1998), 191–204, [`doi:10.1093/oso/9780198501275.003.0012`].

[PM93] C. Paulin-Mohring, *Inductive definitions in the system Coq – Rules and Properties*, in: *Typed Lambda Calculi and Applications* TLCA 1993, Lecture Notes in Computer Science **664** Springer (1993), [`doi:10.1007/BFb0037116`].

[Pav91] D. Pavlović, *Categorical interpolation: Descent and the Beck-Chevalley condition without direct images*, in: Category Theory, Lecture Notes in Mathematics **1488**, Springer, Berlin (1991), [`doi:10.1007/BFb0084229`], [`isg.rhul.ac.uk/dusko/papers/1990-BCDE-Como.pdf`]

[Po98] A. Polyakov, *String Theory and Quark Confinement*, Nucl. Phys. Proc. Suppl. **68** (1998), 1-8, [`doi:10.1016/S0920-5632(98)00135-2`], [`arXiv:hep-th/9711002`].

[Po99] A. Polyakov, *The wall of the cave*, Int. J. Mod. Phys. A **14** (1999), 645-658, [`arXiv:hep-th/9809057`], [`doi:10.1142/S0217751X99000324`].

[Po02] A. Polyakov, *Gauge Fields and Space-Time*, Int. J. Mod. Phys. A **17** S1 (2002), 119-136, [`arXiv:hep-th/0110196`], [`doi:10.1142/S0217751X02013071`].

[PE97] P. Potts and A. Edalat, *Exact real computer arithmetic*, preprint (1997), [`ncatlab.org/nlab/files/PottsEdalat-ExactRealComputerArithmetic.pdf`]

[Pr18] J. Preskill, *Quantum Computing in the NISQ era and beyond*, Quantum **2** (2018) 79, [`arXiv:1801.00862`], [`doi:10.22331/q-2018-08-06-79`].

[Pr22] J. Preskill, *The Physics of Quantum Information*, talk at *The Physics of Quantum Information*, 28th Solvay Conference on Physics (2022), [`arXiv:2208.08064`].

[Qu67] D. Quillen, *Homotopical Algebra*, Lecture Notes in Mathematics 43 Springer, (1967), [`doi:10.1007/BFb0097438`].

[RSDC22] A. Rajak, S. Suzuki, A. Dutta, and B. K. Chakrabarti, *Quantum Annealing: An Overview*, Phil. Trans. Royal Soc. A **381** (2022) 20210417, [`doi:10.1098/rsta.2021.0417`], [`arXiv:2207.01827`].

[Rao16] S. Rao, *Introduction to abelian and non-abelian anyons*, In: *Topology and Condensed Matter* Phys. Texts & Read. in Phys. Sci. **19** Springer (2017), 399-437, [`doi:10.1007/978-981-10-6841-6_16`], [`arXiv:1610.09260`].

[Re22] J. M. Renes, *Quantum Information Theory*, De Gruyter, Basel (2022), [`doi:10.1515/9783110570250`].

[Re10] C. Rezk, *Toposes and homotopy toposes*, lecture notes 2010, [`ncatlab.org/nlab/files/Rezk_HomotopyToposes.pdf`]

[RZ16] M. Rho and I. Zahed (eds.), *The Multifaceted Skyrmion*, World Scientific, Singapore (2016), [`doi:10.1142/9710`].

[Ri20] B. Richter, *From categories to homotopy theory*, Cambridge Studies in Advanced Mathematics **188**, Cambridge University Press (2020), [`doi:10.1017/9781108855891`].

[RP11] E. Rieffel and W. Polak, *Quantum Computing – A gentle introduction*, MIT Press, Boston (2011), [`ISBN:9780262526678`].

[Ri14] E. Riehl, *Categorical Homotopy Theory*, Cambridge University Press (2014), [`ISBN:9781107048454`].

[Ri22] E. Riehl, *On the ∞-topos semantics of homotopy type theory*, lecture at *Logic and higher structures*, CIRM (Feb. 2022), [`emilyriehl.github.io/files/semantics.pdf`]

[RO12] G. Rigolin and G. Ortiz, *The Adiabatic Theorem for Quantum Systems with Spectral Degeneracy*, Phys. Rev. A **85** (2012) 062111, [`doi:10.1103/PhysRevA.85.062111`], [`arXiv:1111.5333`].

[Ri18] E. Rijke, *Introduction to Homotopy Type Theory*, lecture at CMU (2018) [`www.andrew.cmu.edu/user/erijke/hott`] [`ncatlab.org/nlab/files/Rijke-IntroductionHoTT-2018.pdf`]

[Ri19] E. Rijke, *Classifying Types*, PhD Thesis, Carnegie Mellon University, [`arXiv:1906.09435`].

[Ri23] E. Rijke, *Introduction to Homotopy Type Theory*, Cambridge University Press (in print), [`arXiv:2212.11082`].

[RSS20] E. Rijke, M. Shulman, and B. Spitters, *Modalities in homotopy type theory*, Log. Meth. Comp. Sci. **16** (2020) 1, [`doi:10.23638/LMCS-16(1:2)2020`], [`arXiv:1706.07526`].

[RS15] E. Rijke and B. Spitters, *Sets in homotopy type theory*, Math. Struc. Comp. Sci. **25** 5 (2015), 1172-1202, [`doi:10.1017/S0960129514000553`], [`arXiv:1305.3835`].

[Ri22] M. Riley, *A Bunched Homotopy Type Theory for Synthetic Stable Homotopy Theory*, PhD Thesis, Wesleyan University, (2022), [`doi:10.14418/wes01.3.139`].

[RPYLG21] T. Ringer, R. Porter, N. Yazdani, J. Leo, and D. Grossman, *Proof Repair across Type Equivalences*, [`arXiv:2010.00774`].

[Ro22] C. D. Roberts, *Origin of the Proton Mass*, [`arXiv:2211.09905`].

[Ro21] C. D. Roberts, *On Mass and Matter*, AAPPS Bulletin **31** (2021) 6, [`doi:10.1007/s43673-021-00005-4`], [`arXiv:2101.08340`].

[RS20] C. D. Roberts and S. M. Schmidt, *Reflections upon the Emergence of Hadronic Mass*, Eur. Phys. J. Special Topics **229** (2020), 3319–3340, [`doi:10.1140/epjst/e2020-000064-6`], [`arXiv:2006.08782`].

[Ro95] J. J. Rotman, *An Introduction to the Theory of Groups*, Springer, Berlin (1995), [`doi:10.1007/978-1-4612-4176-8`].

[Ro88] J. J. Rotman, *An Introduction to Algebraic Topology*, Graduate Texts in Mathematics **119**, Springer, Berlin (1988), [`doi:10.1007/978-1-4612-4576-6`].

[Ro22] E. Rowell, *Braids, Motions and Topological Quantum Computing*, [`arXiv:2208.11762`].

[RW18] E. C. Rowell and Z. Wang, *Mathematics of Topological Quantum Computing*, Bull. Amer. Math. Soc. **55** (2018), 183-238, [`doi:10.1090/bull/1605`], [`arXiv:1705.06206`].

[Ru64] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill (1976), [`ISBN13:9780070542358`].

[RS17] G. Rudolph and M. Schmidt, *Differential Geometry and Mathematical Physics – Part II. Fibre Bundles, Topology and Gauge Fields*, Springer, Berlin (2017), [`doi:10.1007/978-94-024-0959-8`].

[RW1910] B. Russell and A. Whitehead, *Principia Mathematica*, Cambridge University Press (1910, 1927), [`ISBN:9780521067911`].

[Sak20] K. Sakaguchi, *Validating Mathematical Structures*, in: *Automated Reasoning. IJCAR 2020*, Lecture Notes in Computer Science **12167** Springer (2020), [`doi:10.1007/978-3-030-51054-1_8`].

[San11] A. Santini, *Topological groupoids*, Kandidatproject, Copenhagen University (2011), [`ncatlab.org/nlab/files/Santini-Groupoids.pdf`]

[Sa23] H. Sati, *M-theory and matter via twisted equivariant differential (TED) K-theory*, talk at *M-Theory and Mathematics 2023*, CQTS @ NYU Abu Dhabi (2023).

[SS22-Conf] H. Sati and U. Schreiber, *Differential Cohomotopy implies intersecting brane observables via configuration spaces and chord diagrams*, Adv. Theor. Math. Physics **26** (2022) 4, [ISSN:1095-0753], [arXiv:1912.10425].

[SS20-Tad] H. Sati and U. Schreiber, *Equivariant Cohomotopy implies orientifold tadpole cancellation*, J. Geom. Phys. **156** (2020), 103775, [doi:10.1016/j.geomphys.2020.103775], [arXiv:1909.12277].

[SS20-Orb] H. Sati and U. Schreiber, *Proper Orbifold Cohomology*, [arXiv:2008.01101].

[SS21-MF] H. Sati and U. Schreiber, *M/F-Theory as Mf-Theory*, Rev. Math. Phys. **35** 10 (2023) [arXiv:2103.01877][doi:10.1142/S0129055X23500289].

[SS21-Bun] H. Sati and U. Schreiber, *Equivariant principal ∞-bundles*, [arXiv:2112.13654].

[SS22-Def] H. Sati and U. Schreiber, *Anyonic defect branes in TED-K-theory*, Rev. Math. Phys. (2023), [doi:10.1142/S0129055X23500095], [arXiv:2203.11838].

[SS22-Ord] H. Sati and U. Schreiber, *Anyonic topological order in TED-K-theory*, Rev. Math. Phys. **35** 03 (2023) 2350001, [doi:10.1142/S0129055X23500010], [arXiv:2206.13563].

[SS22-TQC] H. Sati and U. Schreiber, *Topological Quantum Programming in* TED-K, PlanQC **2022** 33 (2022), [arXiv:2209.08331], [ncatlab.org/schreiber/show/Topological+Quantum+Programming+in+TED-K].

[SS23-EoS] H. Sati and U. Schreiber, *Entanglement of Sections: The pushout of entangled and parameterized quantum information* [arXiv:2309.07245].

[SS23-QM] H. Sati and U. Schreiber, *The Quantum Monadology*, [arXiv:2310.15735].

[SSS12] H. Sati, U. Schreiber and J. Stasheff, *Twisted differential string and fivebrane structures*, Comm. Math. Phys. **315** (2012), 169-213, [doi:10.1007/s00220-012-1510-3], [arXiv:0910.4001].

[Sau17] J. Sau, *A Roadmap for a Scalable Topological Quantum Computer*, Physics **10** (2017) 68, [physics.aps.org/articles/v10/68].

[Sc13] U. Schreiber, *Differential cohomology in a cohesive ∞-topos*, Habilitation thesis, [arXiv:1310.7930].

[Sc14a] U. Schreiber, *Quantization via Linear Homotopy Types*, talk notes, Paris Diderot and ESI Vienna (2014), [arXiv:1402.7041].

[Sc14b] U. Schreiber, *Differential generalized cohomology in Cohesive homotopy type theory*, talk at *Formalization of Mathematics*, Inst. H. Poincaré, Paris (May 2014), [ncatlab.org/schreiber/files/SchreiberParis2014.pdf]

[Sc15] U. Schreiber, *Some thoughts on the future of modal homotopy type theory*, talk at German Mathematical Society meeting, Hamburg (Sept 2015), [ncatlab.org/schreiber/files/SchreiberDMV2015b.pdf]

[Sc22a] U. Schreiber, *Knots for quantum computation from defect branes*, talk at *Workshop on Topological Methods in Mathematical Physics*, Erice (Sep 2022), [ncatlab.org/schreiber/show/Knots+for+quantum+computation+from+defect+branes].

[Sc22b] U. Schreiber, *Quantum types via Linear Homotopy Type Theory*, talk at *Workshop on Quantum Software @ QTML2022*, Naples (Nov 2022), [ncatlab.org/schreiber/files/QuantumDataInLHoTT-221117.pdf]

[Sc23] U. Schreiber, *Topological Quantum Gates from M-Theory*, talk at *M-Theory and Mathematics 2023*, CQTS @ NYU Abu Dhabi (2023).

[ScSh14] U. Schreiber and M. Shulman, *Quantum Gauge Field Theory in Cohesive Homotopy Type Theory*, EPTCS **158** (2014), 109-126, [doi:10.4204/EPTCS.158.8], [arXiv:1408.0054].

[Schu72] H. Schubert, *Categories*, Springer, Berlin (1972), [doi:10.1007/978-3-642-65364-3].

[Schw96] J. Schwarz, *The Second Superstring Revolution*, lecture at Sakharov Conference (Moscow, May 1996), [inspire:969846], [arXiv:hep-th/9607067].

[Sc70] D. S. Scott, *Outline of a mathematical theory of computation*, in: Proc. 4th Ann. Princeton Conf. on Information Sciences and Systems (1970), 169–176, [ncatlab.org/nlab/files/Scott-TheoryOfComputation.pdf]

[ScSt71] D. S. Scott and C. Strachey, *Toward a Mathematical Semantics for Computer Languages*, Oxford Univ. Computing Laboratory, Technical Monograph PRG-6 (1971), [www.cs.ox.ac.uk/files/3228/PRG06.pdf]

[See84] R. A. G. Seely, *Locally cartesian closed categories and type theory*, Math. Proc. Camb. Phil. Soc. **95** (1984), 33-48, [doi:10.1017/S0305004100061284].

[Se73] G. Segal, *Configuration-spaces and iterated loop-spaces*, Invent. Math. **21** (1973), 213-221, [doi:10.1007/BF01390197].

[Se04] P. Selinger, *Towards a quantum programming language*, Math. Struc. Comp. Sci. **14** (2004), 527–586, [doi:10.1017/S0960129504004256].

[Sh94] P. W. Shor, *Algorithms for quantum computation: discrete logarithms and factoring*, Proc. 35th Annual Symposium on Foundations of Computer Science (1994), 124-134, [doi:10.1109/SFCS.1994.365700].

[Sh95] P. W. Shor, *Scheme for reducing decoherence in quantum computer memory*, Phys. Rev. A **52** (1995) R2493(R), [`doi:10.1103/PhysRevA.52.R2493`].

[Sh12] M. Shulman, *Minicourse on Homotopy Type Theory*, University of Swansea (2012), [`http://home.sandiego.edu/~shulman/hottminicourse2012`].

[Sh15] M. Shulman, *Univalence for inverse diagrams and homotopy canonicity*, Math. Struc. Comp. Sci. **25** 5 (2015), 1203-1277, [`doi:10.1017/S0960129514000565`], [`arXiv:1203.3253`].

[Sh18] M. Shulman, *Brouwer's fixed-point theorem in real-cohesive homotopy type theory*, Math. Struc. Comp. Sci **28** (6) (2018), 856-941, [`doi:10.1017/S0960129517000147`], [`arXiv:1509.07584`].

[Sh19] M. Shulman, *All $(\infty, 1)$-toposes have strict univalent universes*, [`arXiv:1904.07004`].

[Si83] B. Simon, *Holonomy, the Quantum Adiabatic Theorem, and Berry's Phase*, Phys. Rev. Lett. **51** (1983), 2167-2170, [`doi:10.1103/PhysRevLett.51.2167`].

[Si97] D. R. Simon, *On the power of quantum computation*, SIAM Journal on Computing **26** 5 (1997), [`doi:10.1137/S0097539796298637`].

[Si21] S. H. Simon, *Topological Quantum*, lecture notes and proto-book (2021), [`www-thphys.physics.ox.ac.uk/people/SteveSimon/topological2021/TopoBook-Sep1-2021.pdf`]

[Si99] C. Simpson, *A Giraud-type characterization of the simplicial categories associated to closed model categories as ∞-pretopoi*, [`arXiv:math/9903167`].

[SK95] K. Slonneger and B. Kurtz, *Denotational semantics*, *Formal Syntax and Semantics of Programming Languages*, Addison-Wesley (1995), [`https://homepage.divms.uiowa.edu/~slonnegr/plf/Book/`].

[So00] R. Solovay, *Lie Groups and Quantum Circuits*, talk in: Mathematics Of Quantum Computation workshop, MSRI (2000), [`msri.org/workshops/189/schedules/12826`].

[SU06] M. H. Sorensen and P. Urzyczyn, *Lectures on the Curry-Howard isomorphism*, Studies in Logic **149**, Elsevier (2006), [`ISBN:9780444520777`].

[Sp66] E. Spanier, *Algebraic topology*, McGraw Hill (1966), Springer (1982), [`doi:10.1007/978-1-4684-9322-1`].

[St20] T. D. Stanescu, *Introduction to Topological Quantum Matter & Quantum Computation*, CRC Press (2020), [`ISBN:9780367574116`].

[St43] N. Steenrod, *Homology With Local Coefficients*, Ann. Math., Sec.Ser. **44** (1943), 610-627, [`jstor:1969099`].

[SL13] A. Stern and N. H. Lindner, *Topological Quantum Computation – From Basic Concepts to First Experiments*, Science **339** 6124 (2013), 1179-1184, [`doi:10.1126/science.1231473`].

[St93] T. Streicher, *Investigations into Intensional Type Theory*, Habilitation Thesis, Darmstadt (1993), [`ncatlab.org/nlab/files/Streicher-IntensionalTT.pdf`]

[Str13] F. Strocchi, *An Introduction to Non-Perturbative Foundations of Quantum Field Theory*, Oxford University Press (2013), [`doi:10.1093/acprof:oso/9780199671571.001.0001`].

[Str11] J. Strom, *Modern classical homotopy theory*, Graduate Studies in Mathematics **127**, Amer. Math. Soc., Providence, RI (2011), [`doi:10.1090/gsm/127`].

[St16] A. Stump, *Verified Functional Programming in Agda*, Association for Computing Machinery and Morgan & Claypool (2016), [`doi:10.1145/2841316`].

[Th91] S. Thompson, *Type Theory and Functional Programming*, Addison-Wesley (1991), [`ISBN:0201416670`].

[TH01] I. Todorov and L. Hadjiivanov, *Monodromy Representations of the Braid Group*, Phys. Atom. Nucl. **64** (2001), 2059-2068, [`doi:10.1134/1.1432899`], [`arXiv:hep-th/0012099`].

[TV] B. Toën and G. Vezzosi, *Homotopical Algebraic Geometry I: Topos theory*, Adv. Math. **193** 2 (2005), 257-372, [`doi:10.1016/j.aim.2004.05.004`], [`arXiv:math/0207028`].

[tD87] T. tom Dieck, *Transformation Groups*, de Gruyter (1987), [`doi:10.1515/9783110858372`].

[tD08] T. tom Dieck, *Algebraic topology*, Eur. Math. Soc. (2008), [`doi:10.4171/048`].

[Tr69] A. S. Troelstra, *Principles of Intuitionism*, Lecture Notes in Mathematics **95** Springer, Berlin (1969), [`doi:10.1007/BFb0080643`].

[Tr77] A. S. Troelstra, *Aspects of Constructive Mathematics*, Studies in Logic and the Foundations of Mathematics **90** (1977), 973-1052, [`doi:10.1016/S0049-237X(08)71127-3`].

[TvD88] A. S. Troelstra and D. van Dalen, *Constructivism in Mathematics – An introduction*, Vol 1, Studies in Logic and the Foundations of Mathematics **121**, North Holland (1988), [`ISBN:9780444702661`].

[Ts17] D. Tsementzis, *Univalent foundations as structuralist foundations*, Synthese **194** 9 (2017), 3583–3617, [`doi:10.1007/s11229-016-1109-x`].

[UniMath] UniMath, [`unimath.github.io/UniMath`], [`unimath.github.io/agda-unimath`].

[UFP13] Univalent Foundations Project, *Homotopy Type Theory – Univalent Foundations of Mathematics*, Institute for Advanced Study, Princeton, 2013, [`homotopytypetheory.org/book`].

[Va21] S. J. Valera, *Fusion Structure from Exchange Symmetry in* $(2+1)$-*Dimensions*, Ann. Phys. **429** (2021) 168471, [`doi:10.1016/j.aop.2021.168471`], [`arXiv:2004.06282`].

[vdBG11] B. van den Berg and R. Garner, *Types are weak ω-groupoids*, Proc. of the London Math. Soc. **102** 2 (2011) 370-394, [`doi:10.1112/plms/pdq026`], [`arXiv:0812.0298`].

[Va18] D. Vanderbilt, *Berry Phases in Electronic Structure Theory – Electric Polarization, Orbital Magnetization and Topological Insulators*, Cambridge University Press (2018), [`doi:10.1017/9781316662205`].

[vD18] F. van Doorn, *On the Formalization of Higher Inductive Types and Synthetic Homotopy Theory*, PhD dissertation, Carnegie Mellon (2018), [`arXiv:1808.10690`].

[VW21] N. Veltri and N. van der Weide, *Constructing Higher Inductive Types as Groupoid Quotients*, Logical Methods in Computer Science **17** 2 (2021), [`doi:10.23638/LMCS-17(2:8)2021`], [`arXiv:2002.08150`].

[VMA19] A. Vezzosi, A. Mörtberg, and A. Abel, *Cubical Agda: A Dependently Typed Programming Language with Univalence and Higher Inductive Types*, Proceedings of the ACM on Programming Languages **3** ICFP 87 (2019), 1–29, [`doi:10.1145/3341691`].

[Vo10] V. Voevodsky, *Univalent Foundations Project*, grant proposal application (2010), [`ncatlab.org/nlab/files/Voevodsky-UFP2010.pdf`]

[Vo03I] C. Voisin, *Hodge theory and Complex algebraic geometry I*, translated by L. Schneps, Cambridge University Press (2002/3), [`doi:10.1017/CBO9780511615344`].

[Vo03II] C. Voisin, *Hodge theory and Complex algebraic geometry II*, translated by L. Schneps, Cambridge University Press (2002/3), [`doi:10.1017/CBO9780511615177`].

[vN32] J. von Neumann, *Mathematische Grundlagen der Quantenmechanik*, Springer (1932, 1971), [`doi:10.1007/978-3-642-96048-2`]; translated as: *Mathematical Foundations of Quantum Mechanics* Princeton University Press (1955), [`doi:10.1515/9781400889921`].

[Vui88] J. Vuillemin, *Exact real computer arithmetic with continued fractions*, in *LFP '88: Proceedings of the 1988 ACM conference on LISP and functional programming* (1988), pp. 14-27, [`doi:10.1145/62678.62681`].

[Wa15] P. Wadler, *Propositions as Types*, Comm. ACM **58** 12 (2015), 75–84, [`doi:10.1145/2699407`].

[Wa10] Z. Wang, *Topological Quantum Computation*, CBMS Regional Conference Series in Mathematics **112** Amer. Math. Soc. (2010), [`ISBN-13:978082184930-9`].

[Wä] D. Wärn, *Eilenberg-MacLane spaces and stabilisation in homotopy type theory*, [`arXiv:2301.03685`].

[Wei96] A. Weinstein, *Groupoids: Unifying Internal and External Symmetry – A Tour through some Examples*, Notices Amer. Math. Soc. **43** (1996), 744-752, [`www.ams.org/notices/199607/weinstein.pdf`]

[We17] F. Wellen, *Cartan Geometry in Modal Homotopy Type Theory*, PhD Thesis, KIT 2017, [`arXiv:1806.05966`], [`ncatlab.org/schreiber/show/thesis+Wellen`].

[We91] X.-G. Wen, *Topological orders and Chern-Simons theory in strongly correlated quantum liquid*, Int. J. Mod. Phys. B **05** 10 (1991), 1641-1648, [`doi:10.1142/S0217979291001541`].

[WZ84] F. Wilczek and A. Zee, *Appearance of gauge structure in simple dynamical systems*, Phys. Rev. Lett. **52** 24 (1984), 2111-2114, [`doi:10.1103/PhysRevLett.52.2111`].

[Will20] L. Williams, *Configuration Spaces for the Working Undergraduate*, Rose-Hulman Undergrad. Math. J. **21** (2020) 8, [`rhumj:vol21/iss1/8`], [`arXiv:1911.11186`].

[Wils18] J. C. H. Wilson, *The geometry and topology of braid groups*, lecture at *2018 Summer School on Geometry and Topology*, Chicago (2018), [`ncatlab.org/nlab/files/Wilson-BraidGroups.pdf`]

[YG91] C. N. Yang and M. L. Ge (eds.), *Braid Group, Knot Theory and Statistical Mechanics*, Adv. Ser. Mat. Phys. **9**, World Scientific, Singapore (1991), [`doi:10.1142/0796`].

[YD95] C.-K. Yap and T. Dubé, *The exact computation paradigm*, in: *Computing in Euclidean Geometry*, Lecture Notes Series on Computing, World Scientific (1995), 452-492, [`doi:10.1142/9789812831699_0011`].

[ZLSS15] J. Zaanen, Y. Liu, Y.-W. Sun, and K. Schalm, *Holographic Duality in Condensed Matter Physics*, Cambridge Univ. Press (2015), [`doi:10.1017/CBO9781139942492`].

[ZR99] P. Zanardi and M. Rasetti, *Holonomic Quantum Computation*, Phys. Lett. A **264** (1999), 94-99, [`doi:10.1016/S0375-9601(99)00803-8`], [`arXiv:quant-ph/9904011`].

[Zu75] J. Zucker, *Formalization of Classical Mathematics in Automath*, Colloq. Internat. Cent. Nat. Rech. Scient. **249** (1975), 135-145, [`www.win.tue.nl/automath/archive/webversion/aut042/aut042.html`]; also in: *Studies in Logic and the Foundations of Mathematics* **133** (1994), 127-139, [`doi:10.1016/S0049-237X(08)70202-7`].

[ZW20] A. Zulehner and R. Wille, *Simulation and Design of Quantum Circuits*, in: *Reversible Computation: Extending Horizons of Computing. RC 2020*, Lecture Notes in Computer Science **12070**, Springer, New York (2020), [`doi:10.1007/978-3-030-47361-7_3`].