The Algorithmic-Device View of Informal Rigorous Mathematical Proof

Jody Azzouni

Contents

Abstract

A new approach to informal rigorous mathematical proof is offered. To this end, algorithmic devices are characterized and their central role in mathematical proof delineated. It is then shown how all the puzzling aspects of mathematical proof, including its peculiar capacity to convince its practitioners, are explained by algorithmic devices. Diagrammatic reasoning is also characterized in terms of algorithmic devices, and the algorithmic device view of mathematical proof is compared to alternative construals of informal proof to show its superiority.

J. Azzouni (✉)
Department of Philosophy, Tufts University, Medford, MA, USA
e-mail: Jody.Azzouni@tufts.edu

B. Sriraman (ed.), *Handbook of the History and Philosophy of Mathematical Practice*,
https://doi.org/10.1007/978-3-030-19071-2_4-1

Keywords

Informal mathematical proof, diagrams · Turing computability · Algorithms · Euclidean geometry · Venn diagrams · Formalization · Conceptual engineering · Games

1 Mathematics Is Strange

A mere couple thousand years ago, mathematics seemed innocent enough. (At first glance, anyway.) You sketched out some fairly simple diagrams, and you talked them up in ancient Greek. You said things like, "look over here," "look at that," and then you drew a conclusion or two (Plato 1963). Somehow, nevertheless, it all amounted to something like nothing else we do – like nothing else we do *on Earth.* The conclusions drawn had an exactness impossible even by careful measurements (interior angles of triangles were proved to be *exactly* 180° – nothing slightly more or less, which is what always happened if, instead, the angles of a triangle were measured directly)[1]; furthermore, the results shown this way seemed to be necessarily true (they *couldn't be* otherwise), and so the process ("reasoning") by which these results were achieved seemed independent of our usual tools for learning about things – our senses. Equally odd was that these conclusions were often unexpected and sometimes undesirable. Many of them were results no one saw coming; some of them were even results that no one *liked* after they arrived – for example, the famous (and scandalous) fact that the length of the diagonal of a triangle isn't a ratio of the lengths of the adjacent sides of that triangle . . . from which it can be shown that √2 isn't rational.[2] Call this *the epistemic strangeness of mathematical proof.*

Philosophy was born, perhaps, from the desire to explain the weird and bizarre – less melodramatically, from the need to explain *odd cases.* This used to be called *wonder*, a "what on Earth is going on *here*?" Shortening a long descriptive history of (very ingenious) explanations of the epistemology of mathematical proof, many of them *supernatural* – such as mathematical proof being our recollection of eternal entities that we perceived directly in a previous life, or as involving innate ideas that have been stamped conveniently into our souls by God, or proofs being sequences of thoughts true by virtue of their conceptual structure alone (syllogistic sequences, "analytic truths"), or proofs as involving the addition of content from "intuition" ("synthetic a priori truths"), or proofs being lists of sentences true by the meanings of the words within them, etc. – a post-mid-twentieth-century conceptual revisiting of the artificial ("formal") languages that were invented by Frege (1879) at the end of

[1] Also *very* striking was that these exact results arise from methods applied to diagrams – compass and straightedge – that themselves don't intrinsically involve exact measurements: no ruler is used, for example.

[2] A rumor passed down from ancient Greek times is that the person who discovered this was *murdered.*

the nineteenth century seemed to *finally* offer a conclusive explanation. Derivations in such *artificial* languages are effectively recognizable patterns of reasoning.[3]

That is (and here's one sort of explanation gestured at in the last sentence above), if the many mathematical proofs in natural languages just *are* (disguised) derivations in artificial languages – if they're abbreviations of such derivations, for example – then this can explain all the odd epistemic qualities mathematical proofs in natural languages have. It can explain, for example, the impression of necessity: Our senses play only as much of a role as they seem to with ancient Greek diagrams (only enough to enable us to manipulate the language entities involved, and nothing more), and the results aren't necessary or eternal, but only as necessarily true as the premises the proofs start from. If logical inference is to be characterized as formal derivation in these artificial languages, then the epistemic qualities of mathematical proof become purely matters of logic. In particular, the strange unchanging and perfect eternal entities that Plato used to explain the apparent properties of mathematical results (exactness and necessity) are no longer needed: If an argument that mathematical entities exist is to be given, it can no longer arise via an inference to the best explanation for the epistemic qualities of mathematical proof; instead, it must result from the requirement, when applying mathematics to empirical science, that one necessarily uses phrases like "there are numbers...," coupled with the linguistic claim that such a phrase commits us to the existence of numbers.[4]

Call a *derivational account of informal rigorous mathematical proof*, any account that takes the epistemic qualities of ordinary mathematical proof to derive from a relationship (of some sort) between ordinary mathematical proofs and formal derivations in one or another (or a family) of artificial languages.

[3]I'm condensing some intellectual history here. "Effective recognizability" as relevant to an explanation for the epistemic strangeness of ordinary mathematical proof emerged after the discovery of formal languages (it emerged, perhaps, even some *decades* after the Church/Turing/Kleene discovery of the mathematical characterization of effective recognizability). The possible explanation Fregean languages initially seemed to offer for epistemic strangeness was instead a development of a much earlier view (one already in, to some extent, Descartes, and certainly in Leibniz) that mathematics is just "logic" *once all the missing inferential steps are explicitly given ("filled in") in the corresponding derivations.* There still is an issue, of course, about the epistemology of logical inference: how challenging this issue is corresponds to how much apparent mathematical content has been pressed into the "logical" principles. In any case, the explanation doesn't work because it has the same problems faced by *any* explanation of the epistemic properties of ordinary mathematical proofs in terms of their relations to derivations in artificial formalisms (see Sect. 3 below).

[4]That is, I'm describing proponents of what have come to be called *indispensability arguments.* There is a large contemporary literature on these arguments. For the version of them that influenced their recent application to mathematics, see Quine (1953). See Azzouni (2009a) for an indispensability argument with weak premises that establishes only the truth of applied mathematics and not the existence of mathematical *objects*; see Panza and Sereni (2015) for a categorization of the whole family of indispensability arguments according to the strength of the premises used in those arguments, and for a survey of this literature as of that date.

2 Effective Recognition Procedures

Before discussing the drawbacks of derivational accounts of mathematical proof, I should first discuss in more detail why formal derivations of formal languages seemed and continue to seem so illuminating about (or, at least relevant to) the epistemic properties of informal rigorous mathematical proof. The key point is that formal derivations come with *effective syntactic recognition procedures.* That is, anyone (including computers or mechanical calculating devices) can acquire a set of rules by which she or he or it can determine after a finite amount of time, *whether or not* a sequence of strings of vocabulary items is a proof. In describing these recognition procedures as "syntactic," I'm pointing out that a person can execute such a procedure successfully without knowing what the sentences "mean"; indeed, such syntactic recognition procedures don't require knowledge that what's being recognized are "sentences" of a language. So far, I've only mentioned well-known truisms about formal languages. I now compare natural languages with formal languages, and how they differ on the recognition procedures they come with. What I say next isn't quite as truistic; some of it is even controversial.

Natural languages also have recognition procedures. What's required of *every* language to begin with (ones we can *use*, anyway – one way or another) is that there be recognition procedures that enable the sorting of vocabulary tokens (the "alphabet") into type-classes. Vocabulary types are sensorily accessible classes of physical items, for example, the type of items, "e," that include those that occur in the paper you're reading. These may be electronically composed of pixels or instead they may be chemical stains on a background medium. We idealize these type-classes of *physical* vocabulary items by assuming that there is a decision procedure (looking "by eye"[5]) that determines what type-class *any* candidate physical item belongs to. Implicitly excluded (call these *exclusion idealizations*), therefore, are all those many physical tokens that can't be decided by eye because (among other things), they're too blurry or badly shaped. The very coherence of physically executing *any* algorithm (by humans, computers. . .) *requires* exclusion idealizations.[6]

Both natural languages and many formal languages – not all formal ones – have recognition procedures for primitive vocabulary. Consider now the recognition procedures for syntactic classes of items other than primitive vocabulary items – in particular, for sentences. The recognition procedures available for these syntactic classes of items of formal languages go beyond those of natural languages in two

[5] "By eye" is metaphor; other human (or nonhuman) senses can be used instead of vision.

[6] This is true of physically real Turing machines: a written symbol in a cell that a Turing-machine head "scans" can't be so lightly printed that the Turing-machine head is unable to tell whether or not the cell is blank. At least when it comes to humans, these recognition procedures are extremely complex, neurophysiologically speaking. If one physically built such a thing, scanning could be designed to be mechanically quite simple, as it isn't for us. Even so, there would still be cases where the physical device failed to scan a cell successfully; this is a fact about any Turing machine that exists in space and time. Indeed, this is true of any actual calculating device whatsoever: there will always be "mishaps."

ways. First, although we *can* assume that those competent in natural languages have (and use) recognition procedures for the sentences of those languages, we can't state these procedures explicitly.[7] We *can* state in mathematically precise ways the recognition procedures for the sentences of many formal languages – in particular, Frege's original language, and the many formal languages currently studied, including the first-order predicate calculus.[8] We can describe and mathematically study, of course, many formal languages – some with infinitely long sentences, but not those alone – that don't have recognition procedures for their sentences or even, say, for their primitive vocabulary (because the set of types of primitive vocabulary in a formal language can be infinite too).

Many formal languages have a second property that's often treated as a constitutive element of those languages. They can have syntactic-transition rules from sentences (of such-and-such syntactic forms) to sentences (of so-and-so forms), as well as *axioms*.[9] Consider the set of finite sequences of sentences of a formal language, where each sentence is either an *axiom* or follows from earlier sentences in the sequence by the application of a syntactic-transition rule to those earlier sentences. Call this a *derivation.* If there is a recognition procedure for the syntactic-transition rules, and for sentences, then there is a recognition procedure for the set of derivations. Call an *interpretation* of a formal language a function that takes all the sentences of a language to truth values (true or false), where the axioms are mapped to true, and so that the syntactic-transition rules take true sentences to true sentences.[10] A formal language (individuated to include axioms and syntactic-transition rules) is *consistent* if it has an interpretation; the foregoing stipulates soundness as a requirement of an interpretation.

The formal languages I've just finished describing are special insofar as the derivations in them have been characterized purely syntactically. This isn't the only possibility. The "semantics" of such languages – what the sentences mean – has been restricted to truth values that are assigned to sentences; but of course richer semantic characterizations of the sentences (and syntactic characterizations of other items of formal languages – predicates and names for example) can be given. The now-traditional way of doing so for classical languages is to supply a Tarskian model for a formal language, in which a domain of objects is assigned as the ranges of the

[7]It seems, and this is a typical assumption of Chomsky-influenced linguists, that competence in a natural language simply *is* the possession of a recognition procedure for its grammatically correct sentences. What such recognition procedures look like is a contested empirical question; it's clear, in any case, that we neither have conscious access to them – as individual users of natural languages (but only to, at best, some of their results when candidate sentences are presented to us) – nor do we, as a matter of linguistic science, know much about their properties (let alone what they are).

[8]Indeed, the grammars (syntax) of most of these languages are extremely simple – stated by a handful of easily understood rules.

[9]It's possible to drop axioms altogether; replace them with syntactic-transition rules from the empty set to sentences.

[10]I momentarily consider interpretations that are richer than the mere assignment of truth values to sentences – so this isn't a *definition* of "interpretation."

quantifiers, individuals from that domain are assigned to individual constants (names), and classes of *n*-tuples of objects from the domain are assigned to *n*-place predicates (Tarski 1933).[11] This isn't, of course, the only way to assign semantic properties to the sentences of formal languages. Regardless, semantic-transition rules can be formulated for formal languages (given one or another characterization of the semantic properties for the sentences of such a language), and a resulting notion of a derivation can depend on semantic properties of sentences instead of (or in addition to) the syntactic properties of such sentences. The resulting set of derivations will have a recognition procedure (or not) depending in part on whether there is a recognition procedure for the relevant semantic properties assigned to sentences.

It *seems*, as I mentioned, that semantic properties (of *some* sort) are intrinsically involved in informal rigorous mathematical proof. This empirical question is entangled in part with the broader empirical question of whether reasoning in natural languages is (purely) syntactic or not. Syllogisms are long-known examples of *bits* of natural-language reasoning that can be characterized syntactically. Nevertheless, it's not obvious that when nonprofessionals reason syllogistically that they are reasoning syntactically (i.e., independently of the perceived content of the sentences in such syllogisms).[12] It's also not obvious that reasoning doesn't occur purely syntactically in cases where it doesn't appear to: that depends on exactly what the grammar of natural languages turns out to be – what aspects of what (pretheoretically speaking) seem to be meaning properties of items of natural languages turn out to be such and which don't (anaphora, e.g., seems to be a syntactic matter in natural languages; so seem aspects of noun/verb agreement). In this respect, informal rigorous mathematical proof – inference steps in informal proofs – couched as they are in natural language, may also (or not) turn out to be based (at least some of the time) on meanings.

Regardless, we can now characterize the family of derivational accounts of informal rigorous mathematical proof with a little more precision. The idea is this: Informal rigorous mathematical proofs have the epistemic properties they have because there is a family of formal languages, where each informal proof is associated with a derivation or (possibly) a class of derivations (which may be from more than one formal language), and where the epistemic properties of informal rigorous mathematical proofs are explained by this association. For example (and, again, this crude example can't be right, but it illustrates the idea), any informal rigorous mathematical proof is an abbreviation of a particular derivation, and mathematicians (subconsciously) appreciate this fact.

---

[11]In nonclassical settings – for example, paraconsistent ones – generalizations of the Tarskian approach to model theory have been developed. So too, for intuitionism, multivalued logics, and other "logics."

[12]For reasons from the psychological literature to think otherwise, see Thompson et al. (2011).

3 Why Derivational Accounts of Mathematical Proof Don't Work

Mathematical proofs were expressed for centuries in natural languages – long before Frege's invention of artificial ones. And, despite the fertile infusion of formal tools into mathematics during the twentieth century – both as methods of proof but almost more importantly, as subject matters *of* mathematical study – mathematical practice continued and continues to occur in natural language.[13] We can ask why; and one straightforward answer places a heavy burden on derivational accounts of mathematical proof. This is that it's proofs, after all, in *natural languages* that convince mathematicians of the theorems that are established by these proofs. Furthermore, and this is an important datum too, it's *not* that mathematicians are convinced (finally) when these proofs are explicitly transliterated into their formal cousins. That, pretty much, never happens: "Oh wow, *now* I see why this is true." There would be no issue if the word "abbreviation" that I used in Sect. 1 was literal – that the version of the derivational account that worked was one where informal rigorous mathematical proofs were abbreviations of formal items – but if that were the case, then Frege's invention would not have been the original and amazingly creative invention it was.

It's important to realize that when I describe Frege's invention as that of an *artificial* language, I'm not exaggerating: he didn't (as a superficial glance might imply) lift out a couple of ordinary-language words or phrases, the German words or phrases for "and," "or," "there is," for example, along with a couple of syntactic rules of German and exhibit *them* as his invention. The syntax (and the semantics) of these artificial languages (i.e., what currently passes for "logic" in contemporary studies) is nothing like that of *any* natural language. Mathematical proof seems epistemically strange; but we can't explain that strangeness by treating mathematical practice as *actually occurring* in languages *that didn't exist* until (about) 1879. Explanations of mathematical practice have long trafficked in inferences to the best explanation of it being our sensitivity to supernatural entities of one sort of another – but perhaps this is really taking things too far.

[13]There is no book, no article, of mathematics that's entirely written in one or another formal language – not one in first-order logic, not one in any higher-order logic, not one in any of the numerous modal logics, etc. The proofs that occur in mathematical articles and books are rarely completely or even partially formalized; even in *logic* books and textbooks, completely formalized proofs occur only as *examples.* Frege, on the other hand, really intended his invention to replace then current mathematical *proof practices.* He saw ordinary language as flawed in a way that he took to create serious problems for mathematical proof. As the subsequent failure of mathematicians to adopt his formalism, or anyway, to switch to one or another formal language when doing mathematics shows, this is wrong. Exactly why it's wrong is one topic of this chapter.

Call *Phenomenology* the requirement that any derivational account explain why formal transcriptions of informal rigorous mathematical proofs don't ever have a property that the original informal rigorous mathematical proofs sometimes have: the property of inducing an experience of recognizing that this is a good proof – based (partially or sometimes entirely) on what the sentences of the proof *say.* It's important not to overstate the datum *Phenomenology* labels. The experience of (pretty much) any informal rigorous mathematical proof is that of *heterogeneous reasoning*: some inferential steps involve an "aha"-epiphany (Feferman 2012) that's one probable source of early modern philosophical views about the a priori; some, however, involve complex semantically obscure mechanical manipulations ("calculations") that are recognized to be valid manipulations because they follow rules that have been shown to be truth preserving (or that we've assumed are); some are taken on the authority of the author or as understood-to-be-background assumptions in the field; some involve significant and rich inferential moves (although this is often tacit, unexplored, and sometimes unrecognized); some are, strictly speaking, meta-proof considerations (e.g., "the other cases are the same"; "the indices on these operators are countable"). A new notion is required to enable the analysis of the heterogeneous nature of reasoning in ordinary rigorous mathematical proof – that of an *algorithmic device* (see Sect. 6 and later sections for discussion).[14]

Leaving the option of abbreviation aside, it wouldn't help if we instead hypothesized that natural-language proofs function successfully as epistemic guides for mathematicians by providing semantic or syntactic directions – "indications" of the corresponding formal proof – where such guidance is posited to be the psychological machinery (conscious or unconscious) by which mathematicians become convinced of theorems by the informal proofs they give. For, unfortunately for derivational accounts, nothing like this looks right either.[15] The initially slow evolution of specialized terminology in mathematics (the invention of place-holder Arabic numerals, algebraic notation, scientific notation, etc.), simultaneously accompanied with an evolving notion of "rigor," can give the impression that mathematical language slowly crawled over the eons toward an artificial-language endstate; Frege's "genius" was just to speed up the process by jump-starting the first of subsequently many such artificial languages in which the heretofore buried structure of mathematical proof could be finally fully exposed.

[14]A terminological conflict: Barwise (1993) uses the phrase "heterogeneous reasoning" to label formal-reasoning systems with both language-based inference patterns and diagrammatic ones. This usage has become influential in this literature. My usage (Azzouni 2017a, Sect. 9) describes something different: the character of typical informal rigorous mathematical proofs, as I describe them in the passage this footnote is appended to.

[15]What follows in the rest of this section is distilled from many discussions about mathematical proof over the past 20 years or so. Among the relevant papers and books are: Hersh (1997), Rav (1999), Azzouni (2006), Rav (2007), Azzouni (2009a, b), Lavor (2012), Tanswell (2015), and many others. Avigad (2019) and Hamami (2019), however, are recent attempts to defend derivational accounts. I discuss these a little in Sect. 15.

Several considerations, however, undercut this view of the terminological evolution of mathematics. There is, most importantly, that the phenomenology of mathematical proof – our experience of validity – doesn't accord with this picture. Mathematicians, pretty uniformly, learn to become convinced of the validity of mathematical proofs in ordinary language, by being trained to appreciate a certain degree of rigor in these proofs – *as they stand.* Such training does involve acquiring a facility with specialized terminology – a terminology that changes over time. Nevertheless, it's clear that the recognition of the validity of proofs often (but not always) involves the earlier mentioned "aha"-epiphany of recognizing that a certain result follows from others. This experience, as I've mentioned, *isn't* preserved when such proofs are transliterated into their purely formal versions but it does occur outside of mathematical practice; it's one we all feel with *certain* items of valid reasoning – for example, with (certain) instances of modus ponens, or when we see that from "All dogs are good pets," and "All good pets are creatures with large carbon footprints," it follows that "All dogs are creatures with large carbon footprints."[16]

That is, this "aha"-epiphany – among all of us – seems to *depend on the perceived meanings of the sentences in question and not on the syntactic properties of those sentences*; it doesn't occur when someone, applying rules, recognizes that a string of symbols follows by those rules from an earlier set of strings of symbols. This is why the formalization of an informal rigorous mathematical proof lacks the epistemic qualities that such proofs (sometimes) possess in ordinary language.[17] We no longer rely on our appreciation – whatever it comes to – of the meanings of the words and sentences we're using. We've switched to the mechanical manipulation of strings according to preassigned rules.

Phenomenology isn't the only challenging aspect of informal rigorous mathematical proof that derivational accounts face. Tanswell (2015, 297–298) requires of any derivational account that it explain (or at least be compatible with) five aspects of informal rigorous mathematical proof as it occurs in natural languages – what he calls *Rigor*, *Correctness*, *Agreement*, *Content*, and *Techniques*.

[16]This validity-experience, however, is frailer (at least in some cases) than many philosophers realize. It can be interfered with by our appreciation of the *content* of sentences – for example, by the conclusion striking us as implausible (see, e.g., Thompson et al. (2011), for studies that show this). I hypothesize that we need to consciously *learn* to distinguish intuitions of validity from intuitions of soundness – something that officially happens in classes on "critical thinking." The distinction, that is, isn't naturally made when we evaluate (in real time) arguments in day-to-day living: we're usually just interested in whether we should believe the conclusion on the basis of what we've heard.

[17]This will turn out to be an *overstatement.* More carefully: formalizations *sometimes* lack the epistemic qualities of the informal proofs they correspond to. They don't if they're *phenomenologically faithful.* See Sect. 13.

- *Rigor*: Informal mathematical proofs must meet conditions of rigor that (implicitly and explicitly) hold of informal proofs. This “rigor” seems different from the rigor exhibited by derivations (the latter is only that the derivation be fully written out according to formal rules). Notably, informal conditions on rigor change over time.
- *Correctness*: Mathematicians, perusing informal proofs, recognize them to be correct and incorrect intrinsically – “intrinsically” is understood here only in the sense that mathematicians don’t recognize these properties of proofs by comparing them to formal transliterations.
- *Agreement*: Mathematicians, upon perusing candidate informal rigorous proofs, largely agree with one another about whether the proofs are suitably rigorous and on whether these proofs are correct or incorrect. Strikingly, mathematicians from different proof-practice backgrounds (e.g., classical as opposed to intuitionistic ones), and who understand the respective proof practices, agree on this as well. As noted earlier, this doesn’t seem to occur by a process of recognizing the derivations (valid or invalid) that these proofs are associated with.
- *Content:* Any derivational explanation must explain how the perceived content of an informal rigorous mathematical proof – what the sentences of that proof are experienced to *say* – determines which formal proof(s) it indicates.
- *Techniques*: Any derivational account must explain the role of informal techniques of proof that aren’t obvious transcriptions of formally licensed inference steps in a way that’s compatible with the posited epistemic role of formal derivations.

Coupled with *Phenomenology*, these legitimate demands on any derivational explanation are a high bar to meet. I’m going to assume – within the framework of this paper – that a derivational account that meets all of these demands adequately isn’t possible.[18] I’m hardly alone in drawing this conclusion, of course; most of those philosophers who have studied informal rigorous mathematical proof agree. If so, epistemically speaking, informal rigorous mathematical proof is sui generis in one sense: we can’t explain its epistemic properties in terms of formal derivations in

[18]See Azzouni (2017a) for discussion of how well various derivational accounts fare in meeting these demands – better than one might expect. I draw the conclusion, nevertheless, that (at the end of the day) they can’t be met by such accounts. It’s here, however, that issues about the large size of formal transcriptions (as compared to the original informal proofs) arise; this poses a challenge, for example, to the derivational account being able to explain *Agreement.* It’s also here that issues arise about the explicit rules governing derivations (including ones of logic) needing to be treated as only tacitly available to ordinary mathematicians – and *essentially* tacit until the emergence of Frege’s explicit characterization of logic.

artificial languages that the informal proofs supposedly correspond to.[19] We must look elsewhere for an explanation. I'll start my particular take on this by noting an historical irony. If I'm right about what I claim in the rest of this paper, then Frege's invention is a garden path (or rabbit hole) – at least as far as the epistemology of mathematical proof is concerned. Formal languages, it turns out, have nothing directly to do with the epistemology of informal rigorous mathematical proof. Formal derivations play no more of a role in explaining the properties of ordinary mathematical proofs – implicitly or explicitly – than strange Platonic objects do.

4 Algorithms

I slightly overstated things in the concluding paragraph of the last section. The study of formal languages *isn't* a garden path or rabbit hole – not with respect to resolving the puzzles of informal rigorous mathematical proof (although it's still true, I'll argue, that formal languages – *directly* – play no role in the epistemology of mathematical proof). What turns out to be extremely important to resolving the puzzle of the epistemic strangeness of ordinary mathematical proof is something that the progenitors of formal logic (Frege, Russell, Hilbert,. . .) didn't anticipate. This is that the study of formal languages would directly lead, in the course of hardly more than a half century, to Gödel's incompleteness theorem (Gödel 1931), which, in turn, would enable (provably equivalent) characterizations of the crucial notion of an algorithm (Church 1936; Kleene 1936; Turing 1936). My claim: it's algorithms – and not formal languages (of any sort) – that are the key to understanding the

[19]I use a qualifier here ("in one sense") and earlier when characterizing Tanswell's correctness condition ("only in the sense that . . .") not to foreclose on the view, held by many of these philosophers (but not me) that sociological vectors explain *Agreement* among mathematicians – more generally, that what we may describe as conformity among mathematicians about mathematical proof is to be explained along the same lines that conformity in other areas (e.g., dietary practices) is to be explained. Although I think such sociological vectors do play a role in what mathematicians write down (for example) in giving an informal proof because what they assume as known by their colleagues shifts over time, or because of inertial effects in styles of proof, I don't think these factors play a very large role otherwise. Indeed, the centuries-long perception that mathematical reasoning is special in a way atypical of reasoning, generally, is a perception of its evident stability vis-à-vis social factors *in contrast to* reasoning elsewhere. (Manders (2008, 67) writes that "Euclid . . . Apollonius and Archimedes, are virtually without error: their every result has a counterpart in modern mathematics, even if subsumed in patterns of claims and proofs recognized much later." Absence of "error" is code for: successfully resists sociological intrusions.) In the course of this chapter I'll invoke intuitively effective recognition procedures that are available for (much of) informal rigorous mathematical proof to explain this. See what follows in the rest of this chapter.

epistemic strangeness of informal rigorous mathematical proof. Seeing exactly what the role of algorithms is in that practice will solve this ancient philosophical problem.[20]

I'll start with a brief description of Turing's well-known characterization of algorithms via what have come to be called (deterministic) *Turing machines*.[21] A Turing machine is an idealized rule-governed writing device – a Turing-machine head – located on an infinitely long tape that's divided into linearly ordered adjacent cells. Each cell is either blank ("□") or has a squiggle ("~") written on it. The Turing-machine head occupies one cell at a time; and it's governed by a finite number of rules which dictate all its possible actions and nonactions, depending on (a) what

---

[20]A *lecture*: Following the rhetorical lead of Lewis (1991, 57–59), many philosophers, for example, Burgess and Rosen (1997, 34) glibly laugh (or otherwise engage in something that's an approximation of laughter) at the idea that philosophers of mathematics can provide insights that might have an impact on the practice of mathematics or even yield something of interest to mathematicians. This purported laughter is silly. To make the first obvious point: philosophical concerns needn't be mathematical concerns. Sometimes, as with the philosophical puzzle of the epistemic strangeness of mathematical proof, this is because the philosophical wonder is *occasioned* by a specific practice, in this case, mathematical proof. What makes mathematical proof an occasion of philosophical concern is a *contrast fact:* mathematical proofs seem different from arguments in other areas of discourse as I indicated in the opening paragraphs of this chapter. Because the professional mathematician needn't be concerned with what arguments look like in other areas of discourse, and because she needn't be concerned with what it is that makes mathematical proofs different, these concerns, strictly speaking, aren't *mathematical* ones. So too, issues about mathematical ontology are philosophical concerns, not because the mathematicians' use of "there is" in their own practice should be challenged in some way, but because (again) it's a usage that offers puzzles across *all* discourses. (It can turn out for purely linguistic reasons that the mathematicians' use of "there is" is metaphor or a terminological extension of the ordinary phrase; discovering this and publicizing it isn't, by virtue of that, to necessarily recommend a revision in mathematical practice.) Recognizing philosophical puzzles and what's making them *philosophical* puzzles isn't easy – not even (apparently) for people who are officially philosophers.

But a rather more straightforward observation should be made, especially after noting one of Lewis' remarks, mentioned in the last paragraph (59) – italics his: "I'm moved to laughter at the thought of how *presumptuous* it would be to reject mathematics for philosophical reasons. . . . If [mathematicians] challenge [a philosopher's] credentials, will you boast of philosophy's other great discoveries: [and now Lewis gives a list of purportedly bizarre claims made by one philosopher or another]? Response to Lewis:" Classical physical theories of solar combustion were refutable in the late nineteenth century on the basis of the fossil record and other geological evidence: the Earth was clearly more long-lived – by a large magnitude of order – than the Sun could possibly be, given the then contemporary physics. Imagine a Lewis-style physicist snobbily turning his nose up at an argument (by a geologist, say) against classical physics *solely because* she's some upstart from outside a "genuine science" who dares challenge accepted physical theory. Alas (for such hierarchical views of knowledge gathering), that's how confirmation *works:* evidence for something can come from *anywhere* – even (dare I say it) from philosophy.

[21]Turing (1936, 118) uses the word "automatic." I should note that my presentation of Turing machines will be slightly eccentric in several ways – one of which is that I'll use meaningless symbols instead of the usual numerals "1" and/or "0." I do this to accentuate certain philosophical points about the interpretation of Turing formalisms that I subsequently make. Standard presentations of Turing machines, incidentally, are everywhere (especially in numerous textbooks) – if the reader wants or needs a (standard) refresher course in Turing-machine formalisms.

state S_i the machine head is in, and (b) what's written on the cell the machine head is located at. The machine head is capable of triple-actions. First, it erases/writes-in (or doesn't) the cell it's at, then it moves to the left/right (or doesn't), and then it shifts to a new state (or doesn't); what it does at any cell is dictated by the "state" it's currently in. Call the finite list of rules that describe what a Turing-machine head does, given the states $S_1, \ldots, S_n$ it's in, the *machine table* of that Turing-machine head. Let R mean, "move right," L mean "move left," and B mean "stay in place." Here is a sample machine table for a Turing-machine head with three states:

$$(S_1, \square) \hookrightarrow (\sim, L, S_1);\ (S_1, \sim) \hookrightarrow (\square, R, S_2);$$

$$(S_2, \square) \hookrightarrow (\sim, L, S_1);\ (S_2, \sim) \hookrightarrow (\sim, B, S_3)$$

This fully describes what this machine head will do, given its location on any cell, and it's being in any state. (I'll now individuate machine heads by the machine tables they obey.) The machine table reveals that if the machine head is in state S_1, and it's on a cell which is blank, it will write a "~" on that cell, move the left adjacent cell, and remain in S_1.[22] If this machine head is in state S_2 on a cell on which a "~" is written, it will leave the "~" in place, not move, and change from the state, S_2 to the state, S_3. There are no instructions for this machine head when it's in S_3; thus, the machine head will, as it's described, "halt."

Call any tape with a distribution of "~"s in its cells a *configuration.* Given a machine head located on a particular cell of that tape (what can be called a *Turing-head configuration*), call an *input* the configuration before the machine head initiates a finite sequence of actions; call the *output* the configuration that results after the machine head has completed a finite sequence of actions. Consider a set of Turing-head configurations, where the configurations are all different. Call this an *algorithmic operation* – that is, an algorithmic operation is a set of differing configurations, each of which has a Turing-machine head (with the same machine table) located on one cell of that configuration. Call the resulting modification by the actions of the machine head after a finite set of steps an *algorithmic calculation.* If the machine head halts, that's an *implementation* of the algorithmic operation.

I've deliberately described the operations of Turing machines as manipulating *meaningless* marks on tapes; consequently, the algorithmic operations (as just defined) are meaningless as well. They're open of course to being interpreted – and it's how widely they can be interpreted that makes the Turing-machine formalism so fundamental to our notion of calculational/effective processes (and, in particular, to the effective recognition procedures of formal languages that I described in earlier sections of this chapter). Let's make the (various) interpretations of Turing machines explicit. Consider a set of Turing-head configurations (consider, that is, an algorithmic operation). An *interpretation* of this algorithmic operation is a

[22]Sometimes, as I illustrated earlier, the machine head is described as "scanning" the cell it's located on. But this talk implicitly builds more machinery into the machine head than is needed.

mapping of its configurations to configurations of objects *along with* a mapping of its machine-head actions to corresponding actions on the corresponding configurations of objects.

Example: Identify "~" as the numeral "1" (interpret "~" as standing for the number 1). Next, single out a set of configurations that can, using this identification, represent the counting numbers (e.g., "~~~" is 3; "~~~~~~" is 6); indeed, single out a set of configurations that represent pairs of such counting numbers (e.g., "~~~□~~~~~~"). Consider the set of Turing-head configurations, where the Turing-machine head is located on the blank cell between strings of "~"s that represent these pairs of counting numbers. We can design a machine table (that halts with respect to these pairs) that can be proven to generate a single representation of a counting number that's the sum of any pair of numbers (represented by an initial pair of represented numbers in this set of Turing-head configurations); we can design a more complex machine table (that also halts with respect to these pairs) that can be proven to generate a single representation of a counting number that's the product of a pair of counting numbers represented by a Turing-head configuration in this set. These are typical exercises in any textbook on (or with chapters on) Turing machines.

Second example: Now consider a mapping of the same set of Turing-head configurations, not to numbers and operations on numbers (as just illustrated), but instead to bead configurations on a physical abacus,[23] and to (certain) physical operations of moving those beads. This is an interpretation of that set of Turing-head configurations too. For that matter, so is any mapping of that set of Turing-head configurations to any numerical representation of numbers (and manipulations of those representations). Interpretations of Turing-head configurations, that is, can be to notational objects or to worldly objects; correspondingly, the actions of a Turing-machine head can be interpreted as manipulations of Platonic *abstracta*, idealized operations on notional objects, or as (also idealized) physical actions on real objects.

The Turing-machine formalism (along with all interpretations of it) is unexpectedly robust along *several* dimensions. I'll discuss this in the rest of this section.

To begin with, describe as *Turing-machine formalisms,* a certain class of variations of the above description of Turing machines. For example, consider Turing-machine heads that scan more than one cell at a time – finitely many, let's say, or ones that can recognize more than one symbol in a cell, finitely many, let's say (or even countably many).[24] And now call *mechanical procedures* those sets of operations (on any collection of objects) that (as above) a Turing-machine formalism can be interpreted as. It has been proven, strikingly, that increasing the powers of a Turing machine in the above ways doesn't lead to an increase in the set of mechanical procedures that Turing machines (under an interpretation) can execute: these remain the same. It's furthermore quite striking what kinds of powers machine heads

[23] I'm going to be fairly spotty in describing abacuses; it's easy to find out about these things in articles in print and on the web.

[24] Consider, in particular, Gandy machines (see Gandy 1980).

must be granted in order to enable the class of mechanical procedures to become strictly larger than what can be managed by Turing machines. If, for example (Abramson 1971), "extended Turing machines" are stipulated that are able to write and read uncountably many symbols, the resulting class of mechanical procedures is strictly larger than those executable by conventional Turing machines. The class of mechanical procedures a "mechanical" device is capable of is strictly larger as well, if, unlike Turing-machine heads, it can complete infinitely many actions in finite amounts of time.[25] It appears that the notion of Turing-machine computability isn't only an upper limit on *human* computation (as Turing originally meant it to be) but also an upper limit on designed-machine (e.g., computer) computation – at least if the physical processes of our universe are what they appear to be.[26]

To continue the discussion of the robustness of the Turing-machine formalism, let's now (temporarily) restrict ourselves to interpretations of Turing-machine algorithms as computing operations on counting numbers. These numerical algorithms can be characterized in other mathematical ways – in terms of being definable from certain classes of functions for example.[27] These various characterizations have been proven to be equivalent to the set of Turing-machine algorithms (as defined above). With a little bit of argument (turning partly on the fact that any interpretation of a set of Turing-head configurations can be routed through an intermediate interpretation in terms of numbers), the same set of mechanical procedures is picked out by all of these characterizations.

Second, consider Church's thesis. (Or, alternatively described, the Church-Turing or the Turing-Church thesis.) A preliminary "intuitive" notion: call a function of the natural numbers an *intuitively effective function* if it's one that can be carried out by human(s) using pencil and paper. Church's thesis is: A function of the natural numbers can be calculated by an intuitively effective method if and only if it is computable by a Turing machine (subject to the above interpretations). "Intuitively effective," so described, *really is* an "intuitive" notion. The use of "intuitive" means various things to various philosophers and logicians[28] – but for current purposes it can be taken to indicate a hand-waving (unanalyzed and nonmathematical) characterization of an open-ended class of what strikes us as humanly possible calculations.

[25]See Copeland (2019) for further discussion and citations of the growing literature on this.

[26]All bets about this are off, of course, if the study of gravitational singularities and the like, where time – in particular – seems to have unexpected properties, allows the design (at least in principle) of devices that can complete infinitely many calculations in what amounts to finite amounts of time.

[27]In particular, the early publicized characterizations in terms of lambda-definable and recursive functions have been joined by further (provably equivalent) characterizations in terms of register machines, canonical and normal systems, combinatory definability, Markov algorithms, and others.

[28]Turing (1936, 135), notably, uses the word "intuition" in relation to his statement of the thesis, writing, that "no attempt has yet been made to show that the 'computable' numbers include all numbers which would naturally be regarded as computable. All arguments which can be given are bound to be, fundamentally, appeals to intuition, and for this reason rather unsatisfactory mathematically." Warning: "intuition," as used here – following Turing's usage – can be misleading. See the parenthetical remarks in footnote 114.

In mathematical practice (specifically, in the study of recursive functions and effective computability), Church's thesis is often used as a proof-shortcut, to avoid the tedium of exhibiting a Turing-computable function for an intuitively effectively calculable numerical function, for example, in Rogers (1967) where a numerical operation is described (or recognized) as humanly calculable and then via Church's thesis that operation is deduced to be Turing computable. The intuitive notion of an effective procedure is idealized in various ways. In particular, we imagine the humans doing these calculations as having sufficient time, tenacity, tools (pencils and paper), and abilities to continue the task (without errors) as long as needed for the particular algorithmic calculations involved. Call this a *persistence idealization*.[29]

A way of attempting to *prove* – in the (informal rigorous mathematical sense) Church's thesis is to argue that the hand-waving ("intuitive") description of an intuitively effective procedure can be characterized by (is coextensive with) a mathematical characterization (of some sort), and then use that characterization to prove Church's thesis. There are several versions of this strategy. Kripke's (2013) version relies on "Hilbert's thesis":

> Every (human) computation can be formalized as a valid deduction couched in the language of first-order predicate calculus with identity.

"Human computation," however, as used here, is every bit an intuitive notion as "intuitively effective method"; thus, crucially, Hilbert's thesis relies (as Church's thesis does) on an identification of an "intuitive" notion with a mathematically precise notion – namely (using my terminology) an identification of "humanly computational" with "derivable in the language of the first-order predicate calculus with identity." The reason that these empirical theses are needed is because a branch of (pure) mathematics, algorithm theory, is being applied empirically to human (and machine) computation. It's *always* an empirical question (a "thesis," if you will)

---

[29]Here's a generalization of "intuitively effective numerical function" – beyond cases of numerical functions – to any method whatsoever (I'm borrowing this formulation from Copeland (2019)):

M is an intuitively effective method if:

1. *M* is set out in terms of a finite number of exact instructions (each instruction is expressed in a finite number of symbols);
2. *M* will, if carried out without error, produce the desired result in a finite number of steps;
3. *M* can (in practice or in principle) be carried out by a human being unaided by any machinery except pencil and paper;
4. *M* demands no insight, intuition, or ingenuity, on the part of the human being carrying out the method.

I treat the above description as allowing that the instructions in question can be written in a natural language. I also take it to be topic-neutral: *M* can be a method that's applied to anything whatsoever. (I'm not claiming this is intended – or not intended – by Copeland.) I'll take up the topic neutrality of the intuitive notion explicitly in Sect. 5.

whether or not a branch of pure mathematics can be applied successfully to the entirety of an empirical domain that we try to apply it to.[30]

In any case, there is a related family of equally intuitive theses sometimes described as Church's thesis, and sometimes distinguished from it (see Copeland (2019) on this). Consider, in particular, the "maximality thesis": All functions that can be generated by machines (working in accordance with a finite program of instructions) are computable by intuitively effective methods (Gandy 1980). An argument deriving this thesis about "machines" is, in my view, possible along the lines I indicated in the material footnote 26 is appended to; I won't pursue this any further now. In any case, evidence for Church's thesis (and its relatives) continues to accrue by the explicit exhibition of particular Turing algorithms (directly or through equivalences) as able to execute specific intuitively effective procedures and those that machines (of various sorts) carry out. This, as I indicated, is a different dimension along which the applications of Turing-machine formalisms are robust.

There is yet a third dimension, along which Turing-machine formalisms, treated as characterizations of mechanical calculability, have proven strikingly robust. It's one already indicated by a central aspect of the proof of Gödel's incompleteness theorem – that it turns on a reinterpretation of computable functions of number theory as functions of the syntax of a language. This *hints* of the vast potentiality for the interpretations of algorithmic operations – a vast potentiality that's being executed (as I write) by developments in computer technology. For intuitively effective procedures are undertaken by humans on all sorts of objects (real and unreal) in all sorts of ways that are only slowly being interpreted in what amounts to Turing-machine formalisms – for example, a robotic hand picking up a paper cup without crushing it, or all the numerous (and increasingly realistic) video and virtual-reality games and films.

5 Recognition Procedures in Games

I turn now to developing a point I made in footnote 29 – that intuitively effective methods can be stated in ordinary language. Indeed, this is why Hilbert's and Church's theses are "intuitive" theses: ordinary reasoning and ordinary effective methods are invariably couched in natural language – and transliterating such to a formal setting is precisely to apply a branch of pure mathematics to these natural-language activities (as I indicated at the end of the last section). Coupling this point with the observations in the last section about interpreting algorithmic operations allows us to see how ubiquitous intuitively effective methods are in our lives. Let's start with a discussion of games. Many games (although not all games) essentially involve elements and configurations of elements with recognition procedures that players (and observers) of those games rely on. Chess, and other board games, are

[30] I return to a discussion of Church's thesis, generalizations of Hilbert's thesis and formal systems, in Sect. 15.

obvious cases (and so are card games). Recognition procedures are required to distinguish chess pieces, the squares on the board (and the spatial properties of those squares, e.g., adjacency relations), and to recognize admissible moves. Other games, despite involving "luck," are similar. In many board games, the pieces are moved along designed paths after throwing dice (to determine the acceptable number of steps a piece can take). Without appropriately designed intuitively effective recognition procedures, these games would be impossible to play – we'd be fighting over whether an *appropriate* move had been made; we'd even be fighting over what move, exactly, we were fighting over.[31]

Central to the Turing-machine formalism seems to be – to use a commonly used distinction – that the algorithms it describes are digital and not analog. Mechanically recognizable operations are required to be step-by-step ones, where each step is clearly distinguishable from every other step. Turing (1936, 136) makes this point explicitly:

> We may suppose that there is a bound *B* to the number of symbols or squares which the computer [human calculator] can observe at one moment. If he wishes to observe more, he must use successive observations. We will also suppose that the number of states of mind which need be taken into account is finite. The reasons for this are of the same character as those which restrict the number of symbols. If we admitted an infinity of states of mind, some of them will be "arbitrarily close" and will be confused.[32]

And yet, many games, unlike the board games I've just described, seem to be intrinsically analog: they take place physically, and continuously, in space and time. Such games, baseball for example, appear nevertheless to have intuitively effective recognition procedures. One is inclined (at first glance) to say, of any set of admissible moves in this game – hitting a pitcher's throw appropriately with a bat – that it's an *infinite* set of possible moves. After all, the variations of possible movements in space and time are infinite. How then can a game like baseball – and any game that similarly takes place continuously in space and time – be amenable to an application of the Turing formalism?

There are at least two mistakes involved in this line of thought. The first is that a certain kind of idealization – central to all algorithms that humans actually carry out but eliminated (usually) from presentations of such algorithms – is being

[31] It's been claimed against me (more than once) that games are at best analogous to Turing-definable algorithms because games have rules that are underspecified or even inconsistent. Tanswell (forthcoming) thinks these defects "are ironed out by social features of game-playing." It's, however, a ubiquitous feature of algorithms that there is no general test for "bugs" (there is no general test for inconsistency): they must be *spot-corrected* when discovered. In no significant sense does this introduce "social determinants" into practices, like games, that are governed by algorithms.

[32] Notice that the generalizations of Turing's apparatus (e.g., in Abramson 1971) mentioned in Sect. 3 are ones that shift to an essentially transhuman framework in exactly the ways that Turing rules out. For example, that the purported human can distinguish uncountably many symbols or that the purported human can complete an infinite number of discrete actions in a finite amount of time.

misdescribed. Consider the symbol "~" that I characterized all Turing-machine heads as using. Were an actual machine involved, the tokens of this symbol would be a somewhat ill-defined set of physical items – ill-defined because no matter how we designed Turing-machine heads, they will fail to sort *every* otherwise scannable physical token into one or the other class. In the same way, a chess piece can be placed on a chess board so that it's unclear (to the humans playing the game) which of two neighboring squares it's on. For that matter, the borders between squares on a chess board can be fuzzy, or badly designed chess pieces are possible where humans easily confuse the pawns with kings. We idealize the items and configurations of the items in a game (using what I called in Sect. 2 "exclusion idealization") by simply treating such confusing items as not occurring in the game. A last example is the intuitively effective recognition procedures associated with the derivations of formal languages: these too, when the derivations in such formal languages are actual ones that people (or computers) are constructing, involve exclusion idealizations so that by assumption there are no tokens so badly written that the recognition procedure for those derivations can't be executed.

The second mistake is to take the apparent analogness of our experience of space and time (and, in particular, our actions and perceptions of such actions) at face value. Given a finite amount of space and time, and a possible action across that space (and in that time), such as waving a hand – despite the impression otherwise, we *cannot* distinguish infinitely many variations of movements. What we are capable of seeing, and distinguishing, are only finitely many such movements. This is true of anything we can do in space and time. The analog appearance of our own behavior (or its appearance in the behavior of others), consequently, is an illusion.[33]

This shows that that the appearance of analog practices in such games doesn't belie an application of a generalized Church-Turing thesis to them: The intuitively effective recognition procedures that such games exhibit can be taken to be ones amenable to the Turing notion of algorithmic calculations.

The other crucial point that games illustrate in abundance are intuitively effective recognition procedures that are written in *natural* languages. We can use Hilbert's thesis, if we wish, to draw the conclusion that these can be formalized – indeed, to draw the conclusion that these games, in their entirety, can be formalized. And by this, I mean, transliterations to language-based formal systems.[34] Several connected points must be made about this, however. Our grasp of the intuitively effective recognition procedures used in such games – intuitively effective recognition procedures that children grasp – are ones *grasped directly,* and not via an implicit

[33]See Azzouni (2017a), especially Sect. 3, on this. To show the amenability of these games to Turing formalisms, in my view, it must be shown that the intuitively effective recognition procedures at work in these cases involve an unseen grid-structure as the field against which the admissible moves in these games are being evaluated. The details are in the article just cited.

[34]"Formal" systems, as I illustrate in Sect. 8, can be diagrammatic as well as language based.

(psychological) application of Hilbert's thesis. The neurophysiological/psychological facts that underlie these capacities are ones to be explained (that *are* being explained) in terms of visual-system capacities, saliencies, and the like. Our abilities to physically navigate our world are what underwrite the intuitively effective recognition procedures used in games. This is one reason, among others, why these recognition procedures are only partially described by the "rules" that officially govern such games. Usually taken for granted, and left unmentioned, are the physical details of the elements of such games (e.g., baseballs) that are often only partially specified in terms that explain the successful use of recognition procedures during these games; usually, tokens of them are simply exhibited. The same is true of the primitive alphabet for formal systems. These are, in practice, just listed – that is, the intuitively effective recognition procedures are *assumed* – using tokens, for example: the constants, a_1, a_2, a_3. . ., the individual variables, x_1, x_2,. . ., etc. I'll label these, in Sect. 13, *embodied algorithms*, and say a bit more about them then.

When, however, such games are transliterated entirely to language-based systems, the idealized physical properties of the objects and configurations of them that are *used* by the intuitively effective recognition procedures associated with those games must be treated as part of the subject matter to be described; they must be explicitly characterized and treated axiomatically. This is because language-based formal systems use only intuitively effective recognition procedures for distinguishing items of an alphabet: we're required only to see (by "eye") linear concatenation relations between tokens, to be able to distinguish the distinct primitive vocabulary tokens, one from another, and to be able to recognize which they belong to.[35] So, since the other intuitively effective recognition procedures, the ones that enable us to recognize the relations among the chess-board squares, are set aside, these relations must be explicitly described by terms that are axiomatized and treated as governing the "pieces on squares." (I'll say more about this in the next section.)

I turn now to making points about ordinary rigorous mathematical proof that are analogous to those that I've just made about games. Just like a certain subfamily of games, ordinary rigorous mathematical proof involves numerous intuitively effective recognition procedures that mathematicians grasp directly and not via formal transcriptions of those procedures into the medium of formal languages. This is why the derivations occurring in the family of artificial languages that are studied by logicians are irrelevant to the epistemic properties of ordinary mathematical proof. In going forward, I assume a generalized version of the Church-Turing thesis: I assume that all intuitively effective procedures *are* Turing computable.

[35]The other sorts of linear syntactic-relation patterns we – or some of us, anyway – "just see" are helpful but not required by most linear alphabet formal systems we study.

6 Language-Based Algorithmic Systems and Algorithmic Devices

Straightforward examples of language-based algorithmic systems are *axiom systems.* These have been studied extensively. A primitive vocabulary is given and syntactic rules for constructing well-formed expressions, including sentences, are defined. On this basis, in turn, a subset of the sentences are singled out as axioms, the finite sequences of sentences that are *acceptable derivations* are defined, and finally, the theorems (the sentences that appear on any line of an acceptable derivation) are defined as well. All of these have intuitively effective recognition procedures in the ways described earlier – the definitions I've just alluded to are designed to enable these intuitively effective recognition procedures to piggyback on earlier ones (e.g., the intuitively effective recognition procedure for theorems is designed on the basis of the one for acceptable derivations). Intuitively effective recognition procedures in formal systems are nested on one another in the way that the intuitively effective recognition procedures for primitive vocabulary, sentences, acceptable derivations, and theorems are nested here.

Example (Church's P_1)[36]: The primitive vocabulary are parentheses, (,), a two-place sentential connective, $\supset$, a one-place sentential connective, $\neg$, and an infinite list of sentential variables, p, q, r, s, p_1, q_1, r_1, s_1, p_2, q_2, ...[37] All and only the sentences are defined like so: the sentential variables are sentences; if P and Q are sentences, then $\neg$P and (P $\supset$ Q) are sentences. The axioms are: $(p \supset (q \supset p))$, $((s \supset (p \supset q)) \supset ((s \supset p) \supset (s \supset q)))$, and $(\neg\neg p \supset p)$. The syntactic-transition rules are two (I use the symbol "$\vdash$" to indicate syntactic transitions): For all sentences, P and Q, P, (P $\supset$ Q) $\vdash$ Q; if b is a sentential variable that appears in P and if P* is the sentence that results from the substitution everywhere in P by a sentence Q, then P $\vdash$ P*. A derivation is a finite sequence of sentences where each sentence is either an axiom or follows from earlier sentences in the sequence by a syntactic-transition rule.[38]

There are two ways to think about syntactic-transition rules. One is to think of them as structuring every *acceptable derivation:* each sentence in an acceptable derivation is either an axiom or follows from earlier sentences in that derivation by a syntactic-transition rule. The other is to think of them as methods of generating acceptable derivations from other acceptable derivations. Given any acceptable derivation, we can generate a new (augmented) acceptable derivation by introducing a new last sentence that is either an axiom or a sentence that follows from sentences in the original derivation by a syntactic-transition rule.

[36]From Church (1956, 78), I've modified P_1 by introducing an abbreviation that Church (1956, 78) gives later.

[37]Notice that there is an implicit intuitively effective recognition procedure in place for these, turning on an antecedent intuitively effective recognition procedure that's available for numerals.

[38]This apparently austere system is equivalent (via definitions) to the full two-valued classical sentential calculus with all definable (two-valued) connectives.

The point I want to stress about the intuitively effective recognition procedure for derivations in this example (and this is typical of many axiom systems) is that this recognition procedure directly operates in terms of the intuitively effective recognition procedures for whole sentences and not via any intuitively effective recognition procedures on the elements that such sentences are composed of. The intuitively effective recognition procedure for whole sentences of course depends on intuitively effective recognition procedures for the primitive alphabet – as I noted earlier. I'm here stressing a different point, however: Syntactic-transition rules are possible that rely not on whole sentences but instead directly rely on certain expressions *in* those whole sentences. Consider the alternative algorithmic formal system in the next example.

Second example: Same as the previous example, except that there are no axioms, and in addition to the two syntactic-transition rules given in that example, we have three additional syntactic-transition rules. For any sentences P and Q,

$$Q \vdash (P \supset Q);$$
$$(S \supset (P \supset Q)) \vdash ((S \supset P) \supset (S \supset Q))$$
$$\neg\neg P \vdash P.$$

The first two syntactic-transition rules are focused on the connective, $\supset$; the third focuses on the connective $\neg$; it's the occurrence of these connectives in certain syntactically indicated locations in sentences that license these syntactic transitions. In this sense these syntactic-transition rules are centered on a particular syntactic element and not on the sentence the element occurs in. For any expression e, call such syntactic-transition rules *e-centered syntactic-transition rules.* This characterization can be generalized to families of expressions E. Consider, for example, those Peano syntactic-transition rules that are E-centered on the constant 0 and the successor function′.[39]

Now consider an expression e and a language (*any* language, natural or otherwise) in which e appears. I call any set of e-centered syntactic-transition rules an *algorithmic device.*

The individuation conditions on algorithmic devices are importantly weak. Apart from the fact that the above definition allows algorithmic devices to be nested within one another, it's not required of an expression e governed by a set of e-centered syntactic-transition rules that constitutes an algorithmic device A that it be *fully* governed by those syntactic-transition rules. In the second example, just above, both $\neg$ and $\supset$ are governed by the original two syntactic-transition rules as well as by the new ones I then introduced. So too, an expression e, in a formal system, can be governed both by e-centered syntactic-transition rules and other syntactic-transitions rules (e.g., axioms) which aren't e-centered.

[39]For that matter, consider Church's P_1. This is an E-centered algorithmic device, where $E = \{\supset, \neg\}$.

The conditions on something being an algorithmic device are weak in a second sense. Although I've introduced the notion with respect to language-based formal systems, such devices can (and do) occur elsewhere. They show up, specifically, in natural languages and, indeed, they show up everywhere in mathematical practice, as it occurs in natural languages (as I'll indicate in the sections to follow). But, for the moment, consider the old philosophical example, "bachelor," where the meaning of this word is stipulated to mean: unmarried male. We have, according to this aspect of the meaning of the word, the following syntactic-transition rule P |- P*, where P* results from P by zero or more replacements of "bachelor" with "unmarried male."[40] More generally, the meanings of words can be taken (in some cases) to license syntactic-transition rules in arguments. This is compatible with those meanings being governed by other conditions – sufficient-condition application rules of some sort (for example) – which don't involve syntactic-transition rules.

So notice: consider any expression that seems governed by a small set of meaning-transitions (or meaning postulates) that enable the recognition of a class of truths or inferences containing that expression. Such a set of meaning-transitions is a kind of algorithmic device – not by any means the only kind of algorithmic device, of course. And notice, as well, that any expression that's so governed is usually governed by all sorts of other conditions too – both generalizations that involve many other words, and more restricted conditions that govern that expression along with others. The last point to stress is that it's come to be recognized that natural-language expressions – that we otherwise understand the meanings of – don't usually have definitions (necessary and sufficient conditions in terms of other expressions). They have, at best, some necessary conditions and some sufficient conditions. Some of these are encapsulated in the conditions of an algorithmic device; but others may be an implicit (or explicit) understanding of patterns of things we apply such expressions to (e.g., all blue things).

One thing – when studying logic – that makes "Gentzen" or "natural deduction" rules more user-friendly than axiom systems is that in this respect they're modeled on our impression of the meanings of natural-language expressions. We experience the natural-language meanings of expressions as localized to (and due to) those very expressions. This is why, say, a characterization of the "meaning" of the truth-function "&" like so:

$$p\&q \vdash p,$$
$$p, q \vdash p\&q,$$

seems to capture via the *meaning* of "&" the logical properties of "&" – as it operates in derivations of sentences from other sentences.

[40]The syntactic rule must be finessed to avoid counterexamples involving expressions like "Bachelor Buttons," which is the name of a flower. (Sadly, "Unmarried Male Buttons" isn't the name of that flower – or, I think, any flower.) I skip this.

For current purposes, the importance of algorithmic devices is that such items – focused on new expressions or, as it were, modifications of devices already governing old expressions – can be added to any language (formal or otherwise, one that already has one or another proof or derivation system, or one without) so long as the expressions the added devices are focused on (still) syntactically fit in with that language. For example, natural languages have noun phrases, verb phrases, and the like. If the new terms an algorithmic device is focused on are actual noun phrases or verb phrases, all that's required of them is that they operate syntactically (outside the confines of the application of the algorithmic device itself) like other noun or verb phrases already in the language – in particular, as these occur in natural-language sentences.[41]

7 Some Observations About Algorithmic Devices, Conceptual Engineering, Synchronistic Effects, and Imperatives

Imagine a natural language in which we have in place intuitively effective procedures for recognizing some instances of "Q follows from P" facts. Or, imagine that the methods we have aren't intuitively effective. The introduction of algorithmic devices can augment, make easier, or even introduce (to begin with, that is) an intuitively effective recognition procedure for a class of "Q follows from P" facts. Some algorithmic devices (or, sometimes, some transitions of configurations licensed by the device – see Sect. 9) should be seen, essentially, as "black boxes": sentences P, Q, and R (say) are transformed (by an algorithmic procedure) into notational or physical items that the device can operate on; after it does so, the resulting item(s) are transformed (by another algorithmic procedure) back into a sentence, S say. We then draw the conclusion, on the basis of this, that: "S follows from P, Q and R." In other cases, the connections between natural-language words – nouns and verb phrases, etc. – and the inner workings of the algorithmic device are more intimate (as it were). We can describe the transitions of the device in terms of transitions between sentences. The algorithmic devices I'll go on to discuss later will exemplify these differences.

A familiar first illustration: consider the numeral expressions of a natural language ("one," "two," . . ., "eleven," "twelve," . . .). These appear in natural-language sentences as both noun phrases ("one is less than two") and adjectivally ("There are six apples in Granok's basket"). Consider the introduction of a notational calculational device that provides shortcuts – indeed a *method* – for adding (Arabic notation

[41]This introduction of terminology into natural language – grafted from an algorithmic device – needn't be restricted to nouns and verb phrases. Whole sentences possessing what otherwise amounts to an alien grammar are introduced as *idioms* – whole syntactically-unparseable blocks with meanings. This is how – for example – those interpreted sentences of a formal language, untranslatable into natural language, are nevertheless expressed there. (Consider, e.g., those first-order sentences with long sequences of quantifiers that have no syntactic cousins in natural language.) I can't get any further into this now.

and the tableau layout for addition that utilizes, e.g., the base 10 properties of that notation). The inference of "There are seventeen apples in both Mordesh's basket and Granok's basket" from "There are seven apples in Mordesh's basket" and "There are ten apples in Granok's basket" is licensed by this calculational device.

In cases where there is already an antecedent proof or derivation practice, augmenting that practice with an algorithmic device in this way can induce far-reaching global effects on what can be proved or derived. A melodramatic example is when the supplementation of an already in-place derivation system with a new algorithmic device renders the augmented class of derivations trivial: *everything* can then be shown. But, there are many less melodramatic, indeed *routine,* cases.

Describe the introduction of an algorithmic device A into a language – formal or otherwise – with an already-present derivation/proof practice as *syntactically conservative* if, given an augmented derivation/proof D* of a sentence S, there is an antecedent derivation/proof D of S.[42] Often – very often – the addition of an algorithmic device into an already-present proof practice *isn't* syntactically conservative.[43] Failures of syntactic conservativeness – which, with regard to proof, are *successes* not failures – are nevertheless surprising because of our (always in place) local expectations about the noneffects of words (and the rules governing those words) on sentences in which those words don't appear. We don't expect that the introduction of a set of concepts that aren't "about" a subject area can induce new implications that are *purely* about that subject area.

An illustratively surprising result of this sort (Harris 1982) in a purely logical setting is that if we have an antecedent formal language governed by intuitionistic connectives, and we add classical negation (without placing syntactic restrictions on where that negation can appear), so that the system now has both syntactically unrestricted intuitionistic and classical negations, all the *other* connectives become classical. In particular, results like $(((p \supset q) \supset p) \supset p)$ – Peirce's law – and $((p \supset q) \vee (q \supset p))$, as well as (in the predicate calculus – here, let "$(p \Leftrightarrow q)$" stand for "$(p \supset q) \,\&\, (q \supset p)$"), $((\mathrm{F} \vee \forall x \mathrm{G}x) \Leftrightarrow \forall x(\mathrm{F} \vee \mathrm{G}x))$ and $((\mathrm{F} \supset \exists x \mathrm{Gx}) \Leftrightarrow \exists x(\neg \mathrm{F} \supset \mathrm{G}x))$ can now be shown. None of these are intuitionistically valid. This happens *precisely* because these results can be shown via the newly introduced classical negation and they suffice to render "$\supset$," and the other heretofore intuitionistic connectives, classical. This phenomenon is *typical* in mathematics – it's one reason

[42]Notice that built into this definition is that the sentence S doesn't contain any *new* terminology. Not required (and in fact rarely available) is that, given a syntactically conservative augmented derivation/proof of a result, a derivation/proof of that result using the original methods can be *easily* shown. In practice, mathematicians are drawn to augmenting their methods when that enables them to easily show new results – regardless of whether those results can be shown with great difficulty or at all by methods they already have. (Indeed, the question of whether or not innovations are conservative or not in the above sense was rarely entertained until modern times. I say "rarely" because, actually, I know of no cases where it *was* entertained – but I hesitate to be definitive about this. In the case of complex numbers or infinitesimals, the worries were ontological, not proof-theoretical – for example, are these weird numbers real or *imaginary?*)

[43]Or, more commonly, no one knows.

why pure mathematics is so interesting: new concepts introduced into a subject area yield surprising and unexpected results about the antecedent concepts that don't (or don't seem to) follow from reasoning about the antecedent concepts alone.[44] The phenomenon, I should stress, shows up in natural languages too – even in those cases where our concepts are governed (rather weakly) by meaning conditions alone.[45] Call these *synchronistic effects.* Lesson: synchronistic effects are everywhere in mathematics, and, as well, everywhere in our inference practices *outside* mathematics.[46]

Algorithmic devices, as I've described them, are very flexible in two senses. Consider a device that's centered on a set of concepts (terms), c_1, c_2, etc. – for example, a set of natural-language numerals (finite or not). The scope and effects of any such algorithmic device centered on $\{c_1, c_2, \ldots\}$ can be changed by (i) supplementing or changing its rules directly, and apart from this (ii) by introducing other algorithmic devices that are also centered on $\{c_1, c_2, \ldots\}$, on different terms, or both. The usual (informal) impression that there are specific concepts (e.g., the counting numbers) associated with these terms, and that they have certain fixed meanings is usually unaffected by these supplementations of and to algorithmic devices (again, consider the case of counting numbers) which aren't seen as changing their meanings.[47] Strikingly, as I mentioned earlier, the distinction between such

[44]See Azzouni (2000), where I discuss this phenomenon in relation to "relevance intuitions" and recommend it as a crucial element of the intrinsic interestingness of (pure) mathematics.

[45]Example? Consider a natural language in which the set of natural-language numerals is finite.

[46]The label "synchronistic" in "synchronistic effects" is meant to stress not so much simultaneous impact as (apparently) unrelated impact. Field-style nominalist programs, incidentally, are motivated by the (false) belief that the impact of applied mathematics to empirical topic areas can be shorn of synchronistic effects – and that, indeed, this is a *virtue* of mathematical applications. It's neither a virtue nor is it even possible. Because of synchronistic effects, the introduction of new mathematics to an antecedent subject area (mathematical or otherwise) always yields *surprises* – ones in the empirical case that are open to empirical refutation. That is, we can learn that a branch of mathematics should *not* be so-applied empirically. See Detlefsen (2008) for a discussion of the historically long and influential requirement – proponents include notables such as Aristotle and Bolzano – that epistemically good mathematical proofs should be pure (not involve alien or incongruous notions). See the discussion of the work of the mathematician Brunfiel, in Hafner and Mancosu (2008), who insists on pure proofs, and the discussion of Hilbert's analysis of Desargue's theorem in Hallett (2008). Notice what the purity requirement entails, with respect to incomplete axiomatizations. We must replace the proofs using the "alien" vocabulary with ones totally in the language of the unaugmented system. Why assume that the new axioms – in the familiar vocabulary – will be easy to find or that the resulting proofs will be particularly attractive – in *any* sense, other than in the sense of the sheer absence of alien vocabulary?

[47]One source (perhaps *the* source) of the relative insensitivity of meaning intuitions to changes in what I shortly call inferential scope is that such intuitions – especially in the mathematical case – are ontologically focused: the number words, for example, are experienced as referring to particular objects with such and such properties. Therefore, the embedding of such number-words (and rules governing them) into a context with other words (governed by different rules) is seen as simply the putting of numbers – counting numbers, say – into a context with other objects (e.g., rational numbers) which leaves the properties of the counting numbers unchanged. This (falsely – by means of a kind of use/mention error coupled with some sort of intrinsic-property confusion) gives the impression that doing so will be (inferentially) conservative.

supplementations enabling easier and/or quicker proofs (e.g., calculations) and those supplementations introducing new results that can't be shown otherwise, wasn't made – until modern times – except insofar as it was recognized that a change to an algorithmic device or the introduction of a new one into a practice could render the whole thing inconsistent.

Call the *inferential scope* of an expression, or of the concept associated with that expression, the set of truths *intuitive-effectively* determinable by the collection of inferential tools available to practitioners at a time (by "inferential tools," I mean, for example, licenses to infer based on meanings, such as "B is a brown house" implies "the outside of B is brown," logical principles, if any, algorithmic devices, if any, etc.).[48]

The heart and soul (as it were) of *conceptual engineering* in mathematics occurs via the development and/or supplementation of informal rigorous algorithmic devices. One important lesson: Most of the literature on conceptual engineering focuses on meaning and meaning change, and on issues of definition.[49] But "meaning" so-called and definitions – even in mathematics – are (largely) red herrings. I've already explained in the last few previous paragraphs why meaning is a red herring. Here's why definitions are too. Definitions are *local* in the following sense: they provide necessary and sufficient conditions for an expression *in terms of other expressions*.[50] In this way the inferential scope of a defined expression is linked to the inferential scopes of the other expressions occurring in the definition. But these, as I've illustrated, are always directly open to shifts in referential scope – even by the introduction of new expressions governed by new algorithmic devices, and so *that's* where the engineering action is.

Let me summarize this way: The contours of our concepts are always inferentially specified by our actual (intuitively effective) inferential tools; that is, the scopes of these concepts change as we change the tools we use to determine the truths that the expressions corresponding to these concepts occur in. *This* is conceptual engineering. Much of it is invisible to us because we understand our concepts and corresponding expressions *not* to change in their meanings even when the actual contours of their inferential scopes shift drastically.

I turn now to an important symptom of the presence of algorithmic devices in informal rigorous mathematical proof. Tanswell (forthcoming) has noticed that

[48]I do *not* mean, notably, *intuitive-ineffective* semantic determinations of classes of truths, such as "intended models" or intuitively ineffective "tools" such as validity defined in second-order terms using "standard semantics" – these truths are *not* intuitively effectively determinable.

[49]See, e.g., Cappellan (2018), Tanswell (2018). My work on this topic doesn't so focus. See, e.g., Azzouni (2006).

[50]Thus, definitions can range from the apparently trivial ("a triangle is anything that's a triangle") to the relatively "deep" – where the impression of depth arises from how significant or illuminating the expressions being used in the definition appear to be: in every case, trivial or not, a definition gives "necessary and sufficient conditions."

imperatives occur a lot in mathematics.[51] Indeed, the form of many informal rigorous mathematical proofs is that of *recipe-instructions,* of the form: Do U; do V; notice W; and then notice X.[52] Often, imperatives occur seamlessly with indicatives in a proof, as in the example of a proof from Rudin (1970, 357) that Tanswell (forthcoming) quotes (I've changed some of the lettering in what follows):

> Theorem: If A is a commutative complex algebra with unit, every proper ideal of A is contained in a maximal ideal. If, in addition, A is a Banach algebra, every maximal ideal of A is closed.
>
> Proof: The first part is an almost immediate consequence of the Hausdorff maximality principle (and holds in any commutative ring with unit). Let I be a proper ideal of A. Partially order the collection $\mathfrak{R}$ of all proper ideals of A which contain I (by set inclusion), and let M be the union of the ideals in some maximal linearly ordered subcollection $\wp$ of $\mathfrak{R}$. Then M is an ideal (being the union of a linearly ordered collection of ideals), $I \supset M$, and $M \neq A$, since no member of $\mathfrak{R}$ contains the unit of A. The maximality of $\wp$ implies that M is a maximal ideal of A.

Tanswell (forthcoming) suggests that the practice of using imperatives is *so* ubiquitous – especially in the context of diagrammatic proofs – that it requires a view of mathematics that centralizes imperatives. (He calls it the *recipe model of proofs*.) The first point to make is that – syntactically speaking, at least – the presence of imperatives is a relatively shallow matter. They can always be replaced with "if . . . then" indicative-locutions. This can be seen already in the above quotation, in which the imperatives may be naturally rewritten like so:

> If the collection $\mathfrak{R}$ of all proper ideals of A which contain I (by set inclusion), is partially ordered, and if M is the union of the ideals in some maximal linearly ordered subcollection $\wp$ of $\mathfrak{R}$, then M is an ideal (being the union of a linearly ordered collection of ideals), $I \subset M$, and $M \neq A$, since no member of $\mathfrak{R}$ contains the unit of A.

I described imperative-to-indicative translations as relatively shallow (grammatical) matters. Someone may resist this characterization for the following reason. Consider a (cooking) recipe Tanswell (forthcoming) gives:

1. Preheat the oven to 170°/gas.
2. Peel the bananas and lay them in a snug-fitting heatproof dish. Finely grate the orange zest and put aside, then halve and squeeze over the [bananas the] juice of 1½ oranges.
3. Drizzle with the honey and sprinkle over 2 teaspoons of the cinnamon, then roast in the oven for around 20 minutes, or until golden and soft, then set aside to cool.
4. Place the bananas and their syrup in a blender, then blitz with the milk and yoghurt.

---

[51]Indeed, several philosophers of mathematics have noticed and stressed the use of action terminology in mathematics. Larvor (2012) and De Toffoli and Giardino (2015) are three. They think its presence is significant for philosophically understanding informal rigorous mathematical proof. I disagree – what follows indicates why.

[52]Permissions seem to come up as well. See the discussion of the postulates of Euclidean geometry forthcoming in Sect. 11.

5. Add in a handful of the zest, the juice of the remaining orange half, a pinch of sea salt and the rest of the cinnamon, then churn in an ice-cream machine. If you don't have one, freeze the mixture in a suitable container for about 3 hours; every hour, take it out, whip with a whisk, then return to the freezer.
6. Serve in bowls with a scattering of orange zest and a fine grating of dark chocolate.

This can be rewritten, letting "(*n*)" stand for what follows each numeral *n* in the above list of commands:

If you (1), you (2), you (3), you (4), you (5), then you can (6).

Or we can introduce a stylized list (corresponding to how recipes are given):
If you:

(1),
(2),
(3),
(4),
(5),
then you can (6).

Rewriting Tanswell's recipe in indicatives, that is, reads as naturally as a list of commands does, but comes across a bit long-winded because (grammatically speaking), we can give ("bark out") commands, one after the other, in single sentences, just as in recipes; but we can't do the same with the list of "if" conditions followed by a "then" conclusion. The use of commands in recipes, however, is a mere stylistic fact about how recipes are given in cookbooks; it may even be due to the fact that a complex set of conditions are more easily surveyable if they're broken up into single sentences, and so one naturally puts them into command-form even though in no real sense is a recipe a list of *commands*. One can certainly imagine a different recipe-tradition, where recipes are instead given as predictions of what will happen under certain circumstances. In such cases, "if" conditions would be written, stylistically, as single sentences. Indeed, it can be argued, as I just have, that – semantically – that's what recipes actually are: they're predictions. (Consider the old organized-crime joke: *B*: Do this or a horse's head in your bed tomorrow morning. *C*: You're telling me what to do. *B*: No, I'm not; I'm predicting the future here.)

This point aside, algorithms (as I indicated in Sect. 4) are very naturally couched in terms of commands; machine tables look like lists of commands. It should be no surprise, therefore, that if an algorithmic device is (implicitly or explicitly) present, that presence will be usually – if not invariably – syntactically marked by imperatives.[53]

[53]That said, as Transwell's quotation from Rudin (1970) shows, and as many of the examples from De Toffoli and Giardino (2015) also illustrate, sometimes instructions on how to manipulate notation are ontologically disguised as descriptions of actions on *abstracta*. Again, a motivation for this may be that it's easier to understand single sentences (in command form) than it is to understand long "if, if, . . . if, then" sentences. In addition, it should be noted, this is an illustration of the ubiquitous use/mention conflations that occur everywhere in mathematics – right from Euclid's work on, as I illustrate in Sect. 11.

We do many things with the sentences of the natural languages we speak; in particular (and perhaps fundamentally), we express our thoughts and describe states of affairs; neither of these uses of language directly involve inference or inferential devices. The focus in this chapter is what we do with language entities by the machinery of intuitively effective inferential tools: these license the expression (speaking, writing) of certain language entities given the expression of other language entities – often sentences from sentences, but this isn't just a matter of sentences. In any case, we carry out such inferences in ways that (sometimes) phenomenologically appear rule-governed. Central examples – the ones under study here – are some of the cases where the process of inferring one sentence from another is seen as the antecedents semantically implying the consequents. These goings from one item of language to others are syntactic *functions* (their *domains* are pieces of language which they map to other pieces of language – what's described as their *ranges*). There are, of course, many ways we go from one sentence (or piece of language, words, for example) to others – other than by inference. In any case, in the foregoing, I've been focusing on syntactic functions – and in particular, on syntactic functions from sentences to sentences that have intuitively effective recognition procedures. Call these *syntactic algorithmic functions*; call the particular syntactic algorithmic functions that (at least in part) govern inference *inferential algorithmic functions*.

It should be obvious by this point in the paper that an inferential algorithmic function can occur in a language or context that hasn't (or doesn't appear to have) other inferential algorithmic functions. Indeed, it's possible for there to be an E-centered algorithmic device in a language that has no other inferential algorithmic functions. Return, for example, to the case of categorical syllogisms, and consider the expressions, "all," "no," "not," etc. that occur in such syllogisms; let these expressions be our family E, and let the syntactic method(s) of recognizing valid syllogisms be the algorithmic device in question. Recall that I mentioned (in Sect. 2, footnote 12) that – empirically speaking – we don't know whether natural-language reasoners actually use the intuitively effective syntactic recognition procedures (that are in principle) available to determine when they're faced with a valid syllogism. Imagine, however, natural-language reasoners who *do* use an intuitively effective recognition procedure for categorical syllogisms, but who are otherwise unable to reason effectively. Imagine, that is, that these language users have no intuitively effective recognition procedures for inference patterns *other than* the one they use for categorical syllogisms. Such language-users – even if they have languages that otherwise are as rich as natural languages – can't *reason* except via categorical syllogisms. I hypothesize that such reasoners would strike us as strangely unable to grasp certain things that strike the rest of us as "obvious."[54]

[54]This is because, as we now understand, syllogistic reasoning is quite restricted in its inferential scope. However, an important cost in understanding "logic" more broadly is the loss of something that was (tacitly) assumed about reasoning until modern times – that there are *decision* procedures for recognizing validities and contradictions, and indeed (this was believed by Hume), except for issues of length, they're trivial ones.

I define an *algorithmic inferential system* as one in which *all* the inferential functions have intuitively effective syntactic recognition procedures. A first-order language which is Gödel-complete is one where the *semantic* implication function for the sentences of that language is coextensive with an intuitively effective syntactic recognition inference procedure. A higher-order language, with a model theory (an interpretation), according to which its syntactic deduction rules are Gödel-complete (Henkin models, as they're called), also has a semantic implication function that's coextensive with an intuitively effective syntactic recognition procedure. A higher-order language, however, which (relative to what's described as "standard model theory") is Gödel-incomplete, is one where the implication relation lacks *any* intuitively effective recognition procedure, let alone a syntactic one.[55]

8 Diagram-Based Formal Systems; Shin's Venn-II

As the discussion of games in Sect. 5 illustrated, just as we manipulate pieces of language (using intuitively effective recognition procedures), we can manipulate pieces of *anything* using intuitively effective recognition procedures. And similarly, there are functions, and indeed algorithmic functions, with domains (and ranges) governing some of the ways that we manipulate *things* – be those things baseballs, chess pieces, robotic hand-movements, items of notation, etc. Call a *context* the set of objects (and configurations of objects) that a set of algorithmic functions is defined on. A context can be the context of an algorithmic system or of an algorithmic device.

Diagram-tokens are a kind of object and so diagrammatic algorithmic systems and diagrammatic algorithmic devices are subcollections of object-based algorithmic systems and object-based algorithmic devices.[56] Examples of diagrammatic devices are Euler and Venn diagrams (as introduced originally by Euler and Venn), Euclidean diagrams, as practiced by ancient Greek mathematicians, and so on; examples of object-based algorithmic devices are physical tools like abacuses, various mechanical calculating devices, etc.

The "formal" in "formal language-based algorithmic system" conveys that the system in question is mathematically *fully-explicit* – in particular, the primitive vocabulary is given, and the rest of its syntax is explicitly defined on that basis (regardless of

[55]See Van Benthem and Doets (1983) for details. Notice that the absence of intuitively effective recognition procedures for sentence-functions turns on those functions being characterized in some other manner – as I suggested, for example, semantically. This will be significant for the later discussion of mathematics because, unlike games, the items in mathematics – ordinary-language sentences, sentences containing specialized terminology, and algorithmic devices (diagrammatic and otherwise) – are usually (explicitly and implicitly) interpreted. See what follows on this, but especially Sect. 10.

[56]Actually, all notation-tokens are, in this sense, objects (or, in the case of utterances, events). Nevertheless, I'll continue to distinguish between algorithmic devices using what I call objects – and here I'll be alluding to devices like abacuses (but also computers) – and those based in notation that we manipulate, such as diagrams.

whether certain kinds of syntactic items, like sentences, are, as a result, intuitive-effectively recognizable). Call a *language-based formal system* one that defines all syntactic objects in terms of the primitive vocabulary and (one-dimensional) concatenation alone. Nothing, in principle, prevents *equally formal* systems being based on syntactic constructions *other than* one-dimensional concatenation, for example, adjacency constructions in two or three dimensions (what we might call "higher-dimensional concatenation").[57] Call such systems *formal diagrammatic systems.* As with language-based formal systems, some of these formal diagrammatic systems are algorithmic and some aren't. [58]

My use of the distinction between one and higher-dimensional syntactic (concatenation) relations to distinguish non-diagrammatic from diagrammatic formal systems may seem obviously wrong – especially to those who are familiar with Shin's (1994, chap. 6) discussion of the distinction between diagrammatic and non-diagrammatic representation, as well as Giaquinto's (2007) discussion of this. Giaquinto focuses on whether conventions or resemblances are being utilized to determine interpretations – and to what degree; Shin focuses on whether new syntactic devices are introduced by a system of representation – ones that require conventions to use.

Apart from this, it may seem clear (to the reader) that a representation of individuals sitting between one another in space by locating their names one-dimensionally on a line, so as to imitate their relative spatial locations, is diagrammatic (Shin (1994, 156) – despite being one-dimensional – and it may seem, by contrast that the integral notation of the calculus, for example:

[57]I claim, but won't argue here – see Azzouni (2017a) – that if the resulting syntactic classes of items (sentences or admissible diagrams, and syntactically defined parts of these) are to be intuitive-effectively recognizable then there must be an implicit grid – in one, two or three dimensions – that the primitive items appear in; the syntactically defined items must be various (intuitive-effectively recognizable) classes of distributions of the primitive vocabulary across the grid. One approach to this places a lower-limit on the size of the areas of the cells of the grid; in one-dimensional concatenation systems (the ones we are most familiar with – alphabet systems), this is done, typically, by choosing a specific *font* for (all of) the primitive vocabulary items that appear in the (implicitly present and implicitly sized) cells. Underlying grid-frameworks aren't always explicit – they're not in Miller's (2007) Euclidean diagrammatic system FG, but they're made explicit in Mumma's Euclidean system Eu by means of arrays, which are the official backgrounds against which diagrams appear (see, e.g., Mumma 2010). I discuss these two formal diagrammatic systems in more detail in Sect. 11.

[58]I claim (see my comments on Feferman (2012) in Sect. 5 of Azzouni 2017a) that there are *two* diagrammatic systems in play in the proof of the Schröder-Bernstein theorem. One isn't algorithmic because one diagram that's alluded to in the proof is *actually* infinite. This diagram isn't *used* by mathematicians but only referred to. There is another diagram – a series of diagrams, actually – that are unbounded in size (although none of them are actually infinite). A finite number of these *are* used in the proof, and the pertinent properties of these diagrams are intuitive-effectively recognizable. There are many such examples of what should be called two-tiered diagrammatic proofs (the proof used to show what's called "Koch's snowflake" is another), ones which induce a structured semantics where the diagrams actually used by mathematicians belong to an algorithmic system but such diagrams are interpreted as referring to actually infinite diagrams in a system that isn't algorithmic; in turn, these actually infinite diagrams are interpreted as standing for functions (and sets) of various sorts. I can't get any further into this now.

$$\left[\int_0^{2\pi}\int_0^{\infty} e^{-r^2} r\,dr\,d\theta\right]^{1/2}$$

isn't diagrammatic, despite being two-dimensional (neither, one might think, is Frege's two-dimensional notation diagrammatic).[59] I'm tabling further discussion of this until Sects. 13 and 14 because it involves subtle complications involving the interpretation of diagrams; both Shin and Giaquinto's approaches to the diagrammatic/language-based distinction, that is, involve interpretations of diagrams and sentences. Restricting myself solely to dimensional considerations as they play out syntactically yields a rough-and-ready distinction between notations it's important to explore. (Whether it tracks the actual intuitive distinction between diagrams and items that aren't diagrams is relatively unimportant.) I'll consider – at the end of Sect. 14 – whether a principled distinction is available that's more in accord with intuitions about when a representational system is diagrammatic or not diagrammatic. I'll there suggest that the question isn't semantic as it, perhaps, appears to be, but is best taken to be a psychological one that requires empirical study to resolve.

In the rest of this section, meanwhile, I'll illustrate a fully-explicit formal diagrammatic algorithmic system: Shin's (1994) algorithmic system Venn-II. "Fully-explicit" means here that the diagrammatic system is as syntactically explicit (as mathematically explicit) as any typical interpreted language-based formal language. In the section that follows, I'll consider what stands in (in informal settings) for the semantic apparatus that's used to characterize formal systems – for example, model theory. After that, I'll consider some specific cases of algorithmic devices that don't occur in formal algorithmic systems – although they can be embedded in them (this is described in Sect. 12); included among these cases are those where algorithmic devices occur in natural languages; this is what happens in ordinary mathematical practice. One important case is the (ancient) use of Euclidean diagrams – where the details of the practice are somewhat controversial and complicated.

Venn-II, as Shin's choice of nomenclature partially indicates, is a formalization of Euler/Venn/Pierce diagrams.[60] The primitive vocabulary (Shin (1994, 115) calls them "primitive objects") are closed curves, rectangles, complete shadings of

---

[59]Interestingly, Giaquinto (2008a, 39) suggests matrices provide "a significant example of visual operations on symbol displays…." These strike me, on the contrary, as involving "visual operations" to exactly the same extent that one-dimensional catenation systems do – that is, only adjacency is involved and nothing more (e.g., part/whole spatial perceptions *aren't* involved, nor do other perceptual powers for recognizing two-dimensional visual patterns – that don't directly involve adjacency – seem involved: recall how one multiplies and add matrices). In any case, the difference between one and two dimensions looks significant to other philosophers even if (according to me) they misdescribe its import.

[60]Shin (1994) presents two formal diagram systems. Venn-I corresponds to categorical-syllogism reasoning; Venn-II is an expansion of Venn-I to a formal diagrammatic system that she proves is equivalent in expressive strength to a version of first-order monadic logic (one in which the empty set can be a model). There are other formalizations of these diagrams and various formalizations of extensions of them as well. One interesting extension are "spider diagrams" – see Howse et al. (2005) and, more recently, Stapleton et al. (2013), where there are references to earlier work.

(certain) spatial regions, and the diagrammatic elements, ⊗ and —, that appear within those regions. Well-formed diagrams are defined in terms of these elements (116), and persistence and exclusion idealizations are introduced so that we're presumed able to intuitive-effectively recognize (distinguish between) distinct diagrams containing any finite number of closed curves $C_1, \ldots, C_n$ contained properly within any number of rectangles (connected by lines), where, for any rectangle, containing a subset of these curves, $C_{i+1}, \ldots, C_{i+m}$, their intersections in the diagrams, and whether or not the symbols, ⊗ and —, appear in the regions defined by these curves, are also intuitive-effectively recognizable.[61]

The next step is to provide intuitive-effectively recognizable *diagram-transformations* (121–124): these are acceptable (intuitive-effectively recognizable) transformations of diagrams into other diagrams. The intuitively effective recognition procedure for diagram-transformations is based on the well-formed-diagram conditions, which in turn are based on the recognition procedures for the primitive vocabulary. The "obtainability" relation of a diagram from a set of diagrams ("$\vdash$") is defined via diagram-transformations (125). This obtaining relation is an intuitively effective syntactic one; it corresponds to the derivation-relationship of sentences from other sentences definable in formal language-based algorithmic systems.

As I mentioned before, unlike games (or most games, anyway), diagrammatic algorithmic systems and diagrammatic algorithmic devices usually have interpretations. This is one aspect of such diagrams that enable them to have such a significant role in mathematical practice. In the case of Shin's Venn-II, one particular interpretation is pretty natural.[62] It's a model theory where sets are assigned to rectangles, and subsets of those sets are assigned to the closed curves in the rectangles.[63] Using

[61] Shin (1994, 60, footnote 7) here relies on Polythress and Sun (1972) for the result that, given any arbitrary number of closed curves, a configuration exists (an "atomic diagram") in which for every subset of these closed curves, there is a visible region generated by the intersection of all and only the curves in that subset. Assuming a lower bound on the (tacit) cells that such curves are drawn across, this means that such configurations (and the curves themselves) are arbitrarily large. The evident implausibility of these diagrams being intuitive-effectively recognizable (we surely can't "recognize" such diagrams of arbitrary complexity – not even those containing only a small number of such closed curves) involves exactly the same sort of persistence idealizations that are invoked to allow that people, with pencils and paper, can continue a computation in time as long as needed. We are, in this case, idealizing the intuitively effective recognition procedures by allowing arbitrarily large diagrams that a person "scans" progressively (while retaining in her memory what she has seen). Exclusion idealizations are also at work when, for example, Shin (1994, 82) rejects diagrams containing partial shadings as "ill-formed."

[62] Interpretations for algorithmic systems, however, are never uniquely determined – no matter how "natural" any specific interpretation may be. This is equally true of diagrammatic systems; and so this can be confusing (especially where certain interpretations seem natural because of "resemblances," e.g., the supposed resemblances between drawn triangles and real triangles). I'll discuss this later, in Sect. 13, when I turn to debates about the "rigor" of diagrams in mathematical practice.

[63] I'm omitting niceties in describing how Shin uses this model theory to interpret the diagrams and their elements of Venn-I and Venn-II; among such niceties are ones that involve identifications of the sets that a model assigns to rectangles and curves in different atomic diagrams that are connected to one another by lines; this is needed, for example, to enable to Venn-II to express (some) disjunctive expressions.

this model theory, a notion of a diagram "being true in a model," can be defined, and via that, a notion of implication ($\models$) between diagrams can be defined as well (one collection of diagrams implies another diagram if every model that every diagram in the collection is true in, is one that the second diagram is true in as well); in terms of these notions of implication and obtainability, soundness and completeness are shown (i.e., for all diagrams D_1 and D_2, $D_1 \models D_2$ iff $D_1 \vdash D_2$).[64]

As the above model theory indicates, the diagrams of Venn-II can be associated one-to-one with the sentences in a formal language, specifically a first-order monadic language L.[65] In particular, the formal language-based system L and the formal diagrammatic system Venn-II have exactly the same metalogical properties with respect to the model theory they share.[66] Furthermore, as I indicated, this isomorphism between the diagrams of Venn-I and Venn-II and the sentences of L is intuitively effective. That is, we have an intuitively effective (decision) procedure for reading off from any diagram of Venn-I or Venn-II a sentence of one or another (interpreted) formal language L which "says" (in terms of the interpretations of both the language and the diagrammatic system) what the diagram "says." We can similarly depict what any sentence of the formal language L says with a diagram that captures its "content" exactly.

The model theory for L – traditionally understood – provides interpretations for all the sentences of L. That is, the model theory for L indicates what the sentences of L say. Further, given a specific model M of L, we can speak of the sentences of L being true or false relative to M. The same is true for the diagrams of Venn-II. Each diagram, using the same model theory, says something, and what it says (relative to a specified model M) is either true or false. The model theory for Venn-II and for

[64]See Shin (1994, chap. 5). The fully-general result, in terms of infinite collections of diagrams, that enables a compactness result for the implication relation of models, is given by Hammer and Danner (1996).

[65]Again, avoiding certain niceties, that the procedure of associating sentences with diagrams is intuitively effective turns on how the interpretation for a diagram (in terms of sets) is given – this yields a normal-form theorem: each atomic diagram corresponds to a conjunction of existential and negations of existential sentences describing the (finite) intersections of the sets corresponding to all the closed curves in a diagram, and a diagram (composed of atomic diagrams connected by lines) corresponds to a disjunction of these conjunctions. The reverse procedure turns on giving an intuitively effective method of transforming any sentence of the language to its normal-form equivalent. I stress that these procedures are decidable.

[66]On this, see Shin (1994, specifically Chap. 5), Hammer (1994), and Hammer and Danner (1996). One metalogical property the two systems (naturally) share: the theorems of both systems are decidable and not merely effectively enumerable.

Venn-I (for that matter) thus provides interpretations for the elements of the diagrams that also, as I indicated, yields a notion of truth-in-a-model for the diagrams.[67]

Relations between sentences and diagrams when formal systems like the above are in play are not only decidable, but involve metalogical properties such as completeness. Relations between sentences and diagrams are useful in mathematical contexts – and generally – even when such metalogical properties are missing, as they usually are when algorithmic devices are employed in natural-language contexts. Here, usually, model theory is absent as well – and so model theory *isn't* the tool by which mathematicians interpret diagrams. Nevertheless, we can (and do) legitimately speak of interpreted diagrams or, more generally, of interpreted configurations of objects (a certain sequence of executed movements on an abacus, say) as being true or false – or as being implied by later interpreted configurations. The interpretations in question, however, are ones given by specified sentences of natural language – or sentences in a natural language augmented with (technical and specialized) vocabulary items that nevertheless respect the natural-language syntax this vocabulary occurs in. I turn to developing this important point in the next section.

If we combine Shin's diagrammatic formal system together with a language-based formal system, the result is an example of a *formal* system of "heterogeneous reasoning" in Barwise's (1993) sense (Shin 1994, 188–189; Hammer 1994): a *formal* algorithmic system that has both diagrammatic and language-based elements. Although this kind of combination of systems – diagrammatic tokens along with language-based tokens – is atypical of most formal systems currently studied, the heterogeneity it exhibits is endemic in informal rigorous mathematics. I turn to attempting a description of how similarly "heterogeneous reasoning" works in informal settings.[68]

9 Informal Reasoning by Means of Algorithmic Devices

Consider the use of an algorithmic device **A** in a natural language context. As I indicated above, informal versions of the semantic/syntactic tools in play in formal systems like Venn-I and Venn-II arise here too, although notably they don't have all

[67]For example, Shin (1994, 118) gives a definition of a set assignment (a mapping of sets to the rectangles and closed curves of a diagram) *satisfying* a well-formed diagram. Well-formed diagrams, however, correspond to *sentences*; there is nothing in her definition of satisfaction involving variables as they occur in the formal language-based systems (e.g., those studied in Tarski (1933)). Atomic diagrams are given interpretations by mapping the curves to subsets of the domain of the model, and mapping the elements to the sets in question being empty or nonempty (see Shin 1994, Sect. 3.31). These correspond to sentential relations between sets, for example, *The intersection of A and B is empty*; *the intersection of A and B is nonempty*. The additional compositional facts – about diagrams – that Shin's definition of the satisfaction relation turns on involves the linking of atomic diagrams by lines (which corresponds to the disjunction of what the atomic diagrams represent).

[68]See Sects. 8 and 9 of Azzouni (2017a). I there described algorithmic reasoning as occurring on the "surface" of informal rigorous mathematical practice. This is a metaphor that I'm explicating here in terms of algorithmic devices.

the properties that their formal cousins have. I here sketch out what things look like in the informal setting.

Start with the observation that a natural-language algorithmic device **A**, as in formal settings, allows the intuitively effectively recognition of well-formed diagrams D_i.[69] As before, there is an intuitively effective, recognizable obtainability relation, $\vdash$, that **A** licenses between its diagrams. In addition, there are intuitively effective recognition methods of reading ordinary sentences *off of* the diagrams of **A**, $D_i \rightrightarrows S_j$, and reading sentences *into* diagrams of **A**, $S_j \rightarrow D_i$. Specifically, given a diagram D_i and a sentence S_j (from a certain set of designated sentences), we can often change that diagram into another D_j, where for all sentences S_n, we can recognize that if $D_i \rightrightarrows S_n$, then $D_j \rightrightarrows S_n$, and furthermore, we can recognize that $D_j \rightrightarrows S_j$. I'll call $\rightrightarrows$ and $\rightarrow$, respectively, "reading-out" and "reading-into" procedures.

These procedures may help themselves to language-elements that are contained in diagrams (e.g., letters that appear in the diagrams and that label diagrammatic elements of it) but they needn't. It also needn't be the case that, generally, one can mechanically generate all the sentences that can be read off of a diagram – although that's true of a lot of algorithmic devices (e.g., Shin's Venn-I and Venn-II, and, I think, the informal Euclidean-diagram practice). What's assumed here is just that it's mechanically recognizable for any sentence S_i and diagram D_j, whether or not $D_i \rightrightarrows S_j$ or $S_j \rightarrow D_i$.[70]

Here is the first important contrast with Shin's formal diagrammatic system – and with formal diagrammatic systems generally: there is no assumption of uniqueness in the reading-off and reading-into relations – nor need there be "normal-form" sentences that can be described as what captures (uniquely) what the diagrams of an algorithmic system **A** "say." That is, given a diagram D_i, there can be many different sentences, S_j, S_k, . . ., with appreciatively different (although compatible) contents, such that $D_i \rightrightarrows S_j$, $D_i \rightrightarrows S_k$,. . .. Similarly, given a sentence S_i, there can be many diagrams D_j, D_k, . . . (also with appreciatively different contents), where $S_i \rightarrow D_j$, $S_i \rightarrow D_k$, ... All that's assumed is that the reading-into and reading-off relations have intuitively effective procedures. In many cases, these reading-into and reading-off relations are additional decision procedures available to diagram users: given S_i and D_j, it can be determined whether, or not, $S_i \rightarrow D_j$ or $D_j \rightrightarrows S_i$.

[69]For ease of exposition, I'm describing all algorithmic devices as if they're diagrammatic, rather than more general object-based devices. What I say here can be applied easily to the broader case, although I won't (systematically) do that explicitly.

[70]Notice that we can often intuitive-effectively recognize, of two diagrams, D_i and D_j that if $D_i \rightrightarrows S_n$, then $D_j \rightrightarrows S_n$, without necessarily being able to recognize all the sentences S_j of which it's true that $D_i \rightrightarrows S_n$ and $D_j \rightrightarrows S_n$. For example, D_j might result from D_i by the addition of a single diagrammatic element to D_i (e.g., a line or cross) so that it's easy to tell how D_i and D_j differ in their sentential contents.

Define the *content* of a diagram to be all the sentences S_n such that $D_i \rightrightarrows S_n$. (I'll notationally indicate the content of a diagram D_i as Content (D_i).) As noted already, that a given sentence is in the content of a given diagram is intuitively effectively recognizable.[71]

The content of a diagram, importantly, can sometimes look *empty.* In certain standard language-based formal derivation systems, for example, this happens because syntactic-transition rules allow transitions of formal sentences to well-formed expressions with free variables – that don't strictly speaking, say anything – and back again. In the case of mechanical devices that enable calculations, this can happen because certain machine-state transitions can be to and from states that don't nicely correspond to sentences (natural-language or otherwise). Diagrammatic algorithmic devices can operate similarly. Call such syntactic-transitions between diagrams *semantically null* transitions. The algorithmic systems we'll be concerned with will be ones in which any such diagram has, nevertheless, an interpretation – a content; this will be imposed on it by syntactic-transition rules that generate such diagrams from earlier ones. (See the description of soundness results, immediately below.)

The content of a diagram is treated as the *interpretation* of a diagram. This is usually taken to be due to antecedent interpretations of the elements of the diagram, for example, that diagrams of triangles represent triangles, closed curves represent sets, and the interpretation of their syntactic locations in the diagram – for example, that a point-element within a triangle-element represents the presence of a point topologically within a triangle. That is, the interpretation of the diagram is taken to result compositionally from the syntactic placing of these elements in the diagram. There is much more to say about this which I can't do here.

In any case, we usually have the following important *soundness* results that hold of algorithmic devices **A** that are under study in this chapter:

For all diagrams D_i and D_j in **A**, $D_i, \vdash D_j$ only if

(a) For all $S_n \in$ Content (D_i), $S_n \in$ Content (D_j),
(b) For all $S_n \in$ Content (D_j), Content (D_i) $\models S_n$,
(c) for any semantically null transition from D_i to D_j ($D_i, \vdash D_j$), Content (D_i) = Content (D_j).

[71]Contents of diagrams here are sets of sentences. This isn't the only option. We may want to treat the contents instead as "what the sentences say": in other words, the sets of configurations of elements that the sentences are about. Doing it this way or my way doesn't matter, except insofar as the approach indicated in this footnote generalizes straightforwardly to model-theory – whereas my approach involves extra steps. The reader who impatiently wants to get past *sentences* to items that the sentences are about can substitute configurations of elements for sentences in my above definition of content. This is fine except that it introduces extra expository steps in my forthcoming discussion of how practitioners shift from diagrams to natural-language results that such diagrams are taken to show, for example, in the case of Euclidean geometry, Proposition 1 in Book I, that an equilateral triangle exists that has a given straight line as one of its sides.

Some comments about the algorithmic devices these soundness results hold of.

First, as I said, this is a *soundness result* for the proofs generated by the algorithmic device **A**, coupled with the reading-into and reading-off procedures. Unsurprisingly, for algorithmic devices in informal rigorous mathematical contexts generally, we don't have completeness results. These come up – when they're available – when such devices and the informal mathematics they occur in are embedded in one or another formal algorithmic system that's formally interpreted as well (e.g., in relation to a model theory).[72] I'll say more about this in Sect. 12.

Second, I don't think (and so I don't assume) that such soundness results were *shown* in informal settings before modern times. Rather, they were empirically established (or, perhaps more accurately, assumed). In addition (but related to this), practitioners tacitly employed reading-into and reading-off procedures without recognizing much about the properties of these procedures. They recognized that there were potential issues with these procedures only insofar as they grappled with unwarranted-generalization problems. These amount to (ancient and modern) questions about what the elements in a diagram can be taken for stand for (which, exactly, mathematical triangles diagrammatic triangles in a Euclidean diagram can be taken to stand for; what mathematical functions diagrammatic curves can be taken to stand for; etc.); related to this, what the reasoning with specific diagrams can be taken to have shown, generally. (I discuss this in further detail in Sect. 10.) Apart from this, with physically instantiated algorithmic devices, empirical assumptions are needed to show soundness results. This is because persistence and exclusion idealizations must be assumed as successfully operative when these devices are used.

Third (as I've already implicated in the second remark just made), the reading-in and reading-off procedures are crucial to the use of algorithmic devices in informal rigorous mathematical proof. This is because the theorems shown by any execution of an algorithmic device **A**, is read off of the "final state" of that device. In general, any theorem yielded from a diagram by a reading-off procedure is of the form, $S_1, \ldots, S_m \models S_n$, where $S_1, \ldots, S_m$ have been read-into the sequence of diagrams at various earlier points; and S_n is read-out from the final diagram.

Fourth (as I indicated in the discussion in Sect. 7, and especially in footnote 41), the natural-language that the reading-in and -out procedures rely on is invariably terminology-enriched.

This is already obvious in the later Euclidean tradition where "triangle," "line," and so on refer to abstracta, and not to irregularly shaped physical items, that the original Greek words surely referred to. The sentences read off of a diagram can also be, syntactically speaking, "whole cloth" items: alien jargon that functions idiomatically vis-à-vis natural language (i.e., items that are unparceable as far as natural language is concerned).

[72]Of course, completeness results – relative to certain given model theories – may not be possible, as Miller (2012, 34–35) indicates is the case with Euclidean diagrammatic formal systems and certain (intended) models.

Because of the crucial role of reading-into and reading-out procedures in these proofs, there is an important generalization of the notion of "heterogeneous reasoning" beyond cases of formal systems to cases where algorithmic devices occur in informal rigorous mathematics proof (to cases of mathematics, that is, that occur in natural languages). I'll call this *informal heterogeneous reasoning*.[73] The idea is straightforward. Suppose we have an algorithmic system **A** that licenses transitions among diagrams D_i; suppose also that we have ways of recognizing implications ($\models$) between sentences of natural language (be those sentences natural-language ones or sentences composed – in part – of supplementary and specialized mathematical vocabulary). A proof (relying in part on an algorithmic device) can look something like this:

$$S_i \models S_j, S_j \rightarrow D_k, D_k \vdash D_l, D_l \rightrightarrows S_m, S_m \models S_n,$$

where we take, as a result, $S_i \models S_n$.[74] This is justified by a soundness result, of the above form, for **A**.

Example: Suppose the relevant sentences of a natural language are ones like "ten plus seventeen is twenty-seven." Let D_i be the configurations of an object-based or language-based algorithm **A**, such as an abacus, or an (Arabic) numeral system for addition or multiplication. The elements of D_i are interpreted in natural language, say by the assignment of the elements of a diagram, "1," "2," etc. to the natural-language numerals, "one," "two," etc. or instead, by, say, configurations of the elements of a physical algorithmic device (bead patterns in an abacus, for example) being assigned to natural-language numerals. And we (consequently) interpret the diagrams, and (possibly) certain subconfigurations of those diagrams, as having truth values as a result.

Consider an *abacus*. Here are some typical "word problems" we can solve using it:

The auditorium has 75 normal chairs and 20 high chairs. How many chairs does the auditorium have in total?

A bricklayer stacks bricks in two rows with ten bricks in each row. On top of each row, there is a stack of six bricks. How many bricks are there in total?

Serena buys two packs of tennis balls for $12 in total. All together there are six tennis balls. How much does a pack of tennis balls cost? How much does one tennis ball cost?

In each case, we extract an implicit arithmetical question that the word problem poses, and then, using a reading-in procedure, we generate a (finite) sequence of configurations of abacus beads; in the first word problem, for example, the arithmetical statement to be shown is: "Seventy-five chairs plus twenty chairs is X chairs," where X answers the question, and thus the word problem. Call the

---

[73] I'm here generalizing Barwise's (1993) use of the phrase to algorithmic devices in natural language. See footnote 14 above.

[74] In general, of course, many sentences occur, both as antecedents, and in the stages where sentences are read into diagrams.

antecedent description the calculation, and call the substitution of a natural-language numeral for "X" in "X chairs," "X bricks," etc. *the result*.[75] The calculation is coded into the algorithmic device as a sequence of configurations of abacus beads, and the result is read off from the final configuration. Introducing the usual exclusion and persistence idealizations, we imagine that abacus devices can be arbitrarily large in size (and yet humans can still execute operations on them). There is an intuitively effective recognizable class of arithmetical statements (involving the ordinary-language addition, subtraction, multiplication, and division), where the calculation can be intuitive-effectively coded as a sequence of configurations of an abacus, and where the last configuration corresponds to the result. This, therefore, is an example of heterogeneous informal reasoning, where there are both language-based inferences (of, for example, word-problem statements into arithmetical statements) and abacus inferences. The abacus results, in this way, are returned to an ordinary-language arithmetic practice (such as it is).

We can imagine, instead, that the algorithmic devices in question are notational (short division, multiplication, etc.). Notice the result: Via the reading-in and reading-off procedures, algorithmic devices of all sorts can easily augment a sentence-based informal mathematical practice. In particular, the invention of a new notation (that provides an algorithmic device – or several) can be easily added to an already existing mathematical practice in natural language with preexisting algorithmic devices. It's then empirically established that the new algorithmic devices both enable the proofs of apparent new results or they introduce ease in yielding theorems that are already accessible otherwise. That the algorithmic device doesn't render the practice trivial by yielding inconsistent results (either internally or in relation to results established by other means) is also – and must be – empirically established.[76]

I need a label for the characterization I've just given: that informal rigorous mathematical practice involves the use of algorithmic devices (either language-based or not) along with reading-in and reading-out procedures. Call this the "algorithmic-device interpretation of informal rigorous mathematical proof," or (for short) the *device view*.

10 The Generality Problem

The diagrams we make, look at, and study are tokens of syntactic objects (of types) – that is, they're tokens of types of *physical* items (sketches on paper, in sand, fleeting utterance-events, etc.). Call *the generality problem* the question what interpretation

[75]Often, natural language is supplemented, for example, in this case with new numerals, so that the reading-out procedures are still ones into natural language – although now supplemented.

[76]I mean "empirically established," not in the sense that a mathematical proof can't be given for this; I mean it in the sense that we don't have a decision procedure for discovering things like this: not having unearthed an inconsistency *so far* doesn't mean an inconsistency isn't there to be unearthed.

is to be given to the elements of the diagram, and thus, compositionally, to the diagram itself. In the case of formal algorithmic systems, this is the question of what model or set of models the diagrammatic or language-based formal system is to be interpreted in terms of. *The straightforward solution* to the generality problem in the formal context takes Gödel *completeness* results to be the guide to answering this: Choose a model theory to interpret a formal system **A** where the validities and the theorems match.

This, when philosophers study formal languages, usually *isn't* taken to be an acceptable solution to the generality problem. On the one hand, there is *the uniqueness problem.* There are many "unintended" models for any formal theory – models of "nonstandard arithmetic" for first-order Peano arithmetic or the many unintended models for first-order ZFC, for example. It's felt by many practitioners that, despite our having no syntactic grip on any specific model in these cases, there nevertheless *is* an intended model – that is, there is a unique model that we (somehow) *intend* the axioms we write down to capture.[77] But, not only do the syntactic consequences of first-order Peano arithmetic fall short of extending over all the validities (theorems) of this intended model, any augmentation of Peano arithmetic that has an effective syntactically characterizable consequence relation falls short the same way (Gödel's incompleteness theorem).[78] Relatedly, it's felt by many proponents of certain logical systems – for example, higher-order logics – that a model theory vis-à-vis which their syntactically specified theorems are complete doesn't capture the intended meanings of the terms in those systems. According to the "intended" meanings of the quantifiers of such systems, for example, any effectively specified set of syntactically characterized theorems are incomplete.[79]

[77]This model, in the case of Peano arithmetic, is called "the standard model" – for example, by Kaye (1991).

[78]For the record, I'm skeptical there is genuine content to "intended model" in any case in which characterizing it outruns our syntactic resources. I can't get further into this now; see, instead, Azzouni (2018) for discussion of the case of arithmetic – which is a case where intended-model intuitions are especially strong. Notice what the problem is. We feel we understand – grasp the meaning of – the intended extension of a notion when it's given by negating a term whose extension we *do* grasp, for example, the consistent formulas of first-order logic (which are all those which *aren't* first-order validities or contradictions). On the other hand, there is certainly a sense in which we don't understand – don't grasp the meaning of – the extension of a notion if it isn't recursively enumerable. I omit further discussion of this very delicate matter except to observe that it's connected to the discussion – in Sect. 7 – of how our intuitions of meaning are insensitive to what I there called inferential scope.

[79]Nomenclature: I think, for the reasons just given, that the generality problem should be called, instead, the intended-model problem. The latter label more sharply focuses what's at issue, both in formal and informal settings. But the "generality problem" is the label Mumma and other philosophers in this literature use – so I'm sticking with it instead.

I intend to focus on that version of the generality problem faced by a specific informal diagrammatic system: the generality problem of ancient Euclidean diagrammatic practice. This version of the generality problem gets a lot of attention, both now and over the centuries. But it's important to indicate first that it's something of an illusion that a rather different form of interpretation is at work in formal settings – where models are supposed to be used to interpret formal languages – than in the case of informal mathematical practices with algorithmic devices, where, according to my characterization of them, reading-in and reading-off procedures into natural language are used instead.

The reason is one that was made clear long ago by Davidson's (1986) characterization of how Tarski's theory of truth is supposed to be used in interpreting languages: The semantics for any language – logical or otherwise (i.e., one in which there is a nonlogical vocabulary that's allowed to vary across models, or one in which there isn't any such nonlogical vocabulary) – occurs in an *interpreted* metalanguage.[80] That means that the meanings of the elements of sentences and diagrams, and the meanings of the syntactically composed sentences and diagrams occur via transliterations of them into the antecedently meaningful language of the metalanguage. And *that* means that an interpretation of a formal algorithmic system, specifically, a formal diagrammatic system, is – strictly speaking – a formalization of what I described in the last section as the reading-in and reading-off procedures that are applied to algorithmic devices.

More precisely, the formalization in question is accompanied by an interpretation of the algorithmic system that's a transliteration of a bit of informal mathematics – diagrammatic or otherwise – and including some algorithmic devices, reading-in and reading-off procedures, as well as (usually) some accompanying informal natural-language reasoning. The algorithmic system and its interpretation is specified in ordinary language (as always happens in informal rigorous mathematics) that has

[80]That the interpreting language be, strictly speaking, a distinct language – a "metalanguage" – isn't required. That's just the traditional way of doing it. For example, it's possible to have one language in which the interpretation of a part of that language take place in another part, and where the two parts are syntactically isolated from one another. I should add that in classical settings the background interpreted metalanguage is usually (a version of) set theory – that is, a mathematical subject matter that's ordinary informal rigorous mathematics (in a natural language). See what follows.

been supplemented with specialized vocabulary – in standard cases, with set-theoretic vocabulary.[81] If the metalanguage – in which this mathematical specification of the syntax and semantics of the algorithmic system takes place – is, in turn, formalized then the same process (of interpretation) can be repeated.

In a purely informal setting, the generality problem coalesces onto the reading-in and reading-off procedures; and this version of the problem has troubled mathematicians (with respect to diagrams) right from ancient times until the present. The Euclidean version of the problem is this: how is a diagram to be understood – to what do the elements of the diagram refer? Consider a drawing of a triangle, for example. One (natural) interpretation of it is that it represents a class of triangles that (roughly) match the *drawing's* angular properties: the acute triangles are referred to by drawings of acute triangles; oblique triangles are referred to by drawings of oblique triangles; and certain properties of other drawn figures – *and not others* – play similar scope-indicating roles. Call these *resemblance interpretations* of the diagrams in a diagrammatic practice.[82] Euclid's proofs don't seem to respect any obvious resemblance interpretation systematically: sometimes certain properties of a figure seem involved; sometimes other ones do. The generality problem (in the Euclidean setting) is made complicated by the fact that what determines the scope of

[81]Model-theoretic approaches to interpretation (and Davidsonian approaches to interpretation, therefore) cannot, contrary to Davidsonian claims (Davidson 1986), be taken as providing the theoretical framework for explanations of how we understand languages and of how the meanings and references of these languages are facilitated – that is, what the machinery is by which the terms of a language acquire the referential scopes that they have (e.g., that "everything" refers to everything; that "gold" has the extension it has). Call the latter language-world questions; call the former language-mind questions. The reason Davidsonian approaches don't answer language-world and language-mind questions is that, as I've just mentioned, the mechanism by which Tarskian-inspired model theories interpret languages is by providing compositional-element-to-element *correlations* between target-language elements and metalanguage elements that provide the interpretations for targeted elements by simply giving them the referential and meaning properties of the correlated metalanguage elements. Doing this can't answer language- world and language-mind questions unless we *already* have answers to those questions with respect to the metalanguage elements. That is, it *defers* answers to these questions for a targeted language to whatever answers we have for the metalanguage – none as far as the Davidsonian approach is concerned since that approach bizarrely takes the targeted language questions as *answered* by these correlations. Notice the objection *isn't* the truistic complaint that, if a theory of how language elements refer to the world is given, such a theory needs to be given *in* a language which embodies (or has) language-world connections and is one that we understand. The point, rather, is that no illumination of language-world and language-mind questions is given by "explaining" that how "pomme" manages to refer to all and only the pommes is that it happens in just the same way that "apple" manages to refer to all and only the apples. An analogous point holds of the question of how Capucine understands "pomme"; it doesn't help to say that she understands it in just the way that I understand "apple." See Azzouni (2017b) for an attempt to spell out what's required for a genuine attempt to answer language-world and language-mind questions. I'll also note that to complain this way about truth-conditional semantic theories isn't to complain about their ability to capture the compositionality of languages: they *successfully* capture compositionality phenomena.

[82]Philosophically refined versions of resemblance interpretations are still taken seriously – by for example, Shin and Giaquinto. I discuss them in Sect. 12.

a diagram are the reading-in and reading-out procedures; but they're *implicit*.[83] The result is centuries-long controversies over whether Euclid provides sufficient diagrams to cover all the relevant cases that seem to fall within the scope of his established results – whether, in fact additional diagrams are needed for particular diagrammatic proofs to successfully prove results that he seems to understand as having a certain generality (e.g., as about all triangles, as opposed to some obeying such and such specific conditions – for example, acuteness)[84].

The generality problem for informal diagrammatic practices seems to have a different flavor than the one faced by language-based systems. This is because there are additional properties that we're visually sensitive to above and beyond those visually available in one-dimensional concatenation systems. In addition to adjacency, for example, there are topological properties (containedness) that it seems that reading-in and reading-off procedures in some diagrammatic traditions can (and do) exploit. In those diagrammatic practices, additionally, mathematicians "just see" visually available relations that are often interpretationally correlated with properties of the mathematical items the diagrams are about.[85] This "just seeing" is (largely) why the reading-in and -off procedures exploiting what's "just seen" are tacit. Nevertheless, the generality problem is the same one that's faced by language-based systems; this is because nothing requires the visual properties of a diagrammatic system to be exploited by the reading-in and reading-off procedures: it's a convention to so-interpret syntactic properties – diagrammatic or otherwise – *regardless* of whether the procedures, that exhibit these conventions and that provide the interpretations, exploit resemblances or not.

When an informal diagrammatic practice is formalized, one explicitly "solves" the specific version of the generality problem that diagrammatic practice faces – for example, the Euclidean diagrammatic practice – via *two* moves, versions of both of which are used when formalizing *any* informal inferential practice. The first move is to mathematize the (physical) diagrams themselves by identifying them with one or another kind of mathematical object. Shin (1994), for example – recall Sect. 8 – mathematizes Venn-diagram drawings by characterizing them as closed curves

[83]Mumma (2010, 256–257) states the generality problem for the informal rigorous Euclidean practice this way: "The generality problem arises with Euclid's proofs because the diagram used for a proof is always a *particular* diagram. Euclid clearly did not intend his propositions to concern just the figure on display beside the proposition. They are applied in subsequent proofs to other figures, which are not exact duplicates of the original. And so, for Euclid, consultation of the original diagram, with all its particular features, is somehow supposed to license a generalization. But Euclid leaves the process by which this is done obscure."

[84]See Heath 1956, 245–246.

[85]These have come to be called "free rides" – a phrase introduced by Barwise and Shimojima (1995). I'll eventually characterize these in terms of what I'll call "embodied algorithms." See Sect. 13 below. Also see Azzouni (2013a), where I discuss the free-ride phenomenon, although without using that nomenclature.

within rectangles (that can be connected by lines) with certain mathematical properties.[86] This corresponds to our making metamathematically explicit the syntax of a formal language – as when the sentences of a language we speak are identified with certain abstract objects. As I described it (in Sect. 8), well-formed syntactic items are characterized, the syntactic-transition rules are given, and so on. The second move is to provide a model theory (a semantics) for the resulting language-based or diagrammatic formal system. Although the diagrams are recognizably identified with well-formed syntactic items, the reading-in and reading-out procedures are (relatively invisibly) encapsulated into the interpretations given by the correlations of the mathematical objects standing in for the actual diagrammatic devices (drawn circles, etc.) to model-theoretic items.

I now turn to a closer discussion of the formalization of the Euclidean diagrammatic practice.

11 An Algorithmic Device in Natural-Language Mathematics: Euclidean Diagrams

Everyone admits that Euclidean geometry, the *Elements*, is the most influential reasoning practice in human history.[87] Consider Book I of Euclid's the *Elements*. A post-Fregean temptation – one that's still alive among contemporaries – is to think of Euclid's approach as *the* earliest axiom system: a pioneer version of the formal language-based systems that emerged with Frege, but occurring in natural language, and in which diagrams play a purely heuristic role.[88] Views about exactly *how* the

[86]This is often implicit, but important: The abstracta that in the formalization replace the physical items we actually use in the informal practice – for example, the actual drawings – are chosen to preserve (at least in principle) the intuitive-effectiveness of the recognition procedures of the informal rigorous mathematical practice. Preservation of certain properties of the phenomenon being mathematicized is a typical precondition on the application of branches of mathematics; in general relativity, for example, we choose a mathematical object (a manifold) that replicates – at least up to current observational thresholds – the properties of spacetime.

[87]Mumma (2008b, 256, italics his) puts it nicely: "[The *Elements*] represented the limit of mathematical explicitness. It served as *the* paradigm for careful and exact reasoning." This is seen by Manders (1995), Mumma, and others studying the practice, to pose an historical puzzle: why the relatively widespread repudiation as being *obviously* inadequate of the Euclidean proof-practice by later nineteenth-century mathematicians? I provide my answer to this question in what follows.

[88]Note the suggestion: diagrams play a purely heuristic role *in the informal rigorous Euclidean practice.* That the practice can be formalized one way or another in a pure language-based system (e.g., in the formal system of Avigad et al. (2009)) is irrelevant because it follows from Hilbert's thesis that this can always be done.

reasoning with this system works – in particular, what role the diagrams have – has shifted over time. Plato's early characterization treats the diagrammatic elements as crucial to reminding us what the proof elements (words and diagrams) refer to, and therefore why the results diagrammatically shown are true.[89] Later views of reasoning treat a perfect item of reasoning – a proof – as a completely explicit series of inferences without gaps (no "missing steps"); this view is present in Descartes[90] but remained programmatic until Frege. Frege, however, sustained the picture by assuming, naturally enough that the (artificial) language that proofs occur in is one with a fixed semantic interpretation, and where each step follows immediately (by rules) from earlier ones. Once the uniqueness of the possible interpretations of artificial languages (or of the introduced terminology in natural languages) is stripped from the picture, what remains (indicated by the phrase "by rules") are intuitively effective recognition procedures for proofs. The internal syntactic structure of the language and the interpretations of the logical operators across models that it embodies are all that remain to fix the interpretations of its sentences and that of the other language elements.[91]

Until it was recognized that diagrammatic proofs – specifically, the transitions between diagrams in a diagrammatic proof – utilize intuitively effective recognition procedures just as language-based proofs do (and thus are equally open to formalization), the temptation remained to treat our recognition of the inferential properties of proofs as matters of "intuition" to be eliminated. So, I'm hypothesizing that a certain picture of proof (gaplessness) and the impression that gap-free proofs are

[89]My interpretation of Plato's view of reasoning is that it's akin to perception: "seeing" in the sense of understanding and "seeing" in the sense of visually perceiving aren't ambiguously different meanings of "see" as they appear to be. (This is how I'm stating Plato's position in English; I'm certainly not making any sort of claim about the corresponding Greek words.) This shortly mutated into a view where reasoning was seen as a matter of deducing particulars from universals, although, I think, this deducing continued to be regarded as a kind of "seeing" – the *universals*, in particular.

[90]Descartes (1931, 19) writes of deductions that they must be "scrutinized by a movement of thought which is continuous and nowhere interrupted...." I'm tempted to credit Descartes with originating this view of proof, but this is a matter for historians to determine, not me: I don't know whether there were earlier expressions of the claim or (as important) whether Descartes knew of and was influenced by them. The idea, of course, may originate in the impression that inferring is like seeing.

[91]Thus: the emergence of unintended interpretations.

only possible in language-based media were decisive in turning mathematicians against the use of diagrams.[92]

There are several contemporary formalizations of Euclidean geometry. In this section, I'll focus on two: Miller's FG and Mumma's Eu.[93] I'll first give a short overview of how I regard the original Euclidean practice – this involves a bit of speculation, of course.[94] Then I'll turn to sketching how Miller and Mumma formalize the Euclidean practice; this naturally raises issues about which properties

[92]Mumma (2010, 257–259) suggests that what instead turned mathematicians against diagrams, and, specifically, Euclidean diagrams, was the generality problem.

A different challenge to my historical hypothesis is posed by something Frege (1879, 5) writes of his own version of mathematical proof: "To prevent anything intuitive [Anschauliches] from penetrating here unnoticed, I had to bend every effort to keep the chain free of gaps." This makes it sound like the gapless ideal for proof was introduced by Frege to avoid "intuition," and not that the programmatic gapless model of proof was appealing (and motivating him) in its own right, and thus was what was undermining the acceptability of "intuition." My interpretation is saved, however, if Frege's use of "anschauliches," is read – as I think it should be – not as only a label for pictures but as standing for *anything* unarticulated that enables a mathematician to "see" his way from p to q without using the intermediate steps that are presumed present in a "fully-explicit" canonical proof. "Intuition," when used to describe the mathematical insights that diagrams enable involves *several* elements that weren't clearly distinguished. One is the historically unrecognized role of intuitively effective recognition procedures directly involved in informal diagrammatic proof; but a second element (that I discuss in further detail later) is the implicit interpretations of diagrams via the reading-in and reading-off procedures.

Still another apparent challenge to my historical interpretation of Frege being primarily focused on gapless proof is that he also seems to explicitly commit himself to purity views of mathematical proof, as Detlefsen (2008, 187) illustrates with a quotation. Frege's purity condition, however, is specifically focused on the exclusion of "geometrical elements" from arithmetic. And, further, it also seems that Frege is thinking that the "geometrical elements" are ones that aren't *logical* – he writes, in the quotation that Detlefsen gives, "the task of deriving what was arithmetical by purely arithmetical means, i.e. purely logically, could not be put off." This quotation, therefore, is compatible with the gapless logical ideal of proof still being Frege's central motivation, since he doesn't commit himself to a more general form of purity of *mathematical* subject-matter. See footnote 46 for examples of philosophers and mathematicians who extoll purity goals in mathematical proofs, where those goals are driven by subject-area considerations and not logic.

[93]See Miller (2007, 2012), Mumma (2008a, b, 2010, 2019). There is ongoing controversy about Eu, and so this system is, as a result, still somewhat in flux. Miller (2012) shows that the earlier version of Eu (Mumma (2010) – and elsewhere) is inconsistent, unsound, and has several other infelicities as well. Mumma (2019) addresses the issues Miller raises by substantially modifying the system. (Although Mumma has not done so, the two systems should really be distinguished by different names.)

[94]This discussion further supplements the picture I gave of the Euclidean practice in Azzouni (2004). I wrote that paper in 1998, for the *New Trends in the History and Philosophy of Mathematics Conference* held at the University of Roskilde, Denmark, August 6–8, 1998; the conference papers, however, weren't published until 2004 – thus my neutrality vis-à-vis mathematical ontology in the paper (which is the same version I gave at the conference) instead of my by-2004 adopted nominalism. I should add that both in this chapter, and back in 1998, I'm not and wasn't attempting a suitably careful historical analysis of the Euclidean practice. See, instead, for this, Manders (1995) and Manders (2008). I'm avoiding, for example, the complex question of diagrammatic proofs-by-contradiction that he (1995) discusses with such care.

of informal rigorous proofs can be successfully transferred to their formalized cousins, and which can't – in particular if and when such formalizations replicate the phenomenological properties of the original proof practice.

There is an implicit (and sometimes explicit) assumption by those who formalize informal rigorous diagrammatic practices (e.g., Miller, Mumma, Shin, and others) that the informal reasoning practice is *captured* by these formalizations – that the formalizations reveal the reasoning implicit in the informal practice. Mumma (2008b, 256, footnote 1) writes, for example: "One should keep in mind that my criticisms [of Miller (2008)] are closely connected to my views on how my work and Miller's compare with respect to capturing Euclid's reasoning." At least insofar as the implicit reading-in and reading-off procedures are concerned, this is wrong: these informal procedures are invariably indeterminate in various respects that *cannot* be captured by any formalization – simply because formalizations decide interpretational issues that are indeterminate as far as the informal rigorous practice is concerned. Apart from this, both Millers' and Mumma's formalizations fail to replicate the informal practice as I'll indicate in what follows: specifically, these formalizations don't capture the case-structure of the original Euclidean geometry because they substitute different diagrammatic cases for Euclid's; more dramatically, they substitute a *mathematical basis* that diagrams are constructed from that's phenomenologically alien to our experience of the physical diagrams themselves. I'm not making a point about the interpretations of the Euclidean diagrams and those that Mumma and Miller give; I'm making a point about their metamathematics: how they characterize the *syntax* of the diagrams.[95] I now illustrate this.

Consider a preliminary observation about Mumma's approach. Mumma (2010) adopts Manders' (1995) distinction between exact and co-exact properties of diagrams. This (roughly) is a distinction between the metrical properties – the "exact" properties – of the figures in a diagram (the lengths of lines, the areas of triangles, the relative sizes of corresponding figures, etc.) and their topological properties, the "co-exact" properties (what regions are defined by figures, what elements are within and without figures, etc.).[96] Mumma claims – following Manders (1995) – that the original practice involved two proof systems, a diagrammatic one and a language-based one; both philosophers claim that only the co-exact properties of diagrams are used in the (strictly) diagrammatic Euclidean proof-system; the exact properties are either stipulated of the figures or proven in language-based proofs. (Mumma (2008a,

[95]Shin's approach, described in Sect. 8, is much closer to the original informal Venn-diagram practice, and retains much of the phenomenological qualities of the original informal diagrams – despite the fact that she introduces closed curves for circles (not *all* of her diagrams retain the relevant phenomenological qualities, but *some* do; I discuss this further in Sect. 12).

[96]See Manders (2008, 69) for a nice description of his exact/co-exact distinction. He adopts the term "appearance" for the set of co-exact conditions on a diagram. I'll suggest later in this section, at least as far as the experience of these diagrams is concerned, that this is misleading terminology. Even if the proof-machinery operates solely via the co-exact conditions of a Euclidean diagram, our *appreciation* of what it shows – including how much its results hold of diagrams different from the one being seen, turns on our perception of how it appears: its *geometry*.

italics his) writes: "The key observation is that Euclid's diagrams contribute to proofs *only* through their co-exact properties.") I've not determined whether this claim is true of the original proofs in Euclid – it certainly looks like it might be – but, in any case, the informal rigorous Euclidean practice isn't obviously divisible into two proof procedures in this way. Manders' observation about co-exactness, if right, is a substantial *discovery* about Euclidean proof-practice.[97]

Mumma (2010) also takes Eu to "solve" the generality problem (the question of what the diagrammatic elements in Euclidean proofs refer to), and criticizes Miller's earlier approach on the grounds that *its* solution to the generality problem increases the number of diagrams that any diagrammatic proof requires; he denies that the original practice could have operated in this way since Euclid never considers so many cases – and more importantly, couldn't do so (Mumma (2010), 263–264, footnote 4) because of their "staggering" number and because most of them "are not physically realizable."

The first point to make is to object to a mutually held assumption of Mumma and Miller. Formalizations of any informal rigorous mathematical practice always involve well-defined interpretations (models are given), and so they always "solve" the generality problem one way or another. But any such "solution" invariably deviates from the original practice because there is no solution (in the informal rigorous practice itself) to the generality problem *that original practice* faces: the scope of a diagram – the particular theorem it establishes – and, relatedly, the references of its elements are given by the reading-in and reading-out procedures, and (since the interpretations these give occur only tacitly in the practice, and even tacitly in the natural-language sentences that supply the interpretations) the informal practice is necessarily indeterminate in referential scope. I'll discuss this further in Sect. 13. I'll add now, however, that a similar generality problem occurs with respect to language-based informal rigorous mathematical proofs – when specialized terminology is involved. This is because specialized mathematical terminology can (and usually does) referentially *mutate* over time: consider the notorious case of the notion of a "function," during the course of several centuries.

The original Euclidean practice, of course, doesn't explicitly exhibit the neat distinctions *I* drew in the last section *either* – between diagrammatic transitions, on

[97] Although the insight is based on an observation commonly made through the centuries: Euclidean proofs avoid exact measurements (marked rulers) of the figures that are diagrammatically manipulated. One supporting consideration in favor of Mander's two-proof-procedure suggestion – diagrammatic and language-based – is that slash-marks (which are diagrammatic) seem to be a nineteenth-century innovation (Manders 1995, 97, footnote 14). On the other hand, another interpretational possibility is the conventionalist construal of the diagrams (see the definition of the "conventional option" shortly below) coupled with a meta-diagrammatic language in which *all* the proofs take place. According to this interpretation, there are no "diagrammatic proofs": rather, in the meta-diagrammatic language, one has the resources to "point to" aspects of diagrams – both exact and co-exact features. (Diagrams are, as it were, "quoted" in the metalanguage.) I entertain this interpretational hypothesis in Azzouni (2004); I claim that it wasn't available to ancient Greeks as a possible interpretation of their Euclidean practice – even if it was *true* of that practice – because of their implicit philosophical views about (correspondence) truth.)

the one hand, and the reading-in and reading-off recognition procedures, on the other. This is hardly surprising since, in my description above, the reading-in and reading-off procedures (which supply interpretations for the diagrams and their elements) are sharply distinguished from the syntactic obtainability relations among diagrams; by contrast, talk of diagrammatic elements and talk of the interpretations of those elements are, generally, not distinguished in Euclid's *Elements* – less charitably put, they're systematically *confused*. Consider the following definitions:

1. A **point** is that which has no part.
2. A **line** is breadthless length.
3. The extremities of a line are points.
4. A **straight line** is a line which lies evenly with the points on itself.
5. A **surface** is that which has length and breadth only.[98]

These (with the exception of (3)) purportedly describe the *mathematical* objects of study. Items which have no parts, or are breadthless, or have length and breadth only, can't be the actual diagrammatic items we draw and so they're *abstracta.* Call this *the Platonic option.* This option, that is, treats diagrammatic elements as referring to abstracta, and it treats the definitions as attempting to determinately *fix* the referents of these pictorial elements, drawn points, lines, etc. that appear in diagrams. For those who take diagrams to refer to Platonic entities, these definitions must fail as a result (worse, they're pointless): one cannot in this way provide interpretations; one can, at best, restrict the interpretations of primitive terms via axioms, which interlock those interpretations to one another in various ways.

This problem is quite general. The primitive terms – and now I use the phrase "primitive term" broadly to cover both the words in language-based sentences and the pictorial elements in diagrams – cannot be defined in this way, as we now know. Where referents are physically available, they can be indicated by language-exit rules (gestures and the like); when they aren't, these referents can only be fixed by axioms and inference rules (diagrammatic or otherwise) that constrain – but generally can't determine uniquely – possible interpretations. And this exhausts what can be done *within* a language. Within the confines of another language (a "metalanguage") a model can be given – this includes explicit interpretations of the primitive terms, which are given with the model. The *generality problem*, in the context of an informal rigorous mathematical practice, is the problem of how the primitive terms of that practice are given interpretations.

A second option for characterizing the role of these definitions takes them to be meant to describe diagrammatic items in a way that indicates *some* of the conventions governing the drawing of diagrams; and how, as a result, we're to understand such items. In this case the subsequent proofs involve claims – in natural languages – about the conventionalized diagrams themselves. I'll call this *the conventional*

[98]Heath (1956, 153); his numbering, his boldface.

option.[99] On this view, diagrammatic elements are (exactly) like chess pieces; it's *understood*, that when a chess piece is moved, it's subsequently located in one and only one square even if it's been placed sloppily on the board (or even if the board is designed so badly that it can be hard to tell where a piece is). This involves understanding *physical* chess pieces conventionally – by imposing laws of movement on them via stipulations: these are laws of movement they'll otherwise violate. It *isn't* – notice – a matter of treating chess pieces as standing in for (or referring to) chess-piece abstracta that actually have the properties that chess pieces are conventionally treated as having (that's a version of the Platonic option – again).

In any case, the *Elements* both describes diagrammatic syntactic-transitions of diagrams in terms of what can be done to a diagram with respect to the above visual points, lines, etc. but runs this together with assumptions or claims about the interpretations of those visual items when they appear in diagrams, as in the postulates:

1. To draw a straight line from any point to any point
2. To produce a finite straight line continuously in a straight line
3. To describe a circle with any center and distance.[100]

Although naked infinitives are hard to interpret (and although I'm relying on Heath's *translation* when I say this and interpret the words as I do in the next few lines – assuming, that is, that what I say is true of the ancient Greek), the postulates (1) and (2) seem to describe permissions for possible actions that can be taken on diagrams; they don't seem to be characterizations of what the diagrammatic items, points and lines, refer to (i.e., to the mathematical objects: points and lines); they aren't, to put it another way, existence claims about what the elements of the diagrams depict. This "action-interpretation" of these postulates follows from the use of the phrases "to draw" and "to produce." On the other hand, (3), because of the word, "describe," seems, in addition to apparently describing an admissible action that may be taken to construct a diagram, to be offering an interpretation of what that diagram (of a drawn circle with a particular center and distance) is to stand for. According to the Platonic option, these definitions together with the postulates exhibit various use/mention confusions; according to the conventional option, there aren't such confusions (except insofar as interpretational issues are intruding in what otherwise would be a pure description of the syntactic conventions of a diagrammatic practice, and the nature of the syntactic idealizations it involves).

Recalibrating the informal Euclidean practice so that it accords with the device view (as it was described in Sect. 9) entails very little by way of changes to the look

[99]Leibniz writes: "[Geometrical] figures must also be regarded as characters, for the circle described on paper is not a true circle and need not be; it is enough that we take it for a circle" (quoted by Manders (1995, 80). See Azzouni (2004) for further discussion of these two options and the evidence for why the Platonic option was eventually universally adopted despite the *Elements* retaining aspects of a conventional diagrammatic-practice, for example, in the statements of some postulates, as I indicate shortly.

[100]Heath (1956, 154); his numbering, his spelling.

of the actual practice. One needs to, first, distinguish the diagrammatic/notational elements from the interpretations of those elements, second, to describe (in an interpretation-free way) the syntactic-transitions embodied in that practice, and third, to indicate the interpretations ~~of the elements~~ of the diagrammatic/notational elements. That's not changing much.

I claim this is generally the case with the device view – in contrast to current formalizations of informal mathematical proofs, especially those of the Euclidean practice. I feel safe, thus, in regarding the device view as providing a theoretical *schematic* that enables *descriptions* of ordinary informal mathematical practices – diagrammatic and otherwise – despite its drawing a sharp distinction between actions on elements (of diagrams or languages) and how those elements are interpreted, a distinction which isn't exhibited directly in those practices themselves (as the discussion of the Euclidean case indicates). Reformulating an informal mathematical practice according to the device view should leave all the proof-theoretic flaws (if any) in the practice intact. For example, if (as many have illustrated) Euclidean diagrams can be ambiguous or misleading, the presence of these ambiguous or misleading diagrams should survive the practice being recast according to the device view. Second, it should also leave intact interpretational flaws – for example, failures to (fully) solve the generality problem. In this sense, if only because formalizations are consistent and determinate, they impose normative constraints on the informal practice that the practice itself needn't exhibit.[101]

The upshot: Attempts to formalize informal rigorous mathematical proof practices pose challenges that go quite beyond what's required to frame that practice correctly and according to the device view. In particular, with respect to diagrammatic reasoning, since (almost invariably) there is relevant reasoning occurring in the natural language apart from the syntactic transitions from diagram to diagram, that reasoning must be formalized as well. In addition, the reading-in and reading-out procedures must be treated both as supplying interpretations – so they are implicit guides (for example) of the model theory that is to be used to interpret the formal system – and as providing crucial inferential structure between the mathematical content expressed by ordinary-language sentences and that expressed by the diagrams licensed by the algorithmic devices.

[101] I'm not saying, of course, that an informal mathematical practice need be flawed in various ways; but whether it is or not is usually clear to practitioners at the time that the practice is living, and not because they're aware of one or another formalization that they're comparing the informal practice with. This is true, for example, of the ancient debates over how many diagrams should appear in a diagrammatic proof, and whether some diagrams are "ambiguous." This is similarly true of the debates over the acceptable manipulations of the notation for infinite series in the eighteenth and nineteenth centuries. It's important to realize that these debates are (often) resolved in the informal practice itself by notational reforms of various sorts along with changes in the rules for manipulating the syntax of an algorithmic device: these changes can be characterized in ways that don't require discussions of formalizations. (There is an exception, of course, once formalizations themselves become part of the mathematical toolkit among ordinary mathematicians; then a formalization may itself replace the previous informal rigorous mathematical practice – at least in some respects.)

Miller and Mumma formalize ~~the~~ Euclid's *diagrammatic* practice in (quite distinct) formal diagrammatic systems. That means, to begin with (as I indicated earlier), that in both cases the physical diagrams themselves are identified with certain kinds of mathematical objects – certain kinds of two-dimensional abstracta – that can be (intuitive-effectively) recognized as having certain properties and not others. Right at this point both formalizations deviate in a significant way from the Euclidean (book I) practice; this – all by itself – faults claims that either resulting formal system "captures" the reasoning embodied in the original practice.

First, recall that the figures in the original Euclidean practice are constructed *only* by using straightedge and compass.[102] Actual physical drawings – I stress again – can be identified (preliminarily to representing an informal practice formally) with mathematical objects of any sort; the mathematical objects chosen will vary, of course, in how adequately they capture the original practice. But to identify physical drawings with any particular set of abstracta involves decisions of certain sorts that bear directly both on the question of how the figures in the drawings are being interpreted, and specifically, on how those figures are being interpreted to pick out certain items in the formal domain *and not others.* The geometrical properties of the actually drawn figures don't all by themselves dictate an answer to what abstracta they should be identified with. For example, we may choose to identify the physically drawn lines with various nonlinear curves of all sorts – not just straight lines – on the grounds that such physically drawn lines (especially when drawn badly, as they can be) *really do* approximate nonlinear curves of various sorts more closely than straight lines.[103] The same point can be made about the curves generated using a compass – they needn't be literally circular but only approximately circular, and we may take them to better approximate something noncircular. Miller's FA and Mumma Eu are formal systems based on an interpretation of Euclid's figures that identifies them syntactically with what at first glance appear to be quite broad classes of mathematical objects.

Miller, for example, defines the notion of a "nicely well-formed diagram" in terms of four primitive symbols (compare this to Shin (1994) – the strategy is the same one): frames, dots, unbroken lines, dash-lines, and slash-marks.[104] Any diagram

[102]In addition, points can be placed into the diagram – pretty much anywhere you want to drop them (subject, of course, to resolution considerations that prevent points from being too close to one another). This doesn't falsify the claim I just made about *figures.* In the Euclidean context, you *can't* drop a bunch of points in succession and use *them* to design a nonlinear/noncircular curve (that goes through all of them). The automatic existence of various (infinitely many) one-dimensional curves going through a set of points goes quite beyond the Euclidean system. Any one-dimensional curve that connects *any* two points *must result* from the operation of straightedge or compass.

[103]Miller (2007, 3) stresses right at the beginning of his study, that the "different kinds of line segments" that appear in Euclidean diagrams "do not have to really be straight."

[104]"Dash lines," are broken lines, for example, -----, as opposed to unbroken lines, for example, ——. I'm taking a few expository shortcut liberties in my description of Miller's approach, for example, "marked diagrams" are defined, not by the introduction of slash-marks as part of the primitives for diagrams, but as an extension of diagram equivalence to "marked diagrams" (see Miller 2007, 31).

occurs within a frame and it's composed of a finite number of dots, unbroken lines, and dash-lines – the unbroken lines represent Euclidean lines (although they needn't be straight) and the dash-lines represent arcs of circles (although they needn't be circular arcs). The definition of nicely well-formed diagrams includes a list of conditions that force unbroken lines and dotted lines to behave roughly (that is, topologically) like Euclidean lines and circles. (For example, one condition is that two "circles" intersect no more than twice.) To replicate Euclidean reasoning, these diagrams need these and additional characterizations. The result is that, again roughly, Miller's "nicely well-formed" diagrams can be associated with the original Euclidean diagrams. Miller then designs the actual proof-system in terms of topologically equivalent diagrams.[105]

Mumma's Eu, on the other hand, proceeds in a strikingly different way.[106] Instead of correlating Euclid's physical diagrams with a domain of rectangles containing arbitrarily nonlinear curves (except for obeying certain conditions, as mentioned above) and dashed curves (also, generally, noncircular, except for obeying certain conditions), etc., Mumma's domains are finite rectangles defined by square arrays of evenly spaced hollow ("○") and solid ("●") dots. The dots can be connected by lines, and some of those lines represent Euclidean lines; the finite lines, however, that are part of various convex polygons (together) represent circles. Like Miller, Mumma starts with an initial characterization of diagrams (call them "pre-diagrams") but then, like Miller does with *his* pre-diagrams, Mumma places an equivalence relation on them. *A* completion (nonunique) of a diagram D is one, D*, that's like D except that it has additional evenly spaced hollow and solid dots – enough so that every intersection point between the drawn lines and rays occurs as a dot. A definition of equivalent diagrams in terms of completions of them is then given by Mumma.

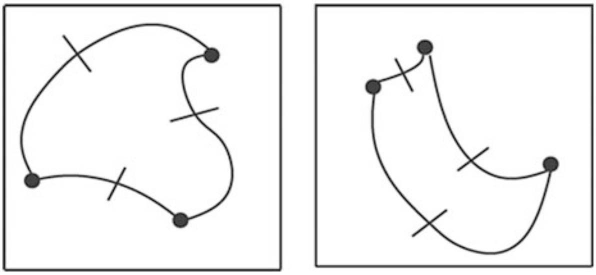
The result still deviates greatly from the informal rigorous diagrammatic practice – the experience of working in Mumma's system, specifically, deviates phenomenologically from Euclid's original diagrammatic practice. Why? For two reasons. The first is relatively insignificant: the diagrams we work with in Mumma's system involve visually recognized lines and polygons – not circles. Circles are conventionally represented by polygons. But the important difference should be visible, even given my rather sketchy description of the approach: the visibility of the array of hollow (and solid) dots that diagrams occur on. This transforms proofs in the Euclidean context – where diagrams are built upon one another by dropping points arbitrarily into what appears to be a manifold, and by drawing lines between points,

[105]This equivalence relation is technically involved – using a mapping between nicely well-formed diagrams and planar graphs. In terms of this mapping, two diagrams are defined as equivalent if they have the same corresponding graph structure. The different cases that must be considered in one of Miller's proofs then correspond to how many diagrams with differing graph structures become involved. It's worth pointing out, when comparing the reasoning in Miller's system with that of the original Euclid that the computer (which carries out the reasoning in Miller's system) works not with diagrams but instead with planar graphs. I make something of this shortly.

[106]I'm here following Mumma's (2019), where he lays out the modified system nicely.

and otherwise, by drawing lines arbitrarily and by drawing circles arbitrarily as well – into something experientially very different. The array-structure of the Euclidean context – in ordinary Euclidean proofs – as I said, isn't directly visible; it's required only by virtue of the incapacity of the practitioners to resolve arbitrarily small differences between drawn figures.

This suggests that Miller's identification of Euclidean figures with curves results in something phenomenologically much closer to the informal Euclidean approach than Mumma's is: it's one, recall, where figures are (apart from certain conditions) are arbitrarily nonlinear and noncircular – despite being "lines" and "circles." The manifold-flavor, that is, of the original diagrammatic proofs seems preserved. Nevertheless, Miller's approach isn't phenomenologically closer to Euclid's original practice than Mumma's reincarnation of it is. One reason (there are several) is because of something about the original Euclidean practice that's right on the surface. As Mumma (2008b, 259) points out in his review of Miller (2007), rather strange – from the Euclidean perspective – diagrams can appear in Millers' system. Mumma's example (his Fig. 3) – of "two ways to represent an equilateral triangle in **FG**" – follows:



Mumma (2008b, 262), further criticizing Millers' approach, writes (italics and boldface his):

> . . . when a case-heavy **FG** formalization is laid beside Euclid's original version, the original does not appear deficient. Rather, the multitude of cases generated by the rules of FG appear excessive. The geometric differences recorded by a case-branching often do not seem material to the issue the proof decides.

This criticism, however, largely misses the point. It primarily focuses on the sheer number of cases; but a formalization captures the phenomenology of an informal rigorous mathematical practice not by (even somewhat closely) matching the *number* of cases that appear informally; what's important is what *kind* of cases are involved: do the cases that are required by a formalization of Euclid's approach track the spirit of why the particular diagrams in the original practice come up? More important is this consideration: does the phenomenology of the original diagrammatic practice match (at least to some extent) the phenomenology of the formalized practice?

Mumma's criticism focuses on the fact that Millers' approach multiplies the number of cases that a Euclidean proof requires in a way that goes beyond the cases that the Euclidean approach seems to consider – even when that approach is supplemented with additional diagrams, as occurred later in the geometrical tradition. But the way that Millers' approach deviates from the Euclidean approach, although indicated by Mumma's Fig. 3, isn't stressed by him. This is that it treats the geometric information a diagram supplies as irrelevant to the proofs. It may be true that the geometric information a diagram supplies is irrelevant to the proof procedures that Euclidean geometry licenses but it's surely not irrelevant to how we come to *understand* these proofs because of how the diagrams *look.* Miller's approach is designed to facilitate a certain computer implementation, CDEG.[107] But the computer doesn't manipulate Miller's *diagrams*; it manipulates the *graphs* that the diagrams determine nonuniquely. Thus, the proof-practice Miller's formalization licenses amounts to this. A Euclidean diagram D is represented by a Miller diagram, M – more accurately, D is represented by a finite set of Miller diagrams. M, in turn, isn't itself manipulated according to a proof procedure; rather, M is used to generate a graph G, and it's G that's manipulated proof-theoretically. In turn, then, G is either transformed into programming code (which can be printed out), or into a canonical Miller diagram M*. This, it should be clear, is very far from the original Euclidean practice. In particular, Miller's approach cannot be used to explain what it is about Euclidean diagrams that makes the results they license so "intuitively" *comprehensible*.

In fact, neither Miller's nor Mumma's approach has even a hope of managing the replication of the understanding/insight that the Euclidean practice provides to its practitioners; this is because of the initial way both approaches mathematicize Euclidean diagrams. Return to the original practice for a moment. Three and only three sorts of primitive diagrammatic elements show up: points, lines, and arcs of circles. *That's it.* Diagrammatically speaking, there are no other entities that appear *in* diagrams. Of course it's true that what's drawn (physically) can unpleasantly deviate from what we (conventionally) take points, lines, and circles to *resemble.* But the expected (and allowable) response, if the intention is to formalize the practice in a way that captures the original reasoning, isn't to formalize the original proof-processes where badly drawn figures that deviate from what looks like circles and lines *become crucial to the cases that a proof must accommodate.* Rather, the response is: "that's drawn badly; I can't see what's going on; can you make a better one?"[108]

This means that, in a formalization of the *syntax* of a formal Euclidean diagrammatic system, the appropriate domain (the appropriate *alphabet*, as it were) should be one in which *only* circles, lines, rays, and points appear, and where these items are restricted in their sizes (and how they interact with one another) so that they are

[107] I'm here relying on Miller (2007, Sect. 3.5).

[108] See Manders (2008, 71) on how the original Euclidean practice tried to avoid "sensitive situations." This dovetails with what I've earlier called exclusion idealizations (defined in Sect. 2).

(by eye) surveyable.[109] The formalization approach I'm sketching treats deviations from circularity and linearity not as diagrammatic phenomena to be captured or replicated in the syntax of the resulting formalization but instead, as it was in the original practice, as items to be *excluded*.[110] The cost of not doing this – as I've indicated – is to "replicate" the informal practice by a formalization that doesn't respect the phenomenological qualities of the original practice – this matters because those phenomenological qualities are what underlie our experiences of understanding proofs, specifically the "aha"-epiphany. I now turn to an explicit discussion of this and its relation to the generality problem.

12 Phenomenologically Faithful Embeddings of Algorithmic Devices in Formal Algorithmic Systems

The phenomenological flavor of a proof is a somewhat elusive matter (and so it's dangerous to venture saying anything about it). Nevertheless, it's, importantly, what I think philosophers are tracking when they describe certain proofs as explanatory and others as unexplanatory; the same phenomena – the phenomenology of a proof (how the proof is experienced) – is also being tracked by mathematicians when they describe certain proofs (or, as commonly, certain subject areas – with their typical proofs) as beautiful or ugly. I'm not going to dwell on issues about explanation in mathematical proof – this is too tangled and complex to take up even in a paper this big. But I do want to describe the phenomenological aspects of informal proofs enough to give a rough characterization of how – and to what extent – formalizations capture these qualities of informal proof practices.

Recall the initial discussion of the "aha"-epiphany described by Feferman (2012) in Sect. 3. Recall, for that matter, the constraint on derivational accounts, *Phenomenology*, that such accounts must explain why formal transcriptions of informal

[109]This characterization of the syntactic domain is, of course, subject to the persistence and exclusion idealizations given in Sects. 2 and 4, and later. Despite appearances (i.e., the phrase "by eye"), I'm describing a purely mathematical exclusion condition on circles, lines, rays, and points: that is, we translate an empirical fact about average-resolution capacities into numerical lower-limit conditions on admissible circles, lines, and rays.

[110]I don't know of, at the moment, any formalization of the original Euclidean practice that approaches it the way I'm suggesting here. I don't see any reason, however (offhand), to rule it out. I'm gambling in saying this, however, because an approach to formalization can always be ruled out by unexpected technical obstacles. Illustration: The original Venn diagrammatic practice used overlapping circles. In order to generalize the approach to finite collections of sets (recall Sect. 8, specifically footnote 61), Shin – relying on a result due to Polythress and Sun (1972) – replaces the circles of the original practice with closed curves. I should add that I don't claim that the resulting formal system I'm sketching an approach to won't be heterogeneous, rather than purely diagrammatic. It might be, and it might respect some descendant or other of Manders' exact/co-exact distinction. But the informal practice underdetermines exactly what belongs to the language-based part of the formal system and what belongs to the diagrammatic part – for example, exact-numerical identifications between figures and their parts might be indicated diagrammatically by slash-mark notation or by language-based characterizations accompanying the diagrams.

rigorous mathematical proofs don't replicate that experience. The first point is that, although formalizations are often described as lacking the epistemic "aha, now I see why this is true" epiphany that's sometimes experienced with an informal proof, this simply isn't true of formalizations, *in general*: they needn't be so lacking. Other philosophers who contrast informal rigorous mathematical proof with formal derivation disagree with me.

Giaquinto (2008a, 27) writes, for example, ". . .there is a reason to avoid going formal: in a formalized version of a proof, the original intuitive line of thought is liable to be obscured by a multitude of minute steps." It should be clear from the foregoing that, even though it's a requirement on a formal system that its syntax be completely given, it's not a requirement that the syntactic transitions licensed by the formal system involve minutia.[111]

Rav (2007, 315, and elsewhere) suggests that formalizations *must* fall short in relation to their informal cousins because built into the appreciation of informal rigorous proofs and not their formalized cousins are the meanings of the terms used in the sentences of that sort of proof. He writes: "I hold that mathematicians' manner of reasoning and inferences are based on *meanings* and an informal notion of truth that a formal deduction calculus cannot capture" (italics his).[112] Rav (1999, 12) some years earlier puts his claim about the experiential distance between informal proofs and formalizations of them far more melodramatically (italics and spelling his):

> Once we have crossed the *Hilbert Bridge* into the land of meaningless symbols, we find ourselves on the shuffleboard of symbol manipulations, and as these symbols do not encode meanings, we cannot return *via* the Hilbert Bridge and restore meanings on the basis of a sequence of symbols representing formal derivations. After all, it is the very purpose of formalisation to squeeze out the sap of meanings in order not to blur focusing only on the logico-structural properties of proofs. Meanings are now shifted to the metalanguage, as is well known.

[111]This is an impression many have from a familiarity with formal language-based systems that involve, say, a lot of parentheses, or where the transitivity and symmetry of "&," and other connectives aren't built into the syntactic recognition procedures. All that's required of a formal system, however, is that its syntactic transitions be effectively recognizable. Thus, a formal system can be substantially modified by "shortcuts" and abbreviations," that make it far more user-friendly – as long as these are open to intuitively effective recognition procedures. I should add that in past work I've also contrasted formal and informal proof in just this way (mea culpa).

[112]A position in the same neighborhood is expressed by Larvor (2012, 723) when he writes: "The benefit of viewing inference as action is that we can see how the *subject-matter* of informal arguments shapes and contributes to inferences" (italics mine), and by De Toffoli and Giardino (2015, 332–333) who explicitly follow Larvor in this, writing: "In Rolfsen's proof, we saw that among the permissible actions on the pictures are continuous transformations. These are part of the background material in the sense that any topologist knows immediately that these transformations *can be interpreted* in terms of homeomorphisms. The validity is thus based on the 'practice' . . ." (again, italics mine). Notice that De Toffoli and Giardino describe "continuous transformations" as permissible actions on *pictures*. This is a confusion.

This misleads how much informal phenomenology – specifically the experience of *meanings* – can occur in a formal setting. First point: meanings aren't, in any case, shifted to the metalanguage, as Rav writes – rather the meanings of the object language expressions are *given in terms of* the metalanguage (recall the discussion of this in Sect. 10). Second: In any case, our ability to psychologically invest formal terms with meanings should already be obvious from the centuries-long practice mathematicians have had with the routine introduction of new terminology into mathematical settings. Indeed, the point should be clear from the ubiquitous practice already in place with ordinary words – we (collectively) both modify the meanings of words and phrases already in public possession and routinely introduce new terms and phrases by indicating their meanings. What's striking, in addition, is that we can have the experience that a meaningful expression has been introduced (and that we grasp it) without being given a definition but just an illustration or two – typically all that's presented in dictionaries. Many of us continue to learn what we experience as new meaningful expressions all our lives.[113] Nothing prevents us doing the same thing to new terminology that's been introduced in a formal setting. (Indeed, we *regularly* do this; Rav's remark, quoted above about Hilbert's Bridge is especially surprising in light of the fact that, even when focusing on syntax, it's rather hard not to interpret certain logical expressions – notably "&" and "$\exists x$," among others – as having *fixed* meanings derived from natural language.)

At this juncture, it's natural to attempt a preliminary characterization of the phenomenological closeness of a formal system to an informal practice that it's taken to formalize. Consider an algorithmic device **A** (diagrammatic or language-based) that's employed in an informal rigorous mathematical practice. Describe that device as *embedded* in a formalization **F** if

(i) There is one-to-one mapping M of the elements of **A** to the syntactic elements of **F**. (M doesn't have to be "onto" – **F** can, and usually does, have additional elements.)
(ii) Given any two elements D_1 and D_2, of **A**, given an intuitively effective-recognizable obtainability relation $\vdash_{\mathbf{A}}$, for that practice, and given that syntactic transitions $\vdash_{\mathbf{F}}$ for **F** are defined, then if $D_1 \vdash_{\mathbf{A}} D_2$, we have $M(D_1) \vdash_{\mathbf{F}} M(D_2)$.

[113] I discuss our experience of meaning, in natural languages in Azzouni (2013b). Especially pertinent is a common experience that we have – with respect to certain expressions – of their being meaningful even though we don't know what their meanings are (Azzouni (2013b), 93). It's true that there are cases where we cannot experientially dislodge a meaning that we experience a word to have and replace it with another (e.g., "cold" for "hot") as I there discuss (Azzouni (2013b, Sect. 1.4). Nevertheless, we're quite good at introducing new words and, in specific contexts, like mathematical ones, of even setting aside the meanings that words apparently have – for example, stripping away the impression that "and" has a temporal dimension (e.g., "John brushed his teeth and got out of bed"), as we routinely do in mathematics.

So, given a formalization **F** and an algorithmic device **A**, call the mapping *M* that obeys (i) and (ii) an *embedding* of **A** into **F**. We say, further, that *M* is *phenomenologically faithful* to **A** if the experience of the inference, accompanying a syntactic transition $D_1 \vdash_{\mathbf{A}} D_2$, is relatively similar to the experience of the inference accompanying the syntactic transition $M(D_1) \vdash_{\mathbf{F}} M(D_2)$.

Why have I introduced the qualification "relatively similar" into the definition of "phenomenologically faithful"? Because (I take it) if we change the notation an inference is couched in – even minimally – we, as a result, change the experience at least somewhat: $A = B$, therefore, $B = A$ is an experience of an inference that feels (slightly) different from $x = y$, therefore, $y = x$. There is, however, a bit more to say about the experience of inference – and the psychological mechanisms – that will give more content to the above characterization of "phenomenologically faithful." I'll turn to that in the next section.

I do need, at the outset, to stress two important aspects of phenomenological-faithful embeddings of algorithmic devices into formalizations. One is that the process of so embedding these devices invariably involves the inclusion of additional proof procedures (in the formalization) that don't possess the phenomenology of the original practice. This is the case no matter how carefully we choose the diagrammatic abstracta to match the original physical diagrams; and *this* is because the phenomenology itself is often fragmentary in its reach (our "aha"-epiphany, e.g., occurs only with fragmentary aspects of any informal proof-practice – let alone the extension of such a practice that results from formalization). A second point is that because we sometimes have free-ride experiences with respect to an informal rigorous proof, the actual algorithmic device involved in such a proof isn't itself – even informally – characterized. The algorithmic devices involved in such visual proofs remain tacit even in the informal mathematical practice. I'll round out this section by giving examples of both of these cases.

Start by recalling the original Venn diagrams, and comparing them to Shin's (1994) formalizations. Our "aha"-epiphany with the original diagrams – immediately seeing ("free riding") certain implications only occurs when a small number of circles are involved (about three). Furthermore, the experience is sensitive to circularity: change the figures involved into different (weirdly shaped) closed curves and the experience vanishes (one has to work out the implication visually – take a few moments to "see it"). That is, the immediate-recognition property doesn't work with other sorts of figures, as the reader can easily determine. As I'll discuss in the next section, this is because the particular "inference packages" we employ to recognize the implications of Venn diagrams are (perceptually) content-sensitive: they work only with certain sorts of figures, and visual patterns, and not others. Thus, any formalization – especially an ambitious formalization (e.g., one, like Shin's, which is designed to capture, say, full monadic reasoning) – and even one that's otherwise diagrammatic, will replicate the phenomenological experience we have with the original diagrams only with a few of the formalized diagrams.[114]

I'll now illustrate the second kind of case with the diagrammatic proof of the "mutilated chessboard." Here's a description of it due to Black (1946, 157) that I've borrowed from Tanswell (2015, 305):

> An ordinary chess board has had two squares – one at each end of a diagonal – removed. There is on hand a supply of 31 dominos, each of which is large enough to cover exactly two adjacent squares of the board. Is it possible to lay the dominos on the mutilated chess board in such a manner as to cover it completely?

And here's the solution, quoted from Gardner (1988) – but I'm again taking the quote from Tanswell (2015, 305):

> It is impossible . . . and the proof is easy. The two diagonally opposite corners are the same color. Therefore their removal leaves a board with two more squares of one color than of the other. Each domino covers two squares of opposite color, since only opposite colors are adjacent. After you have covered 60 squares with 30 dominos, you are left with two uncovered squares of the same color. These two cannot be adjacent, therefore they cannot be covered by the last domino.

This proof and many other diagrammatic proofs[115] do *not* occur against an informal proof-theoretic background the way, for example, that the ancient Euclidean diagrammatic practice (and the Aristotelian syllogistic tradition) do. But it's not difficult to see what such an informal proof procedure would look like – it's not difficult to *create* one. An informal rigorous mathematical two-color-domino practice (on an ordinary chess board) would involve the placing of domino pieces on the board. The results – some immediately seen and others shown – would be various generalizations about what sorts of patterns of domino placements on the chess board are possible, and which aren't. Another family of results – some immediately seen and some not – is about which initial placements (of a few dominos) can be "completed," into a full covering of the board with dominos (with certain color patterns) and which can't. Lastly, one can consider results of domino-tiling various mutilated chess boards, as in Black's posed problem. In turn, these just-described informal practices can be embedded in one or another formal diagrammatic system that's phenomenologically faithful to these posited informal-proof practices. These can – Hilbert's

[114]The problem is already visible in informal practices. That certain proof procedures can be characterized quite broadly using persistence idealizations runs well beyond when such proofs are convincing and understandable. (So my use of "intuitive," in Sect. 4 and elsewhere throughout this paper, in, for example, "intuitively effective function" has to be understood as a somewhat specialized usage of the word – introduced because of Turing's own use of the word (see footnote 28).) Part of the phenomena here is described by Manders (1995, 107, and elsewhere) as "appearance control."

[115]for example, the proof in footnote 118

thesis tells us this – be otherwise formalized in various language-based formalizations.[116]

13 Inference Packages (Embodied Algorithms) and Phenomenological Faithfulness

It's undeniable that the use of certain diagrams enormously facilitates our abilities to reason to and from various mathematical results these diagrams are taken to depict. Related to this is that diagrams also seem to enable us to understand the reasoning involved, and often to provide an understanding of the content of the results and assumptions (that the results follow from) that language-based formal transcriptions of that reasoning fail to provide. One approach to explaining this focuses on the apparent similarity between the elements of certain diagrams and what they're diagrams of – what I earlier called a "resemblance interpretation." If A pictures B, then we can use A to understand aspects of B. A photograph, for example, or a drawing, replicates certain aspects of the thing photographed or drawn, and by directly considering the photograph and the drawing, we can understand (or realize) something about the things themselves.

Despite the temptation to centralize visual similarity – resemblance – in an explanation of the just-described value of diagrammatic proof, pressing a resemblance interpretation looks bizarre almost on the surface; and it has led to serious mistakes among professional mathematicians when evaluating the value of diagrammatic proofs; these mistakes continue to affect contemporary philosophical discussions of diagrams. To start with, consider the case of Euclidean figures: points and lines. Given what these are supposed to be, it isn't even *sensible* to describe them as "resembling" physically drawn points and lines. Apart from the fact that no physically drawn figure lacks the dimensions that the mathematical objects, points and lines, officially lack, there is the related problem that the physical items have – geometrically speaking – *the wrong* "*shapes*." Notice the point. This isn't a criticism of one or another theory of "abstraction" as a theory of understanding – that we

[116]Tanswell (2015) and Tanswell (forthcoming) press against derivation accounts of informal rigorous mathematical proof (Sect. 1) his "underdetermination" problem – that such accounts have the problem that attempts to capture the epistemic properties of informal proof via formalizations face the fact that there are equally good (equally close) formalizations that nevertheless involve drastically different mathematical ideology. Although derivational accounts (Sects. 1 and 3) have been rejected here, Tanswell (forthcoming) frames the underdetermination problem in a way that threatens a different claim that I've made in this paper – that in general, phenomenologically faithful formalizations for informal rigorous mathematical practices exist. He writes of the mutilated chess board proof, in particular, that it "is not obviously algorithmically checkable, so if it is claimed to be then we would like to see the algorithm that does so given explicitly . . .." His suggestion is that if it isn't, then any embedding of the proof in an algorithmic system will face underdetermination. The remark that a system of recognizing patterns of two-color dominos on what amounts to a finite array isn't "obviously algorithmic" is puzzling: it looks exactly like other cases of intuitively effective procedures that – via Church's thesis – are Turing computable.

understand what a line is, say, by abstracting a certain approximate aspect of drawn lines: pure "straight-line directionality," or whatever. It isn't even to criticize a view of understanding that claims we understand mathematical straight lines via "approximations" of them by visually thin lines. Some theory or other like this about how we understand points and lines might be true. The point, rather, is the resulting objects – *abstracta* – even if we can *understand* what they are supposed to be, aren't *seeable.* Despite whatever value a theory of abstraction has as a theory of understanding, it has none as an explanation of the value of diagrams of lines and points taken as "visually resembling" actual points and lines.

Similar complaints can be made about the diagrams that appear in other diagrammatic practices. Shin's diagrams don't visually resemble (in any sense) classes or their relationships to one another: two circles that overlap don't *visually resemble* the relationships two sets have when they have elements in common – this isn't even true if the overlapping region is shaded. Finally, recall (footnote 112, Sect. 12) that De Toffoli and Giardino (2015, 332–333) treat pictures and what's pictured as so similar that they feel they can describe diagrammatic actions on pictures as continuous transformations.

Mistaken resemblance interpretations are very old ones, and their implicit operation is one of the central factors behind the late nineteenth-century repudiation of visual methods of proof. Unaware of the diagrammatic constraints on Euclidean diagrams that were necessary to making Euclidean diagrammatic proof coherent, pictures of functions – printed drawings of curves in books and articles (let's be accurate about what the diagrams in question *were*) were treated as depicting functions, *of any sort*, by sheer visual resemblance. More accurately, drawings of functions – ink on paper – were treated as yielding "intuitions." Intuitions, so called, can only be understood in this case as *visual imaginings* of the curves in question – the scope of those visual imaginings supposedly telling us what sorts of curves are and aren't possible. The implicit idea was this: diagrams elicit our visual faculties. But when we so-visualize functions, we can get it very wrong. This, however, is the wrong picture of diagrams of functions (or the diagrams of any mathematical object, really). We cannot "visualize" functions any more than we can "visualize" lines – and for the same reason. Functions – if characterized geometrically – are one-dimensional[117]: that means we *can't* and *don't* visualize *them*.

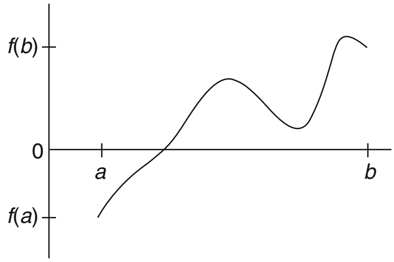
Giaquinto (2008a, 35) channels this nineteenth-century mistake about the role of visualization in mathematics. He writes:

> When it comes to analysis mathematicians of the nineteenth and twentieth centuries were right to hold that visual thinking rarely delivers knowledge Visualizing cannot reveal what happens in the tail of an infinite process. So visual thinking is unreliable for situations in which limits are involved, and so it is not a means of discovery in those situations, let alone a means of proof.

[117]Even "space-filling" curves are, nevertheless, one-dimensional!

The letter of this remark is correct: We cannot design a diagrammatic proof procedure that will handle the tails of infinite processes.[118]

But this remark occurs nested (as was also true of nineteenth- and twentieth-century objections to diagrammatic proofs by mathematicians) among misapprehensions about (informal) diagrammatic proof. Consider this well-known informal diagrammatic proof of the intermediate zero value theorem:

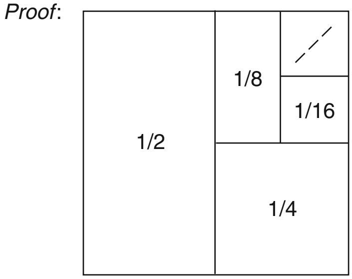


Giaquinto (2008a, 35) criticizes Brown (1999) for suggesting that "reflection on the diagram [above] suffices to prove the Intermediate Zero Theorem." Giaquinto (2008a, 36) notes there are many functions visually indistinguishable from a continuous smooth function defined on the reals, as above, of which this theorem is false.

But speaking this way of "visual thinking" (it should be clear) overlooks that an informal rigorous diagrammatic practice that treats diagrams of functions, as above, as referring to (and only referring to) smooth continuous curves, *does show* the intermediate zero theorem for that class of curves.[119] Diagrammatic practices must be curtailed sharply in what mathematical items are to be treated as the referents of their elements, and thus, what generalizations they can be taken to have shown. This

[118]Actually, no. Giaquinto's remark, about the inability of diagrammatic reasoning to capture "the tails of infinite processes," is *false*, as the following proof that ½ + ¼ + . . . = 1, indicates:

Proof:



What's true is that no diagrammatic proof-procedure can be designed that adequately handles all the tails of infinite processes – more accurately, that handles all possible functions – as "function" is understood in the contemporary mathematical setting.

[119]The informal proof, of course, can be successfully extended to smooth curves with a finite number of discontinuities, and other classes of functions. I omit further discussion of this.

can't be done on the basis of vague impressions of "resemblances" between diagrams and mathematical abstracta.

The same point can be made against Giaquinto (2008a, 31) when he writes:

> Missing an untypical case is a common hazard in attempts at visual proving. A well-known example is the proof of Euler's formula V – E + F = 2 for polyhedra by 'removing triangles' of a triangulated planar projection of a polyhedron. One is easily convinced by the thinking, but only because the polyhedra we normally think of are convex, while the exceptions are not convex. But it is also easy to miss to miss a case which is not untypical or extreme when thinking visually. An example is Cauchy's attempted proof (Cauchy 1813) of the claim that if a convex polygon is transformed into another polygon keeping all but one of the sides constant, then if some or all of the internal angles at the vertices increase, the remaining side increases.... The frequency of such mistakes indicates that visual arguments often lack the transparency required for proof; even when a visual argument is in fact sound, its soundness may not be clear, in which case the argument is better thought of as a way of discovering rather than proving the truth of the conclusion.

This, as earlier, is to use the generality problem – which, recall, is a problem *all* proof-procedures face, diagrammatic *and* language based – against a diagrammatic proof procedure which, otherwise, can be syntactically characterized coherently. There is nothing wrong with taking above diagrammatic proof-procedure to show Euler's result for *convex* polygons. What follows from the frequency of such mistakes isn't that "visual arguments often lack the transparency required for proof" – in particular, that it isn't obvious that they're sound. Soundness turns crucially on interpretation – and what this shows is that interpretations must be chosen with care; this is especially the case when an informal practice is being embedded in a formal practice. One thing these failures *do* show is that diagrammatic proofs often have restricted domains of soundness that are narrower than the range of abstracta that mathematicians are concerned to show results about.[120] But there is nothing obvious, about diagrammatic proofs, that shows this will always be a problem, when comparing such proofs to language-based proof procedures.

My point is that Giaquinto's analysis presupposes (something nineteenth-century mathematicians presupposed as well) resemblance interpretations at work: visual proofs represent functions by drawings that resemble these functions. But this is wrong, and as a result the wrong interpretations are imposed on diagrammatic practices – sometimes (as with the case of functions), the drawings of curves on a page are being confused with the functions they're being taken to depict, with the result that visualization-proof methods are saddled with the expectation that they're responsible for depicting functions for which no syntactic diagrammatic system has ever been characterized, even informally. At root, as I've mentioned, are confusions between the diagrams themselves (in this case, drawings on paper) and functions that can't be seen. The use of the word "intuition," as used in "it's intuitive that a function can't be everywhere continuous but not differentiable anywhere" enables a

[120]This is especially the case with the notion of a function, which from the eighteenth century to the nineteenth century, drastically changed in what it was (implicitly and explicitly) taken to refer to.

confusion between pictorial methods of proof and the items being studied. Lesson: Vague intimations of visual resemblance don't show that a cogent diagrammatic practice has been established; worse, such vague intimations don't sensibly provide interpretations when a diagrammatic practice is cogent.

I've complained about one error that the visual-resemblance views make: confusing a method of *understanding* abstracta with a method of *visualizing* abstracta. This is a failure to realize that a visual process can be used to *understand* something without it supplying a *visualization* of that something.[121] But there is a second way that focusing on visual resemblances – as the machinery that governs the implicit interpretations for diagrammatic elements – goes wrong. This is that there is an implicit *suggestion falsi* that a diagram works its magic by capturing structural resemblances between pictorial elements and the abstracta they depict.[122]

This second important mistake goes far beyond diagrammatic proofs; it infects views of (scientific) modeling generally. Although there is a narrow class of models that can be described as ones that capture aspects of what they model purely by virtue of the models embodying structural resemblances to the items modeled, this is generally not the case.

Consider a computer simulation of a certain dynamical process – the flow of water through a pipe, for example. It would be a mistake to think that we can rely on the simulation to predict certain aspects of that process – when we can – because what happens "in" the computer "resembles" the flow of water through a pipe. That's, frankly, pretty absurd – although this absurd suggestion is made natural by the misleading label: simulation. What happens with a successful computer simulation is quite different: the computer simulation has been designed with certain aspects of water flow in mind – more specifically, it has been designed by taking into consideration certain events that occur in (or during) the flow of water through a pipe, and by ignoring many other ones. Water flow, in particular, is an extremely complex (and badly understood) process where a great deal is going on. It would be a naïve analysis to divide – as it were – water flow into levels: the quantum level, a higher level (where the water flow is approximated by continuum-fluid idealizations), etc. What's actually going on is that certain effects in and to the water are being taken account of and other ones are being ignored, where these effects cross

[121]See Azzouni (2020) for further discussion of this distinction between imagining and conceiving.

[122]See Giaquinto (2008b). It's perhaps no surprise that Giaquinto takes the cognition of abstracta-structure seriously as an explanation of the value of diagrammatic proof: it's a natural move for someone who, to begin with, thinks that resemblance to abstracta is how diagrammatic elements refer to those abstracta. See Resnik (1997), for an earlier book-length argument for this kind of epistemic position vis-à-vis abstracta. Notice that a nominalist must reject this picture of the understanding of proof in any case: I'm not presupposing my nominalism, however, in raising any of the objections of this paper.

“levels” in intricate and subtle ways.[123] The hope, sometimes dashed and sometimes fulfilled, is that the aspects of a phenomenon (water flow) that are being taken account of – by approximations, usually – are ones that will enable the model to capture certain causal effects of that phenomenon (e.g., in the case of water flow the average pressure on the inner pipe surface) while ignoring other ones (e.g., in the case of water flow, changes in temperature of the inner pipe surface). As I said, it would be naïve to impose a structure-picture on such models and what they model – although this is, imaginatively, extremely natural to do. Unfortunately for what’s psychologically natural here, the actual interaction (causal and otherwise) between what’s being ignored by a model and what’s being taken account of is too complicated for that. The process can be *imagined* – misleadingly, to repeat – as one in which the water, in the case of water flow is, at a certain “level” being structurally modeled by the computer simulation. But this, it must be stressed again, isn’t accurate to what’s being retained and what’s being ignored by the model – for example, what’s being ignored aren’t simply the quantum-mechanical, discrete, elements of the water-flow phenomenon.

Instead, and this alternative characterization of the computer simulation directly applies to diagrammatic proof as well, there is a body of *statements* – “predictions,” in the case of water flowing through a pipe. And the model is a piece of computer machinery (that yields statements of the same form as the predictions) that we hope will replicate those predictions that are *true*.[124]

Summary: presumed visual resemblances between diagrammatic elements and the abstracta they supposedly resemble fool mathematicians and philosophers into thinking, in the case of drawing curves, that these resemblances are the proof-theoretic and interpretational machinery of diagrammatic proof – that is, that visual resemblances supply both intuitively effective recognition procedures and visually determined (by resemblance) interpretations for diagrammatic elements. These are mistakes.

14 Inference Packages and Shin on Perceptual Inference

Sun-Joo Shin (1994, Chap. 6) offers a more sophisticated theory of diagrammatic reasoning than resemblance interpretations do. Central to her view are the roles of *perceptual inference* and *convention* which she uses to explain in what ways, and to what extent, any specific formal system is diagrammatic and in what ways and to

[123]Consider the fact that gravity is relatively weak if you’re small enough. Your biological mechanisms for handling your recognition of “up” and “down” can’t turn on gravity as it can for bigger things. Similarly, in engineering design, certain effects, wind, torque, etc. can be ignored when the items (bridges, shoulders) are certain sizes, and not otherwise. How, and to what degree, causal effects can be ignored or minimized by a model is badly captured by thinking in terms of “levels.”

[124]Exactly how inferential practices capture the supposed properties of mathematical entities will be described in terms of our capacity to “simulate algorithms” – where I’m using “simulate” in the same misleading way it’s used above. See Sect. 14.

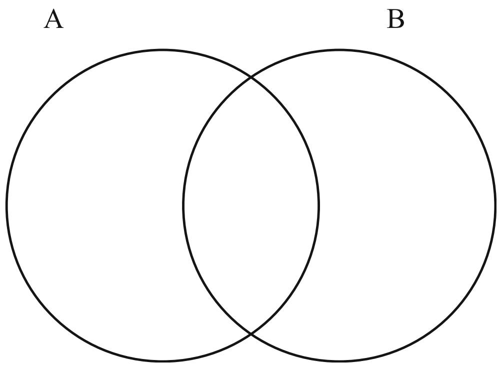
what extent it isn't. The difference between the use of perceptual inference and the use of convention (when looking at something), on her view, is that conventions need to be learnt; on the other hand, when we use perceptual inference, we exploit aspects of a diagram that we don't have to learn how to use. Importantly, she thinks, the respective amounts (as it were) of conventionality and perceptual inference involved in our ability to navigate linguistic texts, diagrams, and pictures comes in degrees. A photograph has no conventional elements, nearly enough: we use and only use perceptual inference to recognize what visual elements in a picture refer to (and, therefore, how a photograph depicts things to be). A pure language-based piece of text, using symbols, is entirely conventional. Diagrams are in-between – usually involving both conventional elements and perceptual inferences.

The examples Shin explicitly gives, and others that we can easily apply her distinction to, make that distinction tolerably clear. A depiction of the seating arrangements of Tom, Susan, and Mary like so:

Tom Susan Mary,

involves convention insofar as "Tom" refers to Tom, "Susan" to Susan, and "Mary" to Mary; it involves perceptual inference insofar as the side-by-side seating arrangement of Tom, Susan, and Mary is recognized to be as depicted above.

Similarly, in Venn-I, it's a convention that the closed curves and rectangles represent sets, and – specifically – that their interiors represent the interiors of those sets.[125] It's equally conventional, in a language-based setting, to introduce capital letters, A, B, . . . for names of sets, lower-case letters, a, b, . . ., for names of elements in sets, and to introduce a symbol, $\in$, for membership. Shin takes the distinctive roles for perceptual inference and for convention in Venn-I, as opposed to a language-based class system to contrastively emerge when we consider relations between sets. In the language-based case, we need to introduce a *new* syntactic device to conventionally represent intersection: $\cap$, so that "$A \cap B$" thus can represent the intersection of A and B. In the case of Venn diagrams, we don't need to do this. Consider, for example, the following figure:



[125]Shin (1994, 169) writes: "In Venn-I we adopt a convention that assigns a set to a basic region."

Shin claims that, given the previous conventions, the interpretation of the common spatial area of the circle associated with A and the one associated with B *must be* the intersection of the sets A and B. This isn't, therefore, the introduction of a new syntactic convention; instead, we use perceptual inference, along with the earlier-mentioned conventions, to see what spatial area in the diagram represents the intersection of the sets.

Shin (1994, 161–162) similarly argues that if we already have in place a convention that (two-dimensional) boxes represent sets and dots represent individuals, then to represent the set-theoretic membership relation, one need only place a dot in a box. She adds, "Putting down an object *in* another object does not involve any new syntactic device. A spatial arrangement between a dot and a box represents a nonspatial relation, 'being a member of'" (italics hers).[126]

There's an important subtlety that Shin's remarks – as quoted – overlook. If we interpret dots and rectangles (and closed curves) – the lines of the figures, that is – as Shin suggests, then we need an additional convention that dictates that the *interiors* of squares and circles represent where the members of the corresponding sets are located. Even if we represent *area-filling* rectangles (as opposed to the borders alone) as sets, it's still open to us to conventionally dictate an item as a member of a set only if it appears *outside* the interior of the figure rather than inside.[127] Putting a figural element inside another figural element *is* a piece of (two-dimensional) syntax that needs to be interpreted, and the interpretation of which goes beyond the sheer convention that a two-dimensional figure (area-filling or otherwise) represents a set. We can see the point I'm making if we consider the typical language-based notation, Pa, in which a piece of syntax (concatenation) is interpreted as predication. The situation is exactly the same one.

Nevertheless, I think Shin's approach to diagrammatic reasoning is largely correct, insofar as she recognizes the role of what she calls perceptual inference. I now introduce some alternative terminology and make a few other changes to her approach. Perceptual inference involves a number of *cognitively embodied algorithms,* what I call *inference packages*.[128] Inference packages are modularized and

[126]She continues:

> An isomorphism between this spatial arrangement and the relation "being a member of" is not as perceptually obvious as an isomorphism between a spatial arrangement in a diagram [as in the Tom, Sally, Mary example above] and the relation "being to the right of." However, this isomorphism is more perceptually obvious than any linguistic symbol that a linguistic representation adopts, since no extra convention involving syntactic devices is required.

In adapting elements of her position, I'll remove talk of isomorphisms.

[127]For manageability, the shadings and dots, etc. to be placed outside squares could be conventionally quite close to the outsides of the figures.

[128]I borrow, in part, this discussion of "inference packages," and other aspects of visualization, from Azzouni (2005).

content-specific.[129] A nice illustration of one, however, is our capacity to imagine triangles on a plane or a sphere. We recognize that if we move the triangles rigidly about on a plane that doesn't change their areas or change the magnitudes of their interior angles. If we imagine an equiangular triangle growing in area in a way that preserves its equiangularity, the sum of its angles won't change. This isn't true, we immediately recognize, if we instead imagine a triangle on a sphere. In this case, moving the triangle about rigidly won't change the magnitude of its interior angles, or the areas of those triangles. But if we grow an equiangular triangle in a way that (again) respects its equiangularity, the magnitudes of those angles change: It gets "more curved" and its angles gain magnitude. One can also imagine blowing up the sphere – like a balloon – uniformly, although keeping the triangle the same in its area; in this case its angles flatten and lose magnitude. Triangles moving around on an ellipsoid act differently still: if we imagine moving a triangle from the flattest part of the ellipsoid to one of its ends, we can see how the magnitudes of the interior angles of the triangle will increase and decrease.

Involved here are various "perceptual inferences": we can see some facts about the angles, lines and areas of the triangles immediately (they're "free rides"); others we see quickly, or we can work them out with a step or two. I call these inference *packages* because, from an axiomatic point of view, a number of different relations between the properties of these figures (that can be distinguished axiomatically) are bundled together. I stress again that these packages are content-specific: the inferences govern exactly what we imaginatively experience them as governing: certain sorts of visual phenomena. I hypothesize that there is more than one inference package that we can apply to two-dimensional visualizations; this is a matter to be determined empirically. Included among them, however, are the part/whole inferences about spatial regions that we see at a glance: for example, when a spatial area is part of another area, and when they're disjoint. It's important to realize that we have inference packages with respect to one-dimensional figures as well as with respect to three-dimensional figures, and that these aren't the same. (Some of them, at best, overlap.) We effortlessly recognize a number of relations alphabetic items have when in concatenated strings. Certain patterns "leap out" at us; others don't. The same is true of the inference packages with respect to three-dimensional objects located in space. These differ from two-dimensional ones simply because we can imagine two-dimensional objects as *on* surfaces, and as inheriting their properties from those surfaces. Three-dimensional objects are "in" space, and not experienced as "on" anything. This affects what kinds of inference packages are available with such space-filling figures.

---

[129]Exactly how inference packages are modularized is a subtle empirical matter that I have to avoid saying much more about. There is more than one visual mechanism at work in visual perception – the evidence for this is that we naturally recognize some visual patterns and not others, and that these recognitional abilities differ greatly between individuals. Exactly what neurophysiological resources are at work, and how they operate in tandem is undergoing (currently) intense study. I *don't* assume one-to-one relationships between inference packages and visual-perception capacities.

Diagrammatic conventions can be chosen in ways that run in parallel with our inference packages or in ways that work against them – even though the diagrammatic conventions in both cases can be equally coherent. Imagine, for example, a systematic diagrammatic Venn system, involving rectangles, closed curves, diagrammatic elements, ⨂, —, and shadings, such as Shin's, but where the representation conventions for regions are inverted: for example, to indicate that a region R has no members, we must shade all the regions *outside* R.[130] This, coupled with the appropriate conventions for the other elements, yields a perfectly consistent system; it's – logically speaking – impeccable. It will, however, be a very *user-unfriendly* system because it works against our inference packages which operate naturally via spatial wholes and *their* spatial parts. This isn't to say we couldn't get used to such an alternative diagrammatic practice – and that some of us would even find ourselves as able to work with it with some comfort; it's to say there would be a learning curve.[131]

In describing this imagined alternative to Shin's Venn-I as user-unfriendly, I'm making a phenomenological point: to use such a diagrammatic system for proofs feels differently from so-using Venn-I (at least with respect to those diagrams in Venn-I that phenomenologically resemble the diagrams in the original Venn practice). When we engage in a proof-procedure that, relatively directly, uses our cognitively embodied inference packages *we feel we understand what's going on*: we *get* it. That's true with all cognitively embodied inference packages: the inferentially packaged reasoning we engage in is one we always feel we understand: we feel we recognize why it must be right.

A simple phenomenological rule for proof procedures – formal and otherwise: The more closely we can use our cognitive-embodied inference patterns to simulate the processes involved in carrying out a proof, the more we experience that proof as one we understand or "get." To illustrate, consider sets. Sets, recall, are supposed to be abstract objects that have members. We can imagine a set as a kind of box in which its members are contained – but that's a spatial metaphor that only goes so far. Sets aren't in space and, in particular, they aren't located where their members are. The set containing the six umbrellas, the one that I have in my possession now, and the five that I lost last week, isn't located at (or where) the umbrellas it contains are. There is a sense in which "sets," therefore, make no sense. What sorts of things can these be? But the reasoning about sets can be imitated – up to a point – by reasoning instead about objects in boxes. More pertinently to Venn diagrams, reasoning with sets can be imitated – up to a point – by reasoning about regions and their interiors. And so, reasoning about sets can be executed by a notational device that exploits our ease with reasoning about regions and their interiors.

[130] Again, with a convention, as described in footnote 127, to make markings outside squares manageable.

[131] Consider Polish notation. Ordinary alphabetic formal systems, using parentheses and the like, operate in parallel with our one-dimensional containment inference packages. The result is that we can read that notation easily, and learn it quickly. Polish notation flouts these inference packages – and so is "intuitively unappealing."

To stress again: inference packages (cognitively embodied algorithms) always come with content; they are always interpreted – and indeed, the interpretations are always *sensory*.[132] But when mathematicians use these inference packages they invariably *reinterpret* them. I've just illustrated one way that this is done: by using them to manipulate diagrammatic notation that our cognitively embodied algorithms are at ease with, although those diagrammatic elements (officially) refer to something else.

Let me return briefly to the question, raised in Sect. 8, about the distinction between when notation is diagrammatic and when it's not. The foregoing implies, I think, the following result. There are distinctive visual capacities that enable distinctive inference packages with respect to one-, two-, and three-dimensional visual experience. When a two-dimensional notation doesn't take advantage of the distinctive inference packages available with two-dimensional notation (but not one-dimensional notation), but only relies on, say, adjacency, that notation isn't diagrammatic. Thus, I suggest, matrices and integral notation aren't diagrammatic notation. When notations rely on distinctively two-dimensional inference packages, as Euclidean and Venn notation do, they are diagrammatic. This distinction requires an empirical underpinning, which I can hope will be supplied in the future; it doesn't turn – notice – on semantic considerations, that is, it doesn't turn on interpretation. (I think this is an advantage of it.)

15 Ideology and Formal Systems

Before (finally) concluding this chapter, I'd like to revisit the interrelationships between Church's thesis, Hilbert's thesis, formal systems, and informal rigorous mathematics, while at the same time discussing some recent work by Avigad and Hamami that defends a particular version of the derivational account.

Twentieth- and twenty-first-century mathematics is extremely rich – and in large part a lot of that richness is a result of mathematical developments subsequent (and in contrast) to the logic that emerged with Frege and was joined with set theory by Russell and Whitehead, and others. In particular, major parts of contemporary mathematics are the alternative axiom systems that have emerged *in contrast to* Zermelo-Frankel set theory with the axiom of choice (ZFC) that's couched in the language of classical first-order predicate logic. Intuitionism is well-known: it has generated a great deal of *informal rigorous* mathematics – in (intuitionistic) analysis, in particular. But, of course, there are also numerous "alternative logics" and some of these alternatives – paraconsistent approaches, among others – are actively pursued not just as logical disciplines but in terms of the mathematics they also enable. There are also explicit alternatives to ZFC, including ZF + the axiom of indeterminacy. Again, these have led to additional pursuits in bread-and-butter mathematics, for example, alternatives to standard measure theory.

[132] Although: they're not always *visual*.

There is no choice, I would have thought, but to be a *pluralist* as far as contemporary pure mathematics is concerned. It's simply a fact of the contemporary mathematics that these numerous alternatives – alternative to classical logical frameworks, alternatives to standard Zermelo-Frankel set theory with the axiom of choice – exist. Call the formalized pursuit of these alternatives *axiomatic pluralism*.[133]

Much – if not most – of this mathematics *isn't* restricted to the deduction of theorems within formal systems, as I've already indicated; rather, it's bread-and-butter informal rigorous mathematics. That's the context, actually, in which such axiomatic systems are studied. More obviously, it's in the context of informal rigorous mathematics that the knock-off studies I described above, in one or another nonstandard framework, a particular logic and/or set-theoretic framework, or something else (e.g., category theory) occur. Understanding informal rigorous mathematical proof as a matter of mechanical recognizability – either according to the algorithmic-device view or some other – enables an accommodation of this pluralism, which I'll call *informal mathematical pluralism*. This also accommodates the datum that mathematicians can recognize good proofs across mathematical traditions.

Informal mathematical pluralism is why, however, that the derivational accounts I've been concerned with in this chapter, and in earlier work, have been *ideology-neutral* accounts. The formalizations in question that are supposed to explain the epistemic properties of informal rigorous mathematics can't be restricted to frameworks governed by standard classical logic and ZFC. Indeed, foundational projects for mathematics – ones dedicated to transcribing all mathematics into one or another *specified* formal system (e.g., ZFC, Russell-Whitehead type theory), haven't been in focus at all.[134] A formalization of arithmetic with an ideology of *numbers* (an intended model of numbers) suffices to provide the formalization needed for an informal rigorous *arithmetical* practice – on the derivational views I've been concerned with. Church's thesis (going beyond functions of natural numbers via reinterpretations of numbers) accommodates axiomatic pluralism. The version of Hilbert's thesis I borrowed from Kripke (2013) in Sect. 4, however, requires changes to enable its application beyond the language of first-order predicate calculus with identity. Notice: Although any Turing machine formalism can be *represented* in the language of first-order predicate calculus with identity – that's a mathematical result about Turing machines – this result doesn't straightaway yield a first-order-predicate-calculus-with-identity version of Hilbert's thesis.

These preliminaries aside, recent work by Hamami (2019) defends what he describes as "orthodoxy," or "the standard view," and he traces it (historically) back to Mac Lane and Bourbaki. This is that "a mathematical proof P is *rigorous* if and only if *P* can be *routinely translated* into a formal proof," where such a proof

133 I committed myself to axiomatic pluralism (without using that label) in Azzouni (1994, Part II).

134 One may well doubt, in light of the richness of contemporary mathematics, whether any straightforward ideologically nonneutral foundational project is possible. Such can't be ruled out, of course, because there can always be rather unexpected approaches – based on relative consistency proofs – to embedding branches of mathematics in one another, and therefore grounding such in specified axiomatic systems.

"is (or could be) written out in the first order predicate language L(∈) as a sequence of inferences from the axioms ZFC, each inference made according to one of the stated rules."[135] As I've just indicated, I think this version of a derivational account cannot accommodate contemporary mathematics because it isn't ideologically neutral. Let us, accordingly, broaden it. Further, let us distinguish (as Hamami (2019) and Avigad (2019) both do) an epistemic/descriptive version of the derivational account from a normative version of that account. The epistemic/descriptive account is required to use the existence of formalizations to explain the epistemic properties of informal rigorous mathematical proof. The normative version instead defines a mathematical proof as "rigorous" if it can be routinely translated to a formal proof. No explanation of the epistemic qualities of informal mathematical proof is required.[136]

Consider the descriptive version of the derivational approach, and recall Tanswell's condition *Content* (Sect. 3). The task *Content* poses to descriptive derivational approaches, pretty much, is to require them to show how the materials, *already present in informal proofs*, indicate their formal transcriptions. I attempted, in Azzouni (2004) and Azzouni (2006), to defend this approach by giving an open-ended set of informal-mathematical tools – inferential abbreviations and shortcuts, meta-proof considerations, etc. – many of which mathematicians have learned, that I hoped would provide sufficient content to informal proofs to show how they indicate the relevant derivations.[137] It strikes me that, insofar as Hamami (2019) considers the descriptive strategy, this is what he offers as available for this – although he stresses more strongly than I did the idea that mathematicians come to the task with certain background competences.

Here is Avigad's (2019) excellent (and approving) description of Hamami's account of informal proof:

> According to Hamami, we accumulate not just theorems but inferential patterns and procedures throughout our mathematical education, and we put them to good use towards understanding an informal proof. When a competent practitioner reads such a proof, the inferential cues and context trigger algorithmic processes that expand the inferential steps to smaller ones. These, in turn, are recognized as applications of prior theorems or lemmas, or at least patterns in memory that have been previous justified.

[135]The first quote is from Hamami (2019); the second is his approving quotation of Mac Lane (1986, 377). "Rigorous," here is specialized terminology; it doesn't mean what it means in the phrase, commonly used (and used by me throughout this chapter), "informal rigorous mathematical proof."

[136]Hamami (2019) writes: "[A] *descriptive account* of mathematical rigor provides a characterization of the mechanisms by which mathematical proofs are judged to be rigorous in mathematical practice; *a normative account* of mathematical rigor stipulates one or more conditions that a mathematical proof ought to satisfy in order to qualify as rigorous" (italics his).

[137]Strikingly, the meta-proof considerations include those where shifts occur during the course of a proof to explicitly notational considerations – for example, that the indexes on such operators have such-and-such properties. These notational considerations are ones, of course, about notation occurring in the informal rigorous proof.

The algorithmic-device view agrees that "inferential cues and context trigger algorithmic processes," but these needn't (and often don't) lead to expansions of inferential steps; they often lead to just seeing the result. More important, they usually *don't* point *past* informal proving to a formal transcription.[138] That's because algorithmic competence is at back of all of this, but algorithmic competence, as I've been stressing throughout this paper, *doesn't* epistemically *require* formalization. Church's thesis, coupled with the mathematical result that any Turing-machine computation, can be represented formally in a first-order predicate language *does not* translate to an epistemic requirement of a formalization of this sort explaining our grasp of any intuitively effective bit of reasoning.

Avigad (2019), on the other hand, although officially accepting Hamami's approach, attempts to give the *normative* version of the derivational account some teeth. He tries to show, that is, that a number of the virtues that good informal mathematical proofs (and proof practices) have – for example, being based on sound analogies, reasonable generalizations, and a wide collection of examples – supply *good guarantees* that an informal proof will have a formal correspondent. Avigad means to show that the existence of corresponding formal derivations is thus an appropriate standard for informal proof practices: the gap between formal derivations and informal proofs isn't a reason to reject derivations-as-standards precisely because good informal proof-practices *raise the likelihood* that an informal proof has a formal correspondent.

I worry (as I have for a long time), however, that "good guarantees" – raising likelihoods in these ways – aren't good enough to explain the epistemic strangeness of informal rigorous mathematical proof. That is, informal mathematical proof practices are *too good* to be explained this way. Recall the quotation from Manders (1995) in footnote 19; for that matter, here's a quotation from MacKenzie (2001, 322–323)[139]:

> . . .many rigorous arguments in ordinary mathematics have been replaced successfully by formal proofs, using automated theorem provers and proof checkers, especially the AUTOMATH and MIZAR systems . . .. What is most remarkable about these many replacements of rigorous arguments of mathematics with formal, mechanized proofs . . . [is that] it is a conservative process. Applied to programs, hardware designs, and system designs, efforts at formal, mechanical proof frequently find faults and deficiencies that have not been detected by other means . . .. Applied to rigorous arguments within mathematics, however, efforts at mechanized proof nearly always suggest at most the need to remedy matters that a mathematician would regard as basically trivial, such as typographic errors or failures to state the full range of conditions necessary for a theory to hold.

[138] They certainly *don't* point past informal rigorous mathematical proofs with one kind of content – arithmetic, say – by virtue of being "decomposable" (Hamami's phrase) into proofs with a strikingly different ideology, that is, that of ZFC.

[139] I'm under the impression that, vis-à-vis various formalization projects, and their impact on informal rigorous mathematics, nothing has changed in the intervening years.

But Avigad's general-virtue approach to mathematical proof *should yield* the same result for programs, hardware designs and system designs that Avigad hypothesizes it yields with respect to informal rigorous mathematical proof. It clearly doesn't.[140]

I've suggested in earlier work (Azzouni 2009b) – in a way related to Avigad's (2019) approach to a normative role for formal derivations – that transcribability to a formal derivation has, in the contemporary setting, become a norm for informal rigorous proof. I want to end this section by revisiting considerations that cut against that idea.[141] The problem is that there are informal rigorous mathematical proofs that are *counterexamples* to the normativity thesis. Consider, for example, the proof in footnote 118. (Visual proofs like that one are the best counterexamples, but there are many similar ordinary-language-based ones.) Phenomenologically – notice – this proof is *utterly convincing* as it stands. Furthermore, there is no sense in which it looks like it needs to be completed or filled in. It's true of many informal rigorous mathematical proofs that they look like they're missing steps; but this is *not* one of those.[142] That is, neither epistemically nor normatively does this proof and many other informal rigorous proofs (those in the Euclidean tradition, e.g., Venn diagrammatic proofs – but *many* other informal rigorous mathematical proofs as well) need supplementation of any sort. Further, although Hilbert's thesis indicates that these proofs can all be formalized (and indeed, as I've illustrated by citation from the literature, many of them *have been*), they themselves don't indicate the existence of formalizations that, in turn, justify why they're true: their content, that is, does nothing of this sort.[143]

The epistemic process, rather, is the exact *reverse* of what normative and descriptive derivational accounts hypothesize. The intuitively effective procedures such proofs exhibit right on their surfaces, when preserved formally, simultaneously preserve the epistemic qualities (the phenomenology) of those informal proofs. The formalization inherits, that is, what it is about the informal proof that convinces us – what *justifies* our being convinced of the result of the proof. It's not, that is, that the formalization reveals what's convincing about that proof or that the formalization justifies that proof. This is enough to show that – at least with respect to *many* informal rigorous mathematical proofs – derivation accounts are intrinsically misleading.

Nevertheless, compatibly with how I've just described the phenomenology of some proofs that formal derivations correspond to informal proofs might *still* be a (general and newly acquired) norm. Whether such a correspondence *is* a norm is,

---

140 Compare this response to Avigad with my earlier (Azzouni 2009b, 16, footnote 12) response to Rav (2007, 317).

141 I first raised this objection to my earlier norm view in Azzouni (2013c) – but this discussion goes beyond that one.

142 That to formalize this proof would involve something like translating visually seen spatial areas into complex axioms about areas isn't to "fill in" the original proof. It's not even to "make explicit" what's tacit in the proof, as it stands. That's an interpretation that treats, by fiat, a language-based transcription of something we see visually, as therefore (on those grounds alone), "more explicit."

143 The fact that other proofs of the same form may be misleading is irrelevant.

ultimately, a sociological matter. Consider, however, the possibility of a flourishing diagrammatic tradition that – contrary to Hilbert's thesis – can't be transcribed to a language-based formalization. More dramatically, consider a flourishing informal diagrammatic tradition (involving proofs, say, as phenomenologically compelling as the one in footnote 118) that can't be formalized *at all* – diagrammatically or otherwise. Rather than there being a norm that would fault this proof-tradition, it seems, rather, that generalized versions of Hilbert's thesis, and, in fact, Church's thesis, would be *overthrown.* Indeed, that these are regarded as *theses* is, in fact, the sociological evidence that formalizability of any sort *isn't* a norm of informal rigorous mathematical proof.

16 Conclusion

I've covered far too much territory for an *article.* I'll nevertheless try to summarize (some of) what I've done here in a couple of concluding paragraphs. The problem this chapter opened with is the epistemic strangeness of mathematical proof. This is a (psychological/sociological) claim about informal rigorous mathematical proof – that there is a widely shared experience of "it must be this way" that proofs in mathematics sometimes have (and that long proofs exhibit, usually, only in part) as well as "aha – I see why this must be" epiphanies. That is, mathematical proof, in full, is a complex heterogeneous experience, and that experience (involving short-cuts, invocations of authority, "aha"-epiphanies, and so on) is hardly a seamless impression of a priori inference. The algorithmic-device view explains this phenomenology: algorithmic devices – diagrammatic and language-based – exploit our ability to execute some intuitively effective recognition procedures – ones that are cognitively embodied in us and that we can apply to mathematical problems.

I'll put the point more picturesquely: A mathematical proof is like a stew with chunks of reasoning governed by algorithmic devices plus lots of other – and some looser – stuff. Usually, but not always, algorithmic devices are derived from recognition procedures we normally use (to navigate our world visually, for example) and then we feel the "aha"-epiphany when we apply them. But we don't otherwise – for example, if we use an abacus for certain calculations, or engage in a difficult computational maneuver with integrals.

We cognitively embody many kinds of algorithms. When applying these to the topics they naturally apply to, we effortlessly experience inference processes, often as immediate inferential insights about things. Successful diagrammatic practices notationally hijack these processes and thus enable us to apply these inference packages to subject areas that such inference packages wouldn't otherwise apply to.

The algorithmic-device view of mathematical proof, unlike derivational approaches, explains the epistemic properties of informal rigorous mathematical proof in terms of the mathematical machinery that's *visible* in informal rigorous mathematical proof *itself*. It doesn't explain these properties via a transmutation of informal rigorous mathematical proof to formal algorithmic systems – diagrammatic or otherwise. In turn, other properties of informal rigorous mathematical proof (e.g., Tanswell's "agreement," "correctness") are explained by the fact that the phenomenology of informal rigorous mathematical proof is based on inference packages and various conventions that are held in common and applied in common by appropriately trained mathematicians.

Nevertheless, transmutations of informal rigorous mathematics into one or another formal system are always possible, via generalizations of Hilbert's thesis. In particular, any natural-language reasoning with diagrammatic algorithmic devices can be formalized in a heterogeneous formal system. Any such formal heterogeneous system, in turn, can always be recast as a language-based formal system because all diagrammatic operations on higher-dimensional grid systems can be recast in one-dimensional concatenation systems – and indeed, as Turing machine calculations. Phenomenologically faithful embeddings of informal rigorous mathematical proof into formal systems retain the apparent content of the mathematical statements – at least insofar as there remains an intended model for the formal systems in question. Respecting this apparent content also explains why the formal systems in question are always based on something in the neighborhood of classical bivalent logic. That classical mathematics is couched in a logic in the neighborhood of the classical first-order predicate logic despite its taking place in natural language which is *not* governed by such a logic is the result of a complex historical process.[144] That means the process is contingent: we could (in the future) find ourselves – collectively – deserting classical logic, as we, in effect, have already with respect to pure mathematics. So too, we might desert ZFC as we, in effect, also have with respect to pure mathematics. This latter possibility turns, largely, on the empirical-science needs that mathematics meets. But discussion of that involves a different set of considerations I can't pursue now; my only point – again – is to stress that our philosophical understanding of ordinary rigorous mathematical proof, certainly as it is and has been pursued in this and the last century, requires axiomatic and informal mathematical pluralism.

References

Abramson FG (1971) Effective computation over the real numbers. Twelfth annual symposium on switching and automata theory. Institute of Electrical and Electronics Engineers, Northridge

Avigad J (2019) Reliability of mathematical inference. Synthese. https://doi.org/10.1007/s11229-019-02524-y

Avigad J, Dean E, Mumma J (2009) A formal system for Euclid's elements. Rev Symb Log 2(4):700–768

Azzouni J (1994) Metaphysical myths, mathematical practice: the ontology and epistemology of the exact sciences. Cambridge University Press, Cambridge

Azzouni J (2000) Applying mathematics: an attempt to design a philosophical problem. Monist 83(2):210–227

[144] I tell this story in Azzouni (2005) and Azzouni (2009b).

Azzouni J (2004) Proof and ontology in Euclidean mathematics. In: Kjeldsen TH, Pedersen SA, Sonne-Hansen LM (eds) New trends in the history and philosophy of mathematics. University Press of Southern Denmark, Odense, pp 117–133
Azzouni J (2005) Is there still a sense in which mathematics can have foundations? In: Sica G (ed) Essays on the foundations of mathematics and logic. Polimetrica International Scientific Publisher, Monza, pp 9–47
Azzouni J (2006) Tracking reason: proof, consequence, and truth. Oxford University Press, Oxford
Azzouni J (2009a) Evading truth commitments: the problem reanalyzed. Log Anal 52(206):139–176
Azzouni J (2009b) Why do informal proofs conform to formal norms? Found Sci 14:9–26
Azzouni J (2013a) That we *see* that some diagrammatic proofs are perfectly rigorous. Philos Math (III) 21:323–338
Azzouni J (2013b) Semantic perception: how the illusion of a public language arises and persists. Oxford University Press, Oxford
Azzouni J (2013c) The relationship of derivations in artificial languages to ordinary rigorous mathematical proof. Philos Math (III) 21:247–254
Azzouni J (2017a) Does reasoning evolve? (does the reasoning in mathematics evolve?). In: Bharath Sriraman (ed) Humanizing mathematics and its philosophy: essays celebrating the 90th birthday of Reuben Hersh. Springer, Cham, pp 253–289
Azzouni J (2017b) The rule-following paradox and its implications for metaphysics. Springer, Cham
Azzouni J (2018) Deflationist truth. In: Glanzberg M (ed) The Oxford handbook of truth. Oxford University Press, Oxford, pp 477–502
Azzouni J (2020) Conceiving and imagining: examples and lessons. In: Moser K, Sukla AC (eds) Imagination and art: explorations in contemporary theory. Brill, Leiden, pp 281–303
Barwise J (1993) Heterogeneous reasoning. In: Mineau G, Moulin B, Sowa J (eds) Conceptual graphs for knowledge representation. ICCS 1993. Lecture notes in computer science (Lecture notes in artificial intelligence), vol 699. Springer, Berlin, pp 64–74
Barwise J, Shimojima A (1995) Surrogate reasoning. Cogn Stud 4(2):7–27
Black M (1946) Critical thinking. Prentice Hall, Upper Saddle River
Brown JR (1999) Philosophy of mathematics: an introduction to the world of proofs and pictures. Routledge, London
Burgess JP, Rosen G (1997) A subject with no object. Oxford University Press, Oxford
Cappellan H (2018) Fixing language: an essay on conceptual engineering. Oxford University Press, Oxford
Cauchy A (1813) Sur les polygones et les polyèdres. Gauthier-Villars, Paris
Church A (1936) An unsolvable problem of elementary number theory. Am J Math 58:345–363
Church A (1954) Introduction to mathematical logic, vol I. Princeton University Press, Princeton
Copeland BJ (2019) The Church-Turing thesis. The Stanford encyclopedia of philosophy (ed: Zalta EN), Spring 2019 edn. https://plato.stanford.edu/archives/spr2019/entries/church-turing/
Davidson D (1986) Inquiries into truth & interpretation. Oxford University Press, Oxford
De Toffoli S, Giardino V (2015) An inquiry into the practice of proving in low-dimensional topology. In: Lolli G, Panza M, Venturi G (eds) From logic to practice: Italian studies in the philosophy of mathematics. BSPS, vol 308. Springer International Publishing, Cham
Descartes R (1931) Rules for the direction of the mind. In: The philosophical works of Descartes (trans: Haldane ES, Ross GRT). Cambridge University Press, London, pp 1–77
Detlefsen M (2008) Purity as an ideal of proof. In: Mancosu P (ed) The philosophy of mathematics. Oxford University Press, Oxford, pp 179–197
Feferman S (2012) And so on . . .: reasoning with infinite diagrams. Synthese 186:371–386
Frege G (1879) Begriffsschrift, a formula language, modeled upon that of arithmetic, for pure thought. In: van Heijenoort J (ed) From Frege to Gödel: a source book in mathematical logic, 1879–1931. Harvard University Press, Harvard, pp 5–82 (1967)

Gandy R (1980) Church's thesis and principles for mechanisms. In: Barwise J, Keisler HJ, Kunen K (eds) The Kleene symposium. North-Holland, Amsterdam, pp 123–148
Gardner M (1988) Hexaflexagons and other mathematical diversions. University of Chicago Press, Chicago
Giaquinto M (2007) Visual thinking in mathematics: an epistemological study. Oxford University Press, Oxford
Giaquinto M (2008a) Visualizing in mathematics. In: Mancosu P (ed) The philosophy of mathematical practice. Oxford University Press, Oxford, pp 22–42
Giaquinto M (2008b) Cognition of structure. In: Mancusu P (ed) The philosophy of mathematical practice. Oxford University Press, Oxford, pp 43–64
Gödel K (1931) On formally undecidable propositions of Principia Mathematica and related systems I. In: van Heijenoort J (ed) From Frege to Gödel: a source book in mathematical logic, 1879–1931. Harvard University Press, Harvard, pp 596–616 (1967)
Hafner J, Mancosu P (2008) Beyond unification. In: The philosophy of mathematical practice (ed) Paolo Mancosu. Oxford University Press, Oxford, pp 151–176
Hallett M (2008) Refletions on the purity of method in Hilbert's Gundlagen der geometrie. In: Mancosu P (ed) The philosophy of mathematical practice. Oxford University Press, Oxford, pp 198–255
Hamami Y (2019) Mathematical rigor and proof. Rev Symb Log. https://doi.org/10.1017/S1755020319000443
Hammer E (1994) Reasoning with sentences and diagrams. Notre Dame J Form Log 35(1):73–87
Hammer E, Danner N (1996) Towards a model theory of diagrams. J Philos Log 25(5):463–482
Harris JH (1982) What's so logical about the 'logical axioms'? Stud Logica 41:159–171
Heath TL (1956) The thirteen books of Euclid's elements, vol I, 2nd edn. Dover, New York
Hersh R (1997) What is mathematics, really? Oxford University Press, Oxford
Howse J, Stapleton G, Taylor J (2005) Spider diagrams. LMS J Comput Math 8:145–194
Kaye R (1991) Models of Peano arithmetic. Oxford University Press, Oxford
Kleene SC (1936) Lambda-definability and recursiveness. Duke Math J 2:340–353
Kripke S (2013) The Church-Turing thesis as a special corollary of Gödel's completeness theorem. In: Copeland BJ, Posy CJ, Shagrir O (eds) Computability: Turing, Gödel, Church and beyond. MIT Press, Cambridge, MA, pp 77–104
Larvor B (2012) How to think about informal proofs. Synthese 187(2):715–730
Lewis D (1991) Parts of classes. Basil Blackwell, Oxford
Mac Lane S (1986) Mathematics: form and function. Springer, New York
MacKenzie D (2001) Mechanizing proof: computing, risk, and trust. The MIT Press, Cambridge, MA
Manders K (1995) The Euclidean diagram. In: Mancuso P (ed) Philosophy of mathematical practice. Oxford University Press, Oxford, pp 80–133
Manders K (2008) Diagram-based geometry practice. In: Mancuso P (ed) Philosophy of mathematical practice. Oxford University Press, Oxford, pp 65–79
Miller N (2007) Euclid and his twentieth century rivals: diagrams in the logic of Euclidean geometry. CSLI Publications, Stanford
Miller N (2012) On the inconsistency of Mumma's Eu. Notre Dame J Form Log 53(1):27–52
Mumma J (2008a) Ensuring generality in Euclid's diagrammatic arguments. In: Stapleton G, Howse J, Lee J (eds) Diagrammatic representation and inference. Springer, Cham, pp 222–235
Mumma J (2008b) Review of Nathaniel Miller, Euclid and his twentieth century rivals: diagrams in the logic of Euclidean geometry. Philos Math (III) 16:256–281
Mumma J (2010) Proofs, pictures, and Euclid. Synthese 175:255–287
Mumma J (2019) The Eu approach to formalizing Euclid: a response to "On the inconsistency of Mumma's Eu". Notre Dame J Form Log 60(3):457–480
Panza M, Sereni A (2015) On the indispensable premises of the indispensability argument. In: Boston studies in the philosophy and history of science: from logic to practice, vol 308. Springer, Cham, pp 241–276

Plato (1963) Meno (trans: Guthrie WKC). In: Hamilton E, Cairns H (eds) Plato: the collected dialogues. Princeton University Press, Princeton, pp 353–384
Polythress V, Sun H (1972) A method to construct convex connected Venn diagrams for any finite number of sets. Pentagon 31:80–83
Quine WV (1953) On what there is. In: From a logical point of view (1980). Harvard University Press, Cambridge, MA, pp 1–19
Rav Y (1999) Why do we prove theorems? Philos Math 7(3):5–41
Rav Y (2007) A critique of a formalist-mechanist version of the justification of arguments in mathematicians' proof practices. Philos Math 15(3):291–320
Resnik MD (1997) Mathematics as a science of patterns. Oxford University Press, Oxford
Rogers H Jr (1967) Theory of recursive functions and effective computability. McGraw-Hill, New York
Rudin W (1970) Real and complex analysis. McGraw-Hill, New York
Shin S-J (1994) The logical status of diagrams. Cambridge University Press, Cambridge
Stapleton G, Howse J, Thompson S, Taylor J, Chapman P (2013) On the completeness of spider diagrams augmented with constants. In: Moktefi A, Shin S-J (eds) Visual reasoning with diagrams. Springer Basel, Heidelberg, pp 101–133
Tanswell FS (2015) A problem with the dependence of informal proofs on formal proofs. Philos Math 23(3):295–310
Tanswell FS (2018) Conceptual engineering for mathematical concepts. Inquiry 61(8):881–913
Tanswell FS (forthcoming) Go forth and multiply! On action, instructions and imperatives in mathematical proof (ed: Joshua Brown, Otávio Bueno), volume on Azzouni. Springer, Cham
Tarski A (1933) The concept of truth in formalized languages. In: Wooder JH, Corcoran J (eds) Logic, semantics, metamathematics (1983). Hackett Publishing Company, Indianapolis, pp 152–278
Thompson VA, Prowse JA, Turner GP (2011) Intuition, reason, and metacognition. Cogn Psychol 63:107–140
Turing A (1936) On computable numbers, with an application to the Entscheidungsproblem. In: Davis M (ed) The undecidable (1965). Raven Press, New York, pp 115–151
Van Benthem J, Doets K (1983) Higher-order logic. In: Gabbay D, Guenthner F (eds) Handbook of philosophical logic, vol 1. D. Reidel Publishing Company, Dordrecht, pp 275–329