Tax Law and Flexible Formalizations

Sarah B. Lawsky[*]

* * *

Introduction

Changing technologies render tax law's intricacy legible in new ways. Advances in large language models, natural language processing, and programming languages designed for the domain of tax law make formalizations, or "representation[s] of [ ] legislation in symbols[ ] using logical connectives," of tax law that capture much of its substance and structure both possible and realistic. These new formalizations can be used for many different purposes—what one might call *flexible formalizations*. Flexible formalizations will make law subject to computational analysis, including creating automated explanations of the analysis and testing statutes for consistency and unintended outcomes.

This Essay builds upon existing work in computational law and digitalizing legislation. After briefly describing what these formalizations might look like, this Essay focuses on four possible outcomes of flexible formalizations for tax law. First, formalizations at the statutory drafting stage could shift interpretive discretion from the executive (including administrative agencies) and judicial branches back to the legislative branch. Second, formalization of existing law may facilitate the development of abusive tax shelters, including making it possible for people with less tax sophistication to develop and defend shelters, which could force changes in how the law addresses tax shelters. Third, flexible formalizations could reduce some kinds of costs for administrators, simplifying currently time-consuming tasks and increasing transparency. And finally, while formalization could increase the complexity of the law itself, it would also make that complexity more manageable.

Flexible formalizations are not artificial intelligence (AI) in the sense of big data or machine learning. But machine learning might

---

[*] Howard Friedman '64 JD Professor of Law, Northwestern Pritzker School of Law. Thanks to Joshua Blank, Erin Delaney, Michelle Falkoff, and Denis Merigoux for helpful conversations and for comments on earlier drafts.

help produce flexible formalizations, and blending rule-based computation with machine learning could change both substantive tax law and tax enforcement and administration more powerfully than either alone. While this Essay focuses on tax law, similar considerations would come into play for any rule-based, complex area of law.

I. Flexible Formalizations

Formalization is not new to tax law. Tax law in the United States has long been formalized in, for example, Internal Revenue Service (IRS) forms, which provide computational steps for taxpayers to follow in order to comply with the law. Tax forms capture parts of the underlying law and then appear in various third-party programs that allow people to compute their tax liability. But these are *rigid formalization*s, by which I mean they reduce the law to algorithms (formalization) in a way that is useful only or primarily for the precise purpose for which the formalizations were created (rigid). In contrast, *flexible formalizations* are formalizations of law that are not tailored to a particular use but rather serve as the underpinning of any kind of task for which formalization or computer analysis is helpful. While this Essay primarily focuses on the legal implications of flexible formalizations, a few words first about what these formalizations should look like.

First, the envisioned formalizations would be machine-*executable*, not just machine-*readable*. The U.S. Code and accompanying regulations are already, in some sense, currently accessible by computers. Both the Code and regulations are available in XML, a machine-readable markup language that imposes labels and structure on text. (The XML scheme used for the U.S. Code, United States Legislative Markup, or USLM, is similar to but distinct from Akoma Ntoso, an internationally accepted XML scheme designed for standard representation of legislative and other legal documents.) These XML versions make legal text accessible (to, for example, free websites) and are used for many purposes (such as producing free selected Code and regulations books). But, having legal text in XML, whether through the USLM or Akoma Ntoso scheme, does not immediately permit computation on the underlying law. Flexible formalizations would.

Additionally, the law should be formalized using formal methods. That is, the law should be formalized in a way that, among other things, allows the outcomes of the formalized law to be mathematically demonstrated through proofs on the underlying formalization. This approach is consistent with the call for software that is secure by design in President Joe Biden's National Cybersecurity Strategy and the National Science Foundation's increased funding for the development of formal methods. Using formal methods for flexible formalizations would mean that the programs, given sufficiently precise specifications, will behave exactly as expected. Unlike large language models and generative AI, these formalizations will not be probabilistic.

The balance of this Essay focuses not on the mechanics or implementation of flexible formalizations, as interesting as that is, but rather on how this formalization will or should shape the law. This Essay assumes that some or all tax laws will be flexibly formalized, either at the drafting stage or later, and the formalization will be executable by a computer for many purposes. Each of the next Parts assesses an implication of this premise for tax law.

II. Shifting Discretion

I am not, of course, arguing that law should be, or even could be, entirely computer code. Some parts of the law are usefully formalized, and some are not. Some laws suggest, benefit from, or even require algorithms to be implemented, but law is more than just algorithms. However, formalizing law as part of the drafting process for tax laws could have significant benefits.

Tax statutes, like most (if not all) statutes, contain intended ambiguities (such as the meaning of a term like "gift" or "income"). Intended ambiguities are entirely consistent with formalizing law, and ambiguities will remain even after the law is formalized. One common way to handle a term with an ambiguous meaning is to leave the term as an input. The formalization might take no position on what, for example, constitutes a gift; it would simply have a place for the user to enter the amount of the gift once the person figures out what that number should be using the instructions to a tax form or other guidance.

This is the current approach on tax forms for various arguably ambiguous terms. For example, Form 1040 requires each taxpayer to enter the amount of gain or loss from the sale of capital assets, but what constitutes a capital asset is not part of the formalization; rather, it is discussed in the instructions. Similarly, there are lines on the form to enter various kinds of "income," but it is up to the taxpayer to determine what counts as income (or is excluded under rules such as the exclusion for gifts).

Tax statutes, however, also sometimes contain what seem to be unintended ambiguities (such as the order in which calculations are supposed to be performed or whether a particular portion of the statute is part of a "definition"). Often, the statutes cannot be applied at all without resolving these ambiguities. Tax statutes also sometimes contain outright errors. Currently, the Treasury Department and the Internal Revenue Service resolve many of the ambiguities and fix many of the errors, sometimes openly, sometimes sub silentio. Some of these fixes are noted in rulings or notices; others are simply implemented in forms. Courts also end up resolving some of the ambiguities if taxpayers object to Treasury's approach.

If Congress formalizes proposed legislation as part of the drafting process, many of these ambiguities and errors would be detected before the law is ever enacted. Some of the ambiguities would be necessary to resolve in order to code the law. And errors could be brought to light when the formalization of the entire law is tested to check outcomes.

To give a concrete example, in 2017, two tax law provisions were changed simultaneously. A new credit was added in 26 U.S.C. § 24 to be available for taxable years 2018 through 2025. Due to a winding path through the statute, the ability to claim this credit depended, in some circumstances, on a person's earning less than the "exemption amount" as defined in 26 U.S.C. § 151(d). This connection made sense because the new credit was partially intended to replace the personal exemption. However, at the same time this new credit went into § 24, Congress added a new paragraph under § 151(d)(5), stating that the personal exemption amount was $0 for taxable years 2018 through 2025. As a result, nobody would ever be eligible to take the new credit—clearly not what Congress intended! (And the IRS did fix this.) If the draft legislation had been formalized, straightforward testing

would have revealed the error, allowing Congress to make the fix before the bill became law.

This example is, in some sense, a silly error; presumably, most people think administrative agencies should have the power to fix an error like this one. But resolving other errors or ambiguities can be more of a judgment call, and currently, those judgment calls are being made by agencies or, sometimes, by courts—not by Congress. Congress's use of flexible formalizations during the drafting period could thus shift power back to itself.

III. Accessible Avoidance

Formalizing tax law could also make tax avoidance and evasion easier for taxpayers. Some abusive tax transactions involve cleverly combining the provisions of the tax law to comply with the literal language of the law but nonetheless reduce tax in ways unintended by Congress. The complexity of the law is currently a friction that makes finding new such tax shelters difficult. But formalization of tax law, making the law accessible to computational analysis and combinations, could make finding tax shelters trivial—or at least much easier.

Tax shelters are now both difficult to develop and sometimes difficult to shut down. The usual cadence is that someone, often an accounting firm, finds some combination of law and facts that allows taxpayers to pay less tax in a way not intended by Congress. Taxpayers engage in the shelter for some amount of time, and then the IRS discovers the shelter, perhaps through a combination of tax filings and audits or perhaps because the shelter comes to its attention in some other way (such as through a news article or a whistleblower). The IRS can then try to prevent the shelter administratively by, for example, challenging it in an audit or, more broadly, classifying the shelter as an abusive or "listed" transaction. But, the taxpayer can still choose to take the transaction to court, and while some courts are willing to use judicial doctrines to strike down the shelter, other courts will uphold a transaction that complies with the letter of the law. All of this takes an enormous amount of time and effort on the part of the government.

If tax shelters become relatively easy for taxpayers and practitioners to discover because the law becomes executable by

computer, with the marginal cost of developing a new tax shelter essentially nothing, the costs to the government could be enormous. (Indeed, one imagines that if some private firm already had access to formalized tax law and were using it to develop tax shelters, it would be very interested in keeping this secret. It could simply roll out a new tax shelter, make money off the shelter for years until the shelter is detected and shut down, and then, at no marginal cost, roll out another tax shelter.) With the unintended interactions and gaps in the tax law laid bare to taxpayers, the government will take a significant hit to the fisc in the absence of new approaches to detecting and managing tax shelters. Removing complexity as a friction slowing down the development of tax shelters (though not, of course, actually removing complexity, as discussed below) will force changes to the law surrounding tax shelters.

First, Congress could codify a general anti-abuse doctrine. Plenty of anti-abuse provisions already exist in the Code (and, indeed, in the regulations), in which the law bans certain actions done, for example, if the principal purpose is to avoid tax. While some courts are willing to use judicial doctrines such as the economic substance doctrine or "substance over form" to strike down tax planning that meets the letter of the law, other courts are not. Congress has not yet codified the economic substance doctrine; 26 U.S.C. § 7701(o) stops short of saying the economic substance doctrine is permitted or even required. Rather, § 7701(o) is styled as a "[c]larification" that provides conditions under which a transaction could be found to have economic substance *if* the economic substance doctrine is relevant.

Under current law, therefore, a judge could still decide that the economic substance doctrine and other judge-made anti-abuse doctrines are not relevant—ever. With a general anti-abuse provision that bans a broad range of transactions if certain standards are met (for example, if the principal purpose is to avoid tax and the transaction goes against Congress's intent, or a provision that bans noneconomic losses absent a clear declaration by Congress permitting such a loss) or some other standard-based approach, Congress could require courts to evaluate and strike down transactions that are abusive, even if the transactions meet the letter of the law.

Second, Congress could use flexible formalizations to reduce tax shelters without relying on courts. While I have imagined in this Essay

that the law will be flexibly formalized as part of a drafting process, that is not necessary for the law to end up formalized. Private parties can proceed with these formalizations on their own, and from a tax compliance perspective, the worst-case scenario is private parties' having access to these tools while the government lags behind. Therefore, another enforcement approach to managing tax shelters is for the government to use formalizations to generate tax shelters, before ever spotting them in the wild, for the purpose of banning them. Congress could give the Treasury direct authority to generate and ban such shelters. The government could also potentially use these formalizations to spot previously unidentified tax shelters when reviewing returns.

Finally, Congress could use these formalizations to prevent laws that create the potential for tax shelters from ever being enacted. In addition to identifying ambiguities and outright errors during the drafting process, Congress could check how envisioned changes would interact with the rest of tax law, determine whether proposed legislation would create opportunities for tax shelters, and modify the language of the statute to avoid those shelters.

As a side note, it would be *ineffective* to try to ban computer-created tax shelters. Computers will be used to generate tax shelters; they are likely already used to generate tax shelters. It would be nearly impossible to define what using a computer to generate a shelter means—would using Microsoft Excel count? And it would be incredibly difficult, if not impossible, to prove whether a computer had been used even if a definition of "using a computer" or "using AI" could be determined. There are, as discussed, various ways for the government to prevent, detect, or strike down tax shelters. But singling out computer-generated shelters is not, as a practical matter, one of those ways.

IV. Reducing Administrative Costs and Increasing Transparency

Flexible formalizations could also reduce administrative costs. For example, the IRS often must create new forms when a new law is passed. I focus here not on the visual design of the form, such as font and color, but on the algorithm implemented by the form: what numbers the taxpayer must enter, what calculations must be

performed on those inputs, and so forth. (Taxpayers can already modify the appearance of forms or submit substitute forms within certain limits.)

The algorithms on the forms are often extremely well designed and efficient for the taxpayer, but creating such forms presumably requires significant effort. If the law were already formalized, creating the forms would be nearly effortless—indeed, no forms would be necessary. No additional work would be needed to convert the law into algorithms. Taxpayers could simply use a computer program to enter whatever inputs are necessary to calculate, for example, tax owed.

This approach might appear to reduce transparency: the numbers go in, and the answer pops out—with no sense of what calculations are being executed. However, this is not so different from current tax filing. The vast majority of taxpayers file electronically and thus already have the experience of putting numbers in and getting an answer out, with no actual computations on the part of the human being. Nonetheless, current electronic filings do result in filled-out tax forms, which expose the reasoning and algorithms that resulted in the final number.

But in fact, flexible formalization could *increase* transparency. The computer code underlying the calculations should be completely public and transparent. And beyond that, as the computer code itself would remain impenetrable to many people, the code should be used to create explanations, written in a style as technical or nontechnical as desired, that could accompany the numerical bottom-line output.

Moreover, tax forms now sometimes resolve ambiguities or errors within the law, but the forms and their instructions contain no hint that these ambiguities or errors even exist. The code used in formalizations could follow the structure of the actual law closely, with lines of computer code matching up to each particular provision of the statute; the computer code itself could also include explanations of where, for example, the coders have had to resolve ambiguity inherent in the statute. Generated explanations could note those resolutions if the resolutions affected the taxpayer's bottom line—whether in favor of the taxpayer or otherwise. These generated explanations would capture, in a meaningful sense, the actual steps taken to reach the result. Formal methods, unlike generative AI, are deterministic, not

probabilistic. Formal proofs—that is, machine generated proofs—can be used to generate natural-language descriptions of the steps the program has taken to reach its conclusions.

V. Increasing Complexity—and Comprehension

Professor Lawrence Zelenak has argued that the ease of using tax preparation software has led to tax law's increased complexity and that this increased complexity is bad because it "turn[s] the income tax into a black box, the inner workings of which are incomprehensible to the average taxpayer, thereby undermining both the democratic legitimacy of the tax system and the ability of taxpayers to engage in informed tax planning." Zelenak proposed that Congress should "resist the lure of tax complexity."

Flexible formalizations may well lead to even greater tax complexity. For example, tax provisions for businesses and high-net-worth individuals, such as the § 199A deduction for "qualified business income," could become even more arcane and elaborate. But this complexity will not raise the same concerns that Zelenak identifies. One reason flexible formalizations could lead comfortably to greater tax complexity is that even as provisions become complex, drafters who formalized the legislation can be confident that the different parts of the provision fit together and that the provision fits with the overall law. Lawmakers can test formalizations, looking for consistency within the law and confirming that the outcome of the proposed provisions is as desired.

And what it means to understand the law may also change. Reading a complex provision "cold" is already very difficult. Tax lawyers rely on secondary sources to help guide them through statutory language. Lawyers will never be able to stop reading the actual statutory language. And lawyers soon will either need to be able to understand formalizations themselves or work with people who can. But flexible formalizations may also be able to automatically generate explanations that carefully track the underlying steps that lead from a set of facts, through the statute, to a particular outcome. The lawyer could then follow those steps, reading the language carefully and considering how it applies to the lawyer's client, while also evaluating whether the formalization and its conclusion seem correct (just as

lawyers should already evaluate whether secondary sources are correct). Lawyers do not read unmediated statutes now, nor will they in the future. But automated help through the statute, if based on precise formalizations, could provide clear paths for the lawyer to follow in the future, unlike the scattershot guidance now available from legal research tools.

Conclusion: Locating Expertise

As this Essay has discussed, using formal methods to create flexible formalizations of tax law will likely change the substance, enforcement, and administration of tax law. The considerations outlined in this Essay are not limited to tax; they would likely apply to any complex, rule-heavy area of law, such as student financial aid, immigration law, or sentencing.

To take full advantage of the potential of flexible formalization, and to avoid the dangers of formalization residing only in private hands, the government should develop the ability to formalize the law and to use these formalizations fully. These formalizations cannot be reserved for private actors, and the government cannot rely on contractors (as it did, for example, in the catastrophic 2024 revised FAFSA rollout and the 2013 rollout of HealthCare.gov) or private industry (such as, for example, TurboTax and other private tax return preparers). Rather, the government should have in-house experts in all parts of the government who can formalize the law and manage and take advantage of those formalizations. Any formalizations should also be public so that the general public can help test, challenge, and improve those formalizations. The relevant expertise will thus be located within the government itself as well as with the public—a people's formalization.

* * *

Sarah B. Lawsky is the Howard Friedman '64 JD Professor of Law, Northwestern Pritzker School of Law.