

# At the Boundary of Law and Software: Toward Regulatory Design with Agent-Based Modeling

Sebastian Benthall<sup>1</sup>, Michael Carl Tschantz<sup>2</sup>, Erez Hatna<sup>3</sup>, Joshua M. Epstein<sup>3</sup> and Katherine J. Strandburg<sup>1</sup>

<sup>1</sup>New York University, School of Law, 40 Washington Sq So, 10012, New York, USA

<sup>2</sup>International Computer Science Institute, 2150 Shattuck Avenue, 94704, Berkeley, USA

<sup>3</sup>New York University, School of Global Public Health, 708 Broadway, New York, NY 10003

## Abstract

Computer systems that automate the making of decisions about people must be accountable to regulators. Such accountability requires looking at the operation of the software within an environment populated with people. We propose to use agent-based modeling (ABM) to model such environments for auditing and testing purposes. We explore our proposal by considering the use of ABM for the regulation of ad targeting to prevent housing discrimination.

## Keywords

agent-based modeling, accountability, regulation, automated decision making

## 1. Introduction

The use of agent-based models (ABMs) can improve software accountability by representing populations of actors (e.g., house buyers, job seekers, college and insurance applicants) affected by software. Unaccountable software can embed and exacerbate societal biases or have other pernicious effects. ABMs can be used to explore the societal effects of software systems to aid the design and enforcement of regulations. Currently, software accountability measures are largely defined very narrowly, as mechanical compliance with regulations, written without a systematic way of predicting societal impact. ABMs are a way to model the societal impact of software and to correct or design regulations accordingly. We argue, in essence, that agent-based models (ABMs) of the interactions between software systems, those systems' social environments, and applicable regulations can help to improve software accountability. We focus on software systems that employ personal data, have significant impacts on individuals, and are subject to regulation or used within regulatory agencies.

For example, anti-discrimination regulations in the housing context focus on societal goals such as fairness in housing or the reduction of residential segregation. Some of these regulations

---

AMPM'21: First Workshop in Agent-based Modeling & Policy-Making, December 8, 2021, Vilnius, Lithuania

✉ spb413@nyu.edu (S. Benthall); mct@icsi.berkeley.edu (M. C. Tschantz); eh109@nyu.edu (E. Hatna); joshua.epstein@nyu.edu (J. M. Epstein); katherine.strandburg@nyu.edu (K. J. Strandburg)

🌐 <https://sbenthall.net> (S. Benthall); <https://www.icsi.berkeley.edu/~mct/> (M. C. Tschantz)

🆔 0000-0002-1789-5109 (S. Benthall); 0000-0003-1367-3784 (M. C. Tschantz); 0000-0002-5989-3224 (J. M. Epstein); 0000-0003-3056-9989 (K. J. Strandburg)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



CEUR Workshop Proceedings (CEUR-WS.org)

are applicable to the software systems that target online housing ads. These systems, typically, are written not by regulators but by private sector software designers. The resulting deployed advertising systems may produce effects that are inconsistent with – even subverting – the regulatory intent. One possible reason for such an outcome is that the *social environment* in which the software is deployed is absent from the analysis of software impact. We want to include it and make it accessible to the regulators. If the social environment were represented, auditors could better anticipate whether the software is likely have the intended social effect. We propose to use ABMs to fill this gap, making software more accountable in this sense.

This sort of transformation has occurred in other fields. Infectious disease modeling is one. Faced with a novel pathogen, like Swine Flu or SARS-CoV-2, it is crucial to have some way to estimate how fast it may spread, how much vaccine to produce, whether to ban international travel or close schools and workplaces. Before the advent of infectious disease transmission models, doctors and public health official were operating in the dark. Now, we have disease simulation models at scales from the local to planetary for forecasting and mitigation. They are part of the fabric of public health decision making, and are used to inform policy at the CDC, NIH, WHO, and many national governments. They are not crystal balls and do not always agree, as in weather forecasting. But collectively, they can bound our uncertainties, estimate sensitivities, explore tradeoffs, and offer headlights in uncertain settings. Agent-Based Models specifically, which can include social networks and cognitive factors, are increasingly used. ABMs have the enormous advantage that they are also visual and rule-based (not equation-based) allowing non-technical audiences and domain experts to understand their results and, indeed, to participate in their construction. At the same time, they can be calibrated to epidemic data. The net effect of all this is that ABMs can have higher impact than complex differential equation models. We are proposing to do this for software accountability.

We see potential in this method because ABMs are legible to software engineers, social scientists, and regulators. We have also identified key challenges to this approach, which are the potential politicization of model choice, the selection of appropriate robustness metrics, and the design of the interface between the ABM and audited software.

## **2. The urgent challenge of software accountability**

Software's importance as an object of regulation is increasing as ever more individually and socially significant private sector activities are automated. Controversial examples include the use of automated decision tools in employment [1, 2], lending [3], targeted advertising [4, 5] and higher education [6]. Increasingly, the use of personal data by information services is being regulated by expansive data protection laws such as the E.U. General Data Protection Regulation (GDPR) and California's combined Privacy Rights Act (CPRA) and Consumer Privacy Act (CCPA), whose interpretations are largely unsettled.

Applying regulatory standards created for human actors to automated systems may fail to further the policy goals of the regulatory regime. Software poses critical challenges to traditional policy and regulatory approaches to accountability, both as an object and as a tool of regulation [7, 8]. Most proposals for meeting these challenges have focused on increasing transparency regarding when and how software is used, its provenance and verification, relationships be-

tween inputs and outputs, or the software code itself. These efforts are complicated by the difficulty of adequately explaining software, even to domain expert regulators who are not trained in computer science [9, 10]. Suggested approaches to this problem include “algorithmic impact assessments,” which systematize consideration of the potential effects of automation on regulatory outcomes [11, 12, 13], summaries modeled on nutrition labels [14, 15, 16, 17] and various forms of black box testing [18, 19]. While these approaches often require regulators to gather input from affected individuals, communities and institutions, they lack a systematic mechanism for using such input to explore the complex interactions between the software, other aspects of the regulatory system and the affected *social systems*.

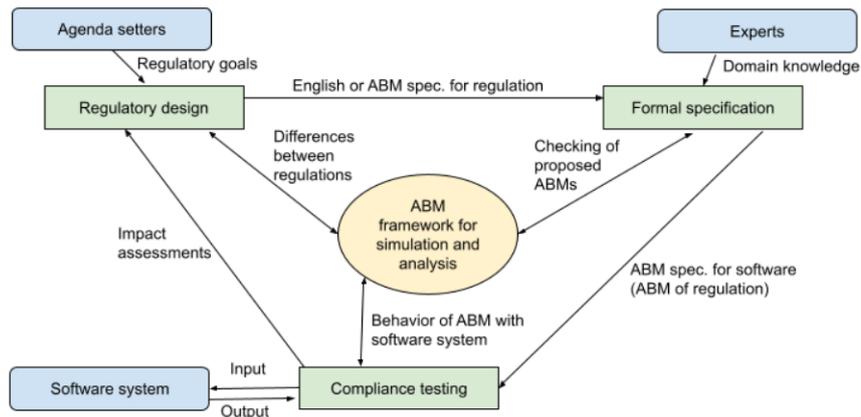
We posit that there are three elements to the design of accountable software systems. The first two are the software system itself and the regulatory environment, including regulations, regulators, and their enforcement mechanisms. Our key insight is that the third, the social environment in which the system operates, is neglected in current software accountability techniques. The appropriateness of the software given a regulation, and in light of regulatory goals, depends not just upon the software’s behavior but upon its impacts on people in the social environment. Accountable software regulation thus requires expertise relating to all three elements and collaboration between software engineers, regulators, and domain scientists, as well as mechanisms for oversight by the legislature and ultimately the public.

### **3. ABMs as an answer to software accountability’s challenges**

ABM is already part of the policy-making toolkit [20]. ABMs may have additional advantages when software is a regulatory object or tool. Coglianese and Lehr [21] argue that as “regulation by robot”, or regulations designed and implemented using machine learning, become more common, other analytic techniques will need to be used to test them. In particular, they recommend the use of ABMs or multi-agent systems (MAS) to test the impact of regulations on the complex social environment.

By modeling the social environment, ABMs can capture regulatory concepts difficult to express in terms of software alone, such as people’s beliefs and purposes, and how regulations are often designed to be open to ongoing reinterpretation as new circumstances arise. We suggest that this method of testing the social impact of a software system can be more intuitive to regulators and domain experts than other means for projecting likely impact. ABMs can also more accurately account for social complexity and feedback loops, including unintended ones not readily apparent from the behaviors of individuals or the software acting in isolation. Furthermore, ABMs are implemented as explicit programs, making them intelligible to software engineers, thereby facilitating dialogue between them and regulators about policy goals and the interpretation and meaning of regulations. The difficulty of expressing the accountability goals of software in the traditional language of software testing, test suites expressed as the approved outputs for a given input, inspires the turn to ABMs.

As a concrete example, the goal of contact tracing is to slow the spread of a disease by identifying people who have likely been exposed to it and notifying them for testing, treatment, and/or isolation. Contact tracing may also be subject to policy making by public health departments, various privacy regulations, and protections of confidentiality. In the COVID-19

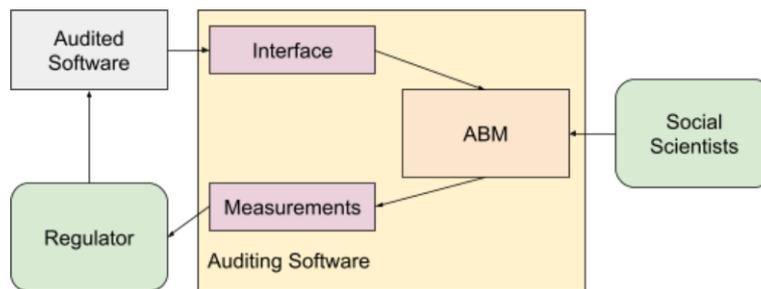


**Figure 1:** Workflow with our three proposed components (regulatory design, formal specification, and compliance testing) as stages of a general method for enforcing software accountability. This figure illustrates one potential configuration of these components with respect to the regulation of a private sector software system.

pandemic, many countries controversially turned to smartphone based contact tracing, despite public outcry in places about inappropriate surveillance. To understand whether an app is compliant with this regulatory environment requires information about the social environment. Policymakers may also seek to encourage or mandate the use of contact tracing. Software designed to be responsive to privacy concerns might increase the public adoption of the apps. ABMs can model this complex social environment, calibrated to empirical data to predict social outcomes, and provide information necessary to assess compliance. Because ABMs are software objects, they may be coupled directly with regulated software systems or model their social and behavior impact.

We picture, for example, a regulator that has been entrusted by the public and legislature (the “agenda setters”) with social goals in a particular domain. The regulator works with domain scientists, who provide a descriptive ABM that includes the entities important to regulations, such as actors and social outcomes. The regulator introduces normative elements into the model, which may be hard rules on agent behavior or objective functions to be tracked and optimized. This model can be used to analyze the intended and unexpected consequences of the regulation.

We propose a regulatory design process that uses ABMs to enhance accountability by spanning the boundaries between law, software, domain expertise, and ultimately, the public (Fig. 1 & 2). ABMs can be developed to simulate the social effects of such a private software system and thus evaluate its implications for regulatory goals. Regular reports to government agencies on the social impact of software systems (“Impact Assessments”), and reports responding to investigations, are an unsettled requirement of many laws governing software systems including the CCPA/CPRA, the GDPR, and the E.U.’s more recently proposed technology regulations such as the Digital Services Act, Digital Markets Act and Artificial Intelligence Act. There remain many open questions about the legal and technical feasibility of software that allows regulatory



**Figure 2:** Schematic of how an ABM, designed by social scientists, can form the core of auditing software used by regulators to analyze the social impact of an audited software.

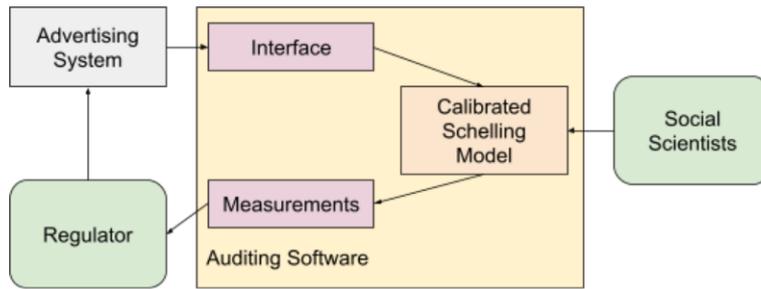
users to interact with and visualize results from the model. A key aspect of this workflow are methods for testing the robustness of the system’s compliance to social changes, modeled as both endogenous change within an ABM and as variation of the ABM’s performance over variations in its parameter space. These will be truly new capabilities.

#### 4. Example: Housing advertising

The Fair Housing Act and related regulations prohibit certain discriminatory housing practices, including ad targeting that would “deny particular segments of the housing market information about housing opportunities” based on “race, color, religion, sex, handicap, familial status, or national origin.”<sup>1</sup> Prior to 2019, Facebook’s ad targeting options included targeting based on Facebook-constructed attributes, including “ethnic affinity” (later renamed “multicultural affinity”). Though based on users’ activities on the platform, “ethnic affinity” could obviously have the effect of targeting ads unevenly across racial groups. In fact, Speicher et al. [22] demonstrated that, even without using an attribute such as “ethnic affinity,” it was not difficult to obtain biased targeting, intentionally or unintentionally, using Facebook’s targeting tools. Related work has shown similar issues with Google’s advertising platform for race [23] and gender [24, 4]. Facebook later changed its practices in response to a charge of discrimination from the Department of Housing and Urban Development (HUD v. Facebook, 2019) and in settlement of related class action litigation. Thus, there has been no judicial ruling on whether its past or current automated ad targeting violates the FHA [5].

One well-studied area of AI accountability is machine learning fairness, which has been developed in part to support the accountability of software to nondiscrimination laws. A result is that different fairness metrics are appropriate under different assumptions about bias in the data generation process [25, 26, 27]. There are also concerns about the “ripple effects” of automated decision-making, such that an automated decision becomes involved in a feedback loop with the social environment it’s acting upon [28]. With just access to the inputs and outputs of an ad targeting program, an auditor can merely detect bias in the presentation of advertisements (e.g., [23, 24]). Adding a calibrated ABM of housing choice, lets the auditor additionally predict

<sup>1</sup>24 CFR §100.75(c)(3)



**Figure 3:** Schematic of the architecture of an ABM-based auditing system designed specifically for anticipating the impact of an advertising system on segregation.

how such biases play out in society, including feedback loops, as represented in the ABM. For example, the ABM could show how the biased ads increase segregation. We are not aware of any auditors combining software and ABMs in this way.

ABMs have, however, long been used to study housing segregation, beginning with models based on ethnic homophily and over time introducing models of market dynamics [29, 30, 31]. The Schelling Segregation Model [29] uses a square grid of residences, most of which house an agent. The main counterintuitive result of this model is that the relationship between the level of in-group preference and the amount of segregation in aggregate is non-linear. Even a weak preference for in-group neighbors can generate strong segregation. For example, with 2 groups and a 30% preference for in-group neighbors, the model will converge to near 75% of all neighboring pairs being of the same group. There are many variations on the Schelling segregation model, including more groups, variable preference rates, more realistic land topologies, and market dynamics [32].

We will develop a modified Schelling segregation model that includes the effects of advertising which may influence the residential behavior of agents and the socioeconomic characteristics of neighborhoods.

The new modified model can be designed to include parameters representing the degree to which advertising is targeted based on (a) race and ethnicity, (b) location, and (c) socio-economic status. Measurements of interest will include metrics for segregation along similar axes. A contribution of the model will be that it can reveal the extent to which targeting based on, for example, socio-economic status contributes to segregation based on race and ethnicity. Rather than rely on point estimates to predict these effects, through multiple simulations the model can be tested over its entire parameter space. The results of this simulation can then be analyzed to determine effects and phase transitions on the dependent variables of interest (such as positive and negative rates on groups; positive and negative error rates; and so on).

## 5. New challenges

We have identified several key challenges to the proposed approach. These pain points are each opportunities for future technical and policy research.

One challenge is designing and selecting the ABM for testing. Modeling the full complexity of the agents interacting with or affected by software may be beyond an auditor's capabilities and knowledge. The auditor will have to make simplifying assumptions to produce an ABM. If the ABM is consequential for regulation, its details will entail different outcomes for different stakeholders. What process can improve the scientific accuracy and objectivity of the simulation results?

A related challenge is identifying the key metrics for translating ABM output into policy-relevant information. The output of an ABM can be very sensitive to the calibration of its input parameters, choices about its rules, as well as its own endogenous dynamics. Policy decisions should be based on robust outcomes of the model, rather than on brittle point estimates or single examples. The use of ABMs for software accountability requires well chosen metrics (e.g., statistical indices of spatial segregation) for outcome robustness.

Lastly, we have identified a challenge at the technical interface between the audited software and the ABM. This part of the accountability process is likely to be least transparent to regulators, as it will be informed by the technical expertise of both the domain scientists and the audited system engineers. The choice of which system outputs are entered into the ABM is a sensitive one for which there is little preexisting guidance. For example, in our housing advertising example, the auditing mechanism requires measurements of the degree to which the advertising system discriminates with respect to race, location, and socioeconomic status. While it is possible to take these measurements using black box testing techniques with synthetic web user profiles, the resulting measurements are likely to vary with the technical specifics of the measurement technique. It would be better to have general guidelines and standards for how to design these interfaces between the audited software and the ABM.

Beyond these scientific and technical challenges is the fact that, in an area as politically fraught as fair housing, the politicization of model design (e.g., behavioral assumptions regarding different social groups) is a distinct possibility. The fact that all assumptions are *explicit* and all runs are *replicable* will help minimize this risk.

## 6. Conclusions

Agent-based models are a promising solution to one significant obstacle to software accountability: the modeling of the social environment of the software system. We envision a regulatory process that embeds ABMs to bridge the expertise of software engineers, regulators, and domain social scientists. In this article, we have developed one scenario for the use of ABMs for software accountability: the testing of advertising system's effects on housing segregation. By developing this example, we have identified several key challenges to our proposed approach. We will address these challenges in future research.

## Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. 2131532, 2131533, and 2105301. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of

the National Science Foundation. One of the authors of this article is supported by the NYU Information Law Institute's Fellows program, which is funded in part by Microsoft Corporation.

## References

- [1] P. T. Kim, Big data and artificial intelligence: New challenges for workplace equality, *U. Louisville L. Rev.* 57 (2018) 313.
- [2] I. Ajunwa, D. Greene, Platforms at work: Automated hiring platforms and other new intermediaries in the organization of work, in: *Work and labor in the digital age*, Emerald Publishing Limited, 2019.
- [3] M. A. Bruckner, The promise and perils of algorithmic lenders' use of big data, *Chi.-Kent L. Rev.* 93 (2018) 3.
- [4] A. Datta, A. Datta, J. Makagon, D. K. Mulligan, M. C. Tschantz, Discrimination in online advertising: A multidisciplinary inquiry, in: *Conference on Fairness, Accountability and Transparency*, PMLR, 2018, pp. 20–34.
- [5] M. Byrd, K. J. Strandburg, Cda 230 for a smart internet, *Fordham L. Rev.* 88 (2019) 405.
- [6] E. Zeide, The structural consequences of big data-driven education, *Big Data* 5 (2017) 164–172.
- [7] D. K. Citron, Technological due process, *Wash. UL Rev.* 85 (2007) 1249.
- [8] K. J. Strandburg, Rulemaking and inscrutable automated decision tools, *Columbia Law Review* 119 (2019) 1851–1886.
- [9] A. D. Selbst, S. Barocas, The intuitive appeal of explainable machines, *Fordham L. Rev.* 87 (2018) 1085.
- [10] R. Brauneis, E. P. Goodman, Algorithmic transparency for the smart city, *Yale JL & Tech.* 20 (2018) 103.
- [11] F. McKelvey, M. MacDonald, Artificial intelligence policy innovations at the canadian federal government, *Canadian Journal of Communication* 44 (2019) PP43–PP50.
- [12] M. E. Kaminski, G. Malgieri, Multi-layered explanations from algorithmic impact assessments in the gdpr, in: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020, pp. 68–79.
- [13] J. Metcalf, E. Moss, E. A. Watkins, R. Singh, M. C. Elish, Algorithmic impact assessments and accountability: The co-construction of impacts, in: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021, pp. 735–746.
- [14] M. Hintze, In defense of the long privacy statement, *Md. L. Rev.* 76 (2016) 1044.
- [15] S. Holland, A. Hosny, S. Newman, J. Joseph, K. Chmielinski, The dataset nutrition label: A framework to drive higher data quality standards, *arXiv preprint arXiv:1805.03677* (2018).
- [16] M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, T. Gebru, Model cards for model reporting, in: *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 220–229.
- [17] J. Stoyanovich, J. J. Van Bavel, T. V. West, The imperative of interpretable machines, *Nature Machine Intelligence* 2 (2020) 197–199.
- [18] S. Wachter, B. Mittelstadt, C. Russell, Counterfactual explanations without opening the black box: Automated decisions and the gdpr, *Harv. JL & Tech.* 31 (2017) 841.

- [19] M. C. Tschantz, A. Datta, A. Datta, J. M. Wing, A methodology for information flow experiments, in: 2015 IEEE 28th Computer Security Foundations Symposium, IEEE, 2015, pp. 554–568.
- [20] S. Benthall, K. J. Strandburg, Agent-based modeling as legal theory tool, *Frontiers in Physics* 9 (2021) 337.
- [21] C. Coglianese, D. Lehr, Regulating by robot: administrative decision making in the machine-learning era, *Geo. LJ* 105 (2016) 1147.
- [22] T. Speicher, H. Heidari, N. Grgic-Hlaca, K. P. Gummadi, A. Singla, A. Weller, M. B. Zafar, A unified approach to quantifying algorithmic unfairness: Measuring individual & group unfairness via inequality indices, in: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2239–2248.
- [23] L. Sweeney, Discrimination in online ad delivery, *Communications of the ACM* 56 (2013) 44–54.
- [24] A. Datta, M. C. Tschantz, A. Datta, Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination, in: *Proceedings on Privacy Enhancing Technologies (PoPETs)*, De Gruyter Open, 2015, pp. 92–112.
- [25] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, On the (im) possibility of fairness, *arXiv preprint arXiv:1609.07236* (2016).
- [26] S. Yeom, M. C. Tschantz, Avoiding disparity amplification under different worldviews, in: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021, pp. 273–283.
- [27] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, The (im)possibility of fairness: Different value systems require different mechanisms for fair decision making, *Commun. ACM* 64 (2021) 136–143. URL: <https://doi.org/10.1145/3433949>. doi:10.1145/3433949.
- [28] A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, J. Vertesi, Fairness and abstraction in sociotechnical systems, in: *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 59–68.
- [29] T. C. Schelling, Dynamic models of segregation, *Journal of mathematical sociology* 1 (1971) 143–186.
- [30] I. Benenson, I. Omer, E. Hatna, Entity-based modeling of urban residential dynamics: the case of yaffo, tel aviv, *Environment and Planning B: Planning and Design* 29 (2002) 491–512.
- [31] A. Sahasranaman, H. J. Jensen, Ethnicity and wealth: The dynamics of dual segregation, *PloS one* 13 (2018) e0204307.
- [32] E. Hatna, I. Benenson, The schelling model of ethnic residential dynamics: Beyond the integrated-segregated dichotomy of patterns, *Journal of Artificial Societies and Social Simulation* 15 (2012) 6.