DO ROBOTS DREAM OF ELECTRIC LAWS?
AN EXPERIMENT IN THE LAW AS ALGORITHM

Lisa Shay
Assistant Professor
Department of Electrical Engineering and Computer Science
West Point

Woodrow Hartzog
Assistant Professor
Samford University's Cumberland School of Law

John Nelson
Assistant Professor
Department of English and Philosophy
West Point

Gregory Conti
Associate Professor
Department of Electrical Engineering and Computer Science
West Point

Due to recent advances in computerized analysis and robotics, automated law enforcement has become technically feasible. Unfortunately, laws were not created with automated enforcement in mind, and even seemingly simple laws have subtle features that require programmers to make assumptions about how to encode them. We demonstrate this ambiguity with an experiment where a group of 52 programmers was assigned the task of automating the enforcement of traffic speed limits. A late-model vehicle was equipped with a sensor that collected actual vehicle speed over an hour long commute. The programmers (without collaboration) each wrote a program that computed the number of speed limit violations and issued mock traffic tickets. Despite quantitative data for both vehicle speed and the speed limit, the number of tickets issued varied from none to one per sensor sample above the speed limit. Our results from the experiment highlight the significant deviation in number and type of citations issued during the course of the commute, based on legal interpretations and assumptions made by programmers untrained in the law. These deviations were mitigated, but not eliminated, in one sub-group that was provided with a legally-reviewed software design specification, providing insight into ways to automate the law in the future. Automation of legal reasoning is likely to be the most effective in contexts where legal conclusions are predictable because there is little room for choice in a given model; that is, they are determinable. Yet this experiment demonstrates that even relatively narrow and straightforward "rules" can be problematically indeterminate in practice.

Do Robots Dream of Electric Laws?
An Experiment in the Law as Algorithm

Lisa A. Shay,[1] Woodrow Hartzog,[2] John Nelson,[3] and Gregory Conti[4]

INTRODUCTION

We are rapidly entering an era when some robots will be commonly programmed to either enforce or comply with various laws. This is particularly true in many consumer, industry, and military contexts, such as with motor vehicle operations. Perceived cost savings, public safety gains, and law enforcement efficiencies drive this proliferation of automated law enforcement systems. We see these systems today in the form of automated traffic law systems that monitor automobile and driver behavior, capture evidence, and issue citations. Artificially intelligent robotic systems, such as driverless automobiles, are being employed in a variety of environments, thus demanding algorithmic logic that can parse and process laws like driving restrictions and take reasonable steps to maximize compliance. Neither automated law enforcement nor compliance is simple. Both aspects are rife with technical, ethical, and legal dilemmas.[5]

Of course, laws are not traditionally written with automated processing in mind. Instead, they are often necessarily ambiguous and continuously contoured by court decisions and evolving social norms. The digital code running in robotic systems requires extreme precision and rigor not resident in analog law. The key shortcoming is the ambiguous translation process between human-readable laws on the books and machine-processable algorithms required by automated systems.

This paper uses traffic laws to study the interplay between laws and automated enforcement and compliance. To better understand the challenges of implementing laws as code,[6] we constructed and executed an empirical experiment in which 52 computer

---

[1] Assistant Professor, Department of Electrical Engineering and Computer Science, US Military Academy at West Point.
[2] Assistant Professor, Cumberland School of Law at Samford University; Affiliate Scholar, Center for Internet and Society at Stanford Law School.
[3] Assistant Professor, Department of English and Philosophy, US Military Academy at West Point.
[4] Associate Professor, Department of Electrical Engineering and Computer Science, US Military Academy at West Point.
[5] For an excellent discussion of ethical considerations and potential metrics for success of automated cars, *see* Bryant Walker Smith, *Driving at Perfection*, STANFORD CENTER FOR INTERNET AND SOCIETY (Mar. 11, 2012), http://cyberlaw.stanford.edu/blog/2012/03/driving-perfection; *see also* Harry Surden, *The Variable Determinacy Thesis*, 12 COLUM. SCI. & TECH. L. REV. 1 (2011).
[6] JONATHAN ZITTRAIN, THE FUTURE OF THE INTERNET -- AND HOW TO STOP IT (2008). Zittrain suggests that technology will support three aspects of perfect enforcement of the law: preemption, specific injunction, and surveillance. *See also* Surden, *supra* note 1 (asserting that legal outcomes in certain contexts are amenable to resolution by computers).

programmers implemented several common traffic laws and determined violations based on real-world driving data. The variance in the resulting code and the number and frequency of citations based on programmer assumptions highlight the complications of algorithmic encoding. This paper explains the experiment and our conclusions. We conclude with an analysis of the implications of automating law enforcement and compliance and provide candidate solutions as well as open questions for future exploration. In our analysis, we explore the likely stumbling blocks that will be encountered when attempting to automate enforcement of or compliance with similar laws.

A deep understanding of the relationship between automated systems that enforce laws and robots that must comply with them is necessary and immediately applicable. Artificially intelligent consumer, industrial, and military robots are operating today, with usage growing rapidly. Accommodating analog laws in an automated system forces designers, purveyors, and users to accept significant risk and allows the opportunity for all parties to cut corners, with potentially dangerous consequences. Our analysis lays a foundation for systems that can better understand the law than through ad hoc coding and, because of this understanding, can better enforce and comply with it.

BACKGROUND

Industry is already developing and even fielding autonomous technologies that necessitate immediate automated law enforcement and compliance policies. For instance, in addition to Google's high-profile driverless car project,[7] Toyota is working on a semi-autonomous package for its cars, where the auto company foresees the future automobile with "an intelligent, always-attentive [human] co-pilot"[8] Certification of these hardware and software packages for legal compliance will be critical for highway safety. As Toyota Motor Corporation's Lexus Division recently announced its Advanced Active Safety Research Vehicle, it previewed the initiative's future: "While key components of these research efforts could lead to a fully autonomous car in the future, the vision is not necessarily a car that drives itself. Instead, Toyota and Lexus envision technologies that enhance the skills of the driver, believing a more skillful driver is a safer driver." [9] Oxford University is likewise developing a sensor & guidance package to automate driving,[10] and Hitachi is developing a one passenger robo-taxi.[11] These recent

[7] John Markoff, *Google Cars Drive Themselves, In Traffic,* NEW YORK TIMES (Oct. 9, 2010), http://www.nytimes.com/2010/10/10/science/10google.html.
[8] The Lexus Division issued the following press release on Ja. 7, 2013: http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/toyota-semi-autonomous-lexus-car-will-keep-you-safe/?utm_source=roboticsnews&utm_medium=email&utm_campaign=012213.
[9] *Id.*
[10] *Robotic driving system used to control Nissan Leaf*, THE ENGINEER (Feb. 15, 2013), http://www.theengineer.co.uk/sectors/automotive/news/robotic-driving-system-used-to-control-nissan-leaf/1015547.article.
[11] *Hitachi's Self-driving Robotic Car ROPITS*, SHIOTSU AUTOTRADE JAPAN (March 22, 1013), http://www.shiotsu-used-car.com/blog/hitachi-selfdrivingroboticcar-ropits.htm.

technological advances foretell a potential proliferation of robotic vehicles on our highways in the near future.

Many relevant stakeholders have increasingly been both captivated and troubled by the possibility of converting laws into a machine-readable format. Scholars have explored how contracts, statutes, and many other regulatory concepts might be made computable.[12] Notably, Harry Surden has proposed a variable determinacy thesis as a framework for computationally automating legal reasoning.[13] According to Surden, "automated legal reasoning systems that exist operate within particular legal contexts in which legal decisions tend to be relatively more determinate"; that is, legal conclusions are predictable because little room exists for choice in a given model.[14]

Surden notes the skepticism around automation of the law, stating:

> Scholars from the legal domain tend to insist upon a nuanced view of legal analysis. In this conception, legal reasoning is too imbued with uncertainty, ambiguity, judgment, and discretion to permit computerized assessment. This literature's common theme is that even if computers were technically able to mimic legal decision making in a mechanical fashion they would necessarily miss the subtle institutional, value-based, experiential, justice-oriented, and public policy dimensions that are the heart of lawyerly analysis.[15]

Yet Surden is more optimistic. He noted that automated legal analysis is more common than one might think, stating: "For example, the Federal Communications Commission (FCC) is investigating whether electronic devices can be made to automatically comply with government-issued spectrum management rules. Similarly, the government of Singapore has explored the possibility of automatically assessing architectural building designs for compliance with building code laws." [16] Surden recognizes the challenge of automated legal process, however, stating: "In

---

[12] *See, e.g.,* Harry Surden, *Computable Contracts,* 46 U.C. DAVIS L. REV. 629 (2012); Lorrie Faith Cranor and Joel R. Reidenberg, *Can User Agents Accurately Represent Privacy Notices?*, THE 30TH RESEARCH CONFERENCE ON COMMUNICATION, INFORMATION AND INTERNET POLICY (TPRC) (2002); Jeremy C. Maxwell, Annie I. Antón and Peter Swire, *A Legal Cross-References Taxonomy for Identifying Conflicting Software Requirements,* 19TH IEEE INTERNATIONAL REQUIREMENTS ENGINEERING CONFERENCE (RE'11), Trento, Italy (2011); *See* Danielle Keats Citron, *Technological Due Process*, 85 WASH. U. L. REV. 1249 (2008).

[13] Harry Surden, *The Variable Determinacy Thesis,* 12 COLUMB. SCI. & TECH. L. REV. 1 (2011).

[14] *Id.* at 5.

[15] Id. at 3 (citing Jeffrey Meldman, *A Structural Model for Computer-Aided Legal Analysis*, 6 RUTGERS COMPUTER & TECH. L.J. 27 (1977); Jon Bing, *Legal Norms, Discretionary Rules, and Computer Programs*, in COMPUTER SCIENCE AND LAW (Bryan Niblett ed., 1980); GUIDO GOVERNATORI & ANTONINO ROTOLO, AN ALGORITHM FOR BUSINESS PROCESS COMPLIANCE, IN LEGAL KNOWLEDGE AND INFORMATION SYSTEMS: JURIX 2008, 186 (2008); Kevin Ashley et al., *Symposium:Legal Reasoning and Artificial Intelligence: How Computers Think Like Lawyers*, 8 U. CHI. L. SCH. ROUNDTABLE 1, 19 (2001); ADAM WYNER & TEVEOR BENCH-CAPON, ARGUMENT SCHEMES FOR LEGAL CASE-BASED REASONING, IN LEGAL KNOWLEDGE AND INFORMATION SYSTEMS: JURIX 2007, 139 (2007)) .

[16] *Id.* at 4-5 ("Within the private sector, numerous corporations are investigating software aimed at automating business-compliance with health care, privacy, corporate, and financial laws. Within the academic realm, multiple

comparative terms, the number of legal contexts in which legal outcomes are tolerably determinate is probably somewhat small."[17]

Surden conceptualized determinacy as a "relative concept that exists along a spectrum rather than as a binary concept."[18] Drawing from the "rules versus standards" debate, Surden explained that different levels of linguistic abstraction can be used by lawmakers in attempting to regulate the same underlying behavior.[19] "Rules and standards can be seen as two poles of a particular dimension of abstractness. Most laws can be thought of as residing on a continuum between rules and standards, with some laws leaning towards the rule end, and others toward the standards end, often with no obvious distinction."[20] To demonstrate the difference between rules and standards, Surden used a classic example directly relevant to this experiment—unsafe driving laws. A "rule" to curb unsafe driving would be articulated as follows: "No one shall drive a vehicle faster than 65 miles per hour."[21] A standard to curb unsafe driving would be articulated this way: "No one shall drive a vehicle at unsafe speeds." The primary characteristic of a rule, then, is that it has a "strong degree of factual determinability."[22] According to Surden, "This means that the legal criterion or category is structured such that one can determine, with a relatively strong degree of certainty, whether a rule has been violated in a given factual situation."[23]

Surden's variable determinacy thesis is an ideal framework for use in analyzing the issues of automated enforcement and compliance relevant to this experiment. Surden asserted that, "[not] only can we characterize the relative degree of determinacy among legal contexts," but that "determinacy can be, and is, consciously architected by lawmakers."[24] Yet our experiment demonstrates that even relatively narrow and straightforward "rules" can be problematically indeterminate in practice.

CODING THE LAW EXPERIMENT

In our experiment, we challenged three groups of programmers—all students taking a third-year programming course as part of a Bachelor of Science program in Computer Science or Information Technology—to implement a subset of New York State traffic law[25] in code to determine algorithmically whether violations occurred (see Appendix A for a copy of the written assignment). These programmers were provided with real-world driving data extracted from

projects are exploring automation in substantive areas as varied as intellectual property, constitutional, criminal, and corporate law") (citations omitted).
[17] *Id.* at 5-6.
[18] *Id.* at 37.
[19] *Id.* at 88.
[20] *Id.*
[21] *Id.* at 89
[22] *Id.*
[23] *Id.*
[24] *Id.* at 88.
[25] Extracted from New York State Vehicle and Traffic Law Art. 30, §§ 1180, 1180-a, 1181 and Art. 20, §510 (2009).

the on-board computer of a commuter's automobile (a late model Toyota Prius) and a second dataset providing manually-constructed, but realistically-derived, speed limit information.[26] Given this data, the first group was asked to implement "the letter of the law" and issue traffic citations accordingly (the datasets provided are available online).[27] The second group was asked to implement "the intent of the law." The third, and final, group was given an additional, carefully-crafted, written specification from which to base their software implementation (see Appendix B). Both a computer scientist and an attorney reviewed this specification for accuracy. The specification was also verbally briefed to the third group to further clarify the requirements. The programmers had two weeks to complete the assignment.

1. Experimental Results

After the coding requirement was completed, we held three hour-long focus group sessions, one with each group of programmers, to draw out programmer assumptions and biases, especially those we had not anticipated. In addition, we carefully examined the code provided by each participant to identify additional insights not identified in the focus group sessions. We found that the even this simple appearing programming assignment possessed surprising degrees of freedom open to programmer interpretation. Figure 1 depicts the entire dataset by plotting speed against time. Note that the driver rarely exceeds the speed limit and then only by a moderate degree. From approximately 15 minutes to 40 minutes the driver had the cruise control set for 55 MPH. However, because the terrain was hilly and the cruise control on the Toyota Prius only controls throttle and not braking, the driver unwittingly exceeded the speed limit on several occasions. Despite the moderate and safe driving exhibited by the vehicle operator, programmer assumptions and the number of tickets issued varied significantly. The following sections examine key areas open to programmer interpretation as well as related issues surrounding implementing the law as code in our experiment.

[26] While we chose to include just speed limit data to scope our experiment, we suggest future research could include additional semantic information such as locations of school zones, construction zones, speed traps, and accidents. The blackbox dataset contained 12,425 time/speed samples, approximately 180 per minute, collected during a 66.621 minute drive.

[27] The driving dataset extracted from the black box is located at http://www.rumint.org/gregconti/publications/201302_driving_data.csv and the speed limit dataset may be found at www.rumint.org/gregconti/publications/201302_speedlimit.csv

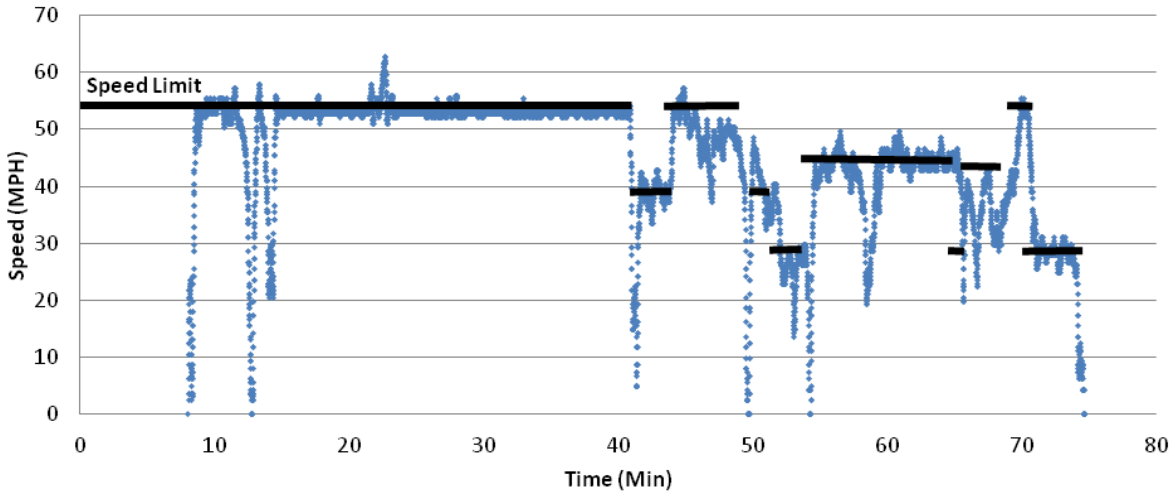

Figure 1: Depiction of the 12,425 samples in the dataset showing speed over a 66.6 minute drive, the horizontal bars represent the varying speed limits. Note the region from approximately 15 minutes to 40 minutes, when the driver had the cruise control engaged at 55 MPH. Despite the cruise control being continually engaged during this period, hilly terrain caused the vehicle to exceed the speed limit on several occasions.

Tolerance

In practice, speeding tickets are not typically issued for violations modestly exceeding the speed limit. Figure 2 illustrates this tolerance on a plot of a notional vehicle's speed over time. We depict the speed limit as a solid line and the enforced speed limit as a dotted line. The difference between these two values is the Tolerance. Assuming the enforced speed limit is greater than the actual speed limit, the tolerance is a positive value that can be measured in Miles Per Hour (MPH) or Kilometers Per Hour (KPH). As one examines the figure, note regions A and C, each of which highlights a region when the driver exceeded the enforced speed limit. Region B exceeds the actual speed limit, but not the enforced speed limit, and is hence not a formal violation. We further characterize the region between the end of one offense and the beginning of a second speeding offense as the Inter-Offense Time. We will use this nomenclature later in our analysis.

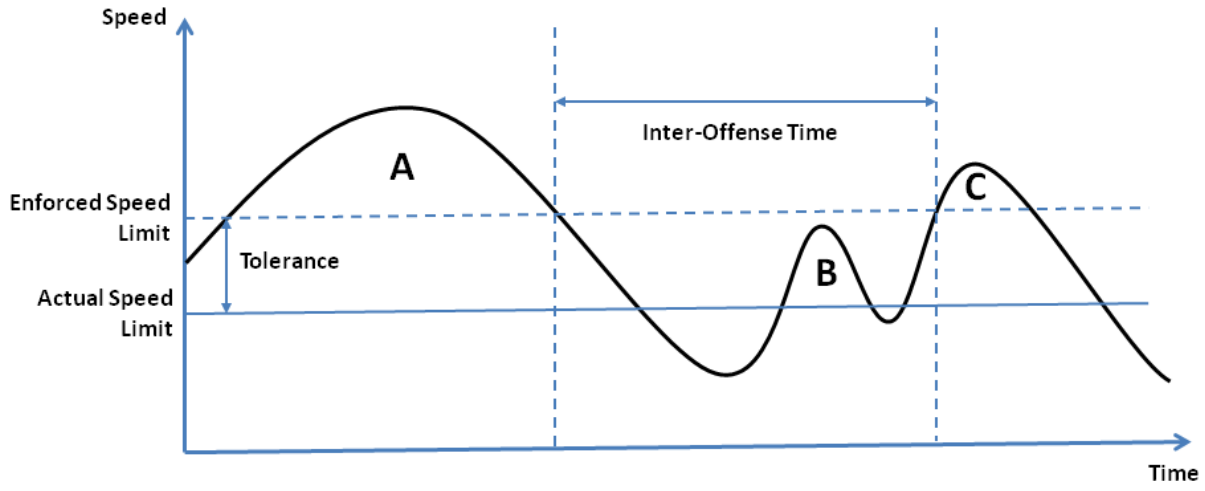

Figure 2: Depiction of a vehicle's speed over time. In this example the speed fluctuates above and below the actual speed limit. Due to norms in policing, we assume the enforced speed limit will be higher, by a value named Tolerance. Annotations A and C highlight regions when the driver exceeded the enforced speed limit. Region B is above the actual speed limit, but below the enforced speed limit and is not considered an offense in practice. The time between offenses A and C is annotated as the Inter-Offense time.

We saw significant differences in how the Letter of the Law and Intent of the Law groups chose to allow a tolerance for exceeding the speed limit. The entire Letter of the Law group (100%) issued tickets when the driving speed exceeded the speed limit by any amount. The entire Intent of the Law group tolerated minor infractions; their values ranged from a minimum of 3.5 MPH to a maximum of 20 MPH over the speed limit (Average = 8.14 MPH , Standard Deviation = 4.50 MPH). Figure 3 illustrates the distribution of these choices. As you examine the figure, note the two horizontal bars indicating the "tolerated" value given in the software specification. One might argue that the people who drafted the law had assumed tolerance in mind, rather than perfect enforcement. We documented assumed values of tolerance in the software specification, but left tolerance decisions up to the programmer in the other groups. The widely-varying interpretations by reasonable programmers demonstrate the human filter (or "bias") that goes into the drafting of the enforcement code. Once drafted, the code is unbiased in its execution, but bias is encoded into the system. This bias can vary widely unless the appropriate legislative or law enforcement body takes extra precautions, such as drafting a software specification and performing rigorous testing.

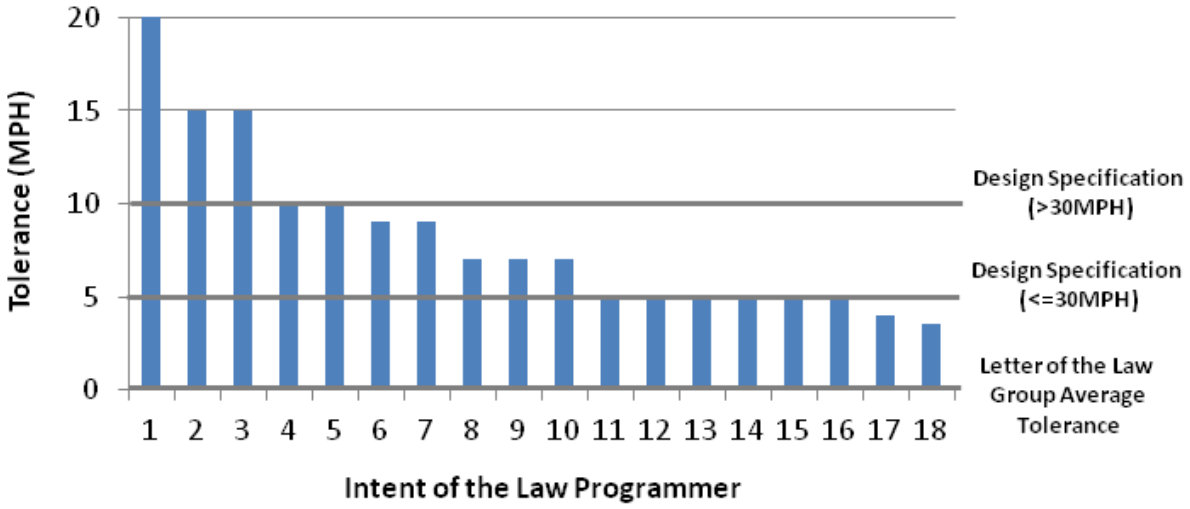

Figure 3: Distribution of tolerance values chosen by each Intent of the Law group programmer (n=18). The Letter of the Law group chose to not tolerate any speeding and issued tickets accordingly. The Specification group followed the specification exactly. See the three horizontal bars for depiction of these thresholds.

Tolerances for speeding are often not necessarily a single fixed value, but may vary based on the speed limit. Figure 4 illustrates the use of differing tolerances for differing speeds. For the Specification group, we chose to have two tiers, a 5 MPH tolerance for speed limits less than or equal to 30 MPH and 10 MPH tolerance for speed limits greater than 30 MPH. All members of the Specification group strictly followed these prescribed tolerances in their implementations. We note, however, that no members of the Letter of the Law or the Intent of the Law groups chose to implement tiered tolerances. Each member of the Intent of the Law Group chose their own value of a single tolerance shown in Figure 3. The Letter of the Law group did not allow any tolerances, tiered or otherwise. While we included two tier tolerances in the design specification, we suggest that additional tiers—or proportional tolerances such as a percentage of the speed limit—may be reasonable alternatives.

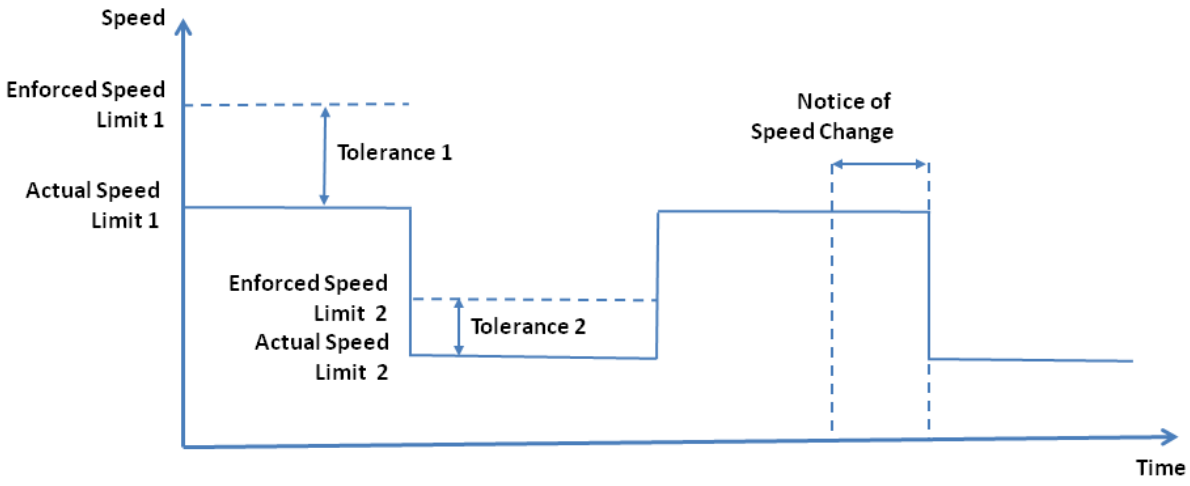

Figure 4: Illustration of variable tolerances based on speed. Note that Tolerance 1 is deliberately not equal to Tolerance 2. In this figure, we also annotate notice of a pending speed change.

Notice of Speed Changes

One aspect we did not consider in our specification (but emerged during the focus group discussion) was notification of speed zone changes. "Speed Zone Ahead" signs are typically employed when a driver needs advance notice to comply with an upcoming speed limit change. In the focus group discussions, we found the programmers assumed that the driver had sufficient advance notice of speed limit changes to comply with the law. They did not suggest an additional allowance that provides time to slow down *after* entering a slower speed zone, although we believe this allowance is worth consideration in the design of future, real-world systems.

Sensor and Timing Error

No sensor or timing source is perfect. In the focus group, we sought to identify whether the programmers assumed speed/time measurements came from an ideal source or if they allowed for a margin of error. If so, was this margin of error biased in favor of the driver (speed assumed lower) or biased toward law enforcement (speed assumed higher)? We found that 37.50% of the Letter of the Law group, 11.11% of the Intent of the Law group, and 100% of the Specification group accounted for error all biased in favor of the driver. All members of the Specification group employed the 3 MPH sensor error standard prescribed in the design specification.

We chose a value of 3 MPH in favor of the driver to take into account potential errors in measuring speed and time. However, error ranges are typically provided by the manufacturer as part of a given product's specifications. In addition, we typically think in terms of equally-weighted inaccuracy, such as a sensor that is accurate to within +/- 3 MPH. Not all such inaccuracy is balanced, however. It is entirely possible for inaccuracy to be +3/-2 MPH, particularly with a sensor that is not calibrated. Because sensor and timing error may degrade over time, accuracy can be validated through third-party testing and re-calibrated at, often appropriately certified, facilities. Upon careful examination of the dataset we noted that the black box sensor generating the data suffered from significant quantization error. Although the speed is expressed to six digits to the right of the decimal point, implying (to the casual observer at least) an accuracy of 0.000001 MPH, the speed is actually only accurate to about 0.6MPH. Careful observation of the data reveals that the speed varies in 0.62137 MPH increments: e.g. 16.15565 followed by 16.77702 in the next sample. Successive samples are all multiples of this 0.62137 increment (some are multiples of that increment if the car is accelerating or decelerating rapidly).

We note that none of the programmers commented on this flaw in the data. Figure 5 depicts two examples of sensor error placed at critical points on the graph: the first as the vehicle exceeds the enforced speed limit, and again as the vehicle falls below the enforced speed limit. We chose to make these error boxes of differing sizes because error need not be constant and may vary over time due to environmental conditions, such as temperature or rain, or due to loss of calibration. While we include two examples, error is in fact continuous and occurs during each sample measurement. As previously mentioned, the programmers who chose to include error, or were directed to include error via the design specification, all biased their results in the driver's favor. Such bias could be depicted on the graph by shifting the error boxes accordingly, with sizing based on time and speed error assumptions.

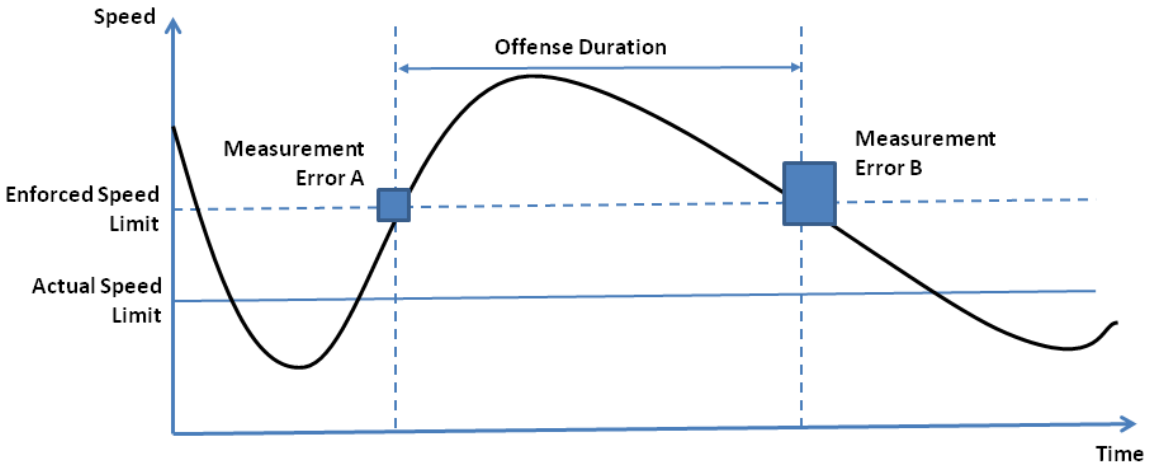

Figure 5: Neither sensors nor clocks provide faultless accuracy. This figure depicts regions of uncertainty due to measurement errors in both time and speed. Note that measurement errors may vary over time due to environmental changes, such as temperature. To illustrate this, we have made Measurement Error B larger than measurement Error A.

Clustering

Automation promises significantly different capabilities for law enforcement. In manual policing regimes, a policy officer might wait in a concealed location and capture a vehicle's instantaneous speed as it passes by. If this speed crosses the officer's own (or department-mandated) particular enforcement threshold, the police officer will stop the car, engage the driver, and potentially issue a ticket. An automated system, however, could maintain a continuous flow of samples based on driving behavior and thus issue tickets accordingly (see Figure 6). This level of resolution is not possible in manual law enforcement. In our experiment, the programmers were faced with the choice of how to treat many continuous samples all showing speeding behavior. Should each instance of speeding (e.g. a single sample) be treated as a separate offense, or should all consecutive speeding samples be treated as a single offense? Should the duration of time exceeding the speed limit be considered in the severity of the offense? In the Letter of the Law group only 12.5% of the programmers chose to treat the continuous “clusters” of offenses as a single offense and in the Intent of the Law group 33.3% chose to do so. The specification was silent on clustering, and instead chose a time-based limit (discussed in the next section), which is related to, but not identical to clustering. Despite the lack of guidance on clustering, no Specification group programmers chose to enact it. We noted that coding for clustering was modestly more complicated than the ticket-per-sample algorithm.

The group discussions highlighted this fact and some programmers stated that they chose the path of least resistance in creating their algorithm. While this aspect is related to the artificial nature of the experiment, we argue that contractors and government employees may make similar time- and effort-reducing decisions in the development of automated law enforcement systems to increase profits or to otherwise rapidly complete tasks. The discussions also highlighted other important issues related to clustering of samples and algorithmic decision-making, including whether the average speed above the limit or the maximum speed in the cluster should be used to determine the magnitude of the offense, as well as the role of offense duration. Should a shorter offense duration with the same maximum speed be given the same fine as a longer, potentially much longer, duration offense? Should the area between the speed curve and the enforced speed limit be used as the basis for the fine? Is there a minimum offense duration required before a citation is issued? These are open questions spurred by automated law enforcement, easily programmable but currently vague in legal application.

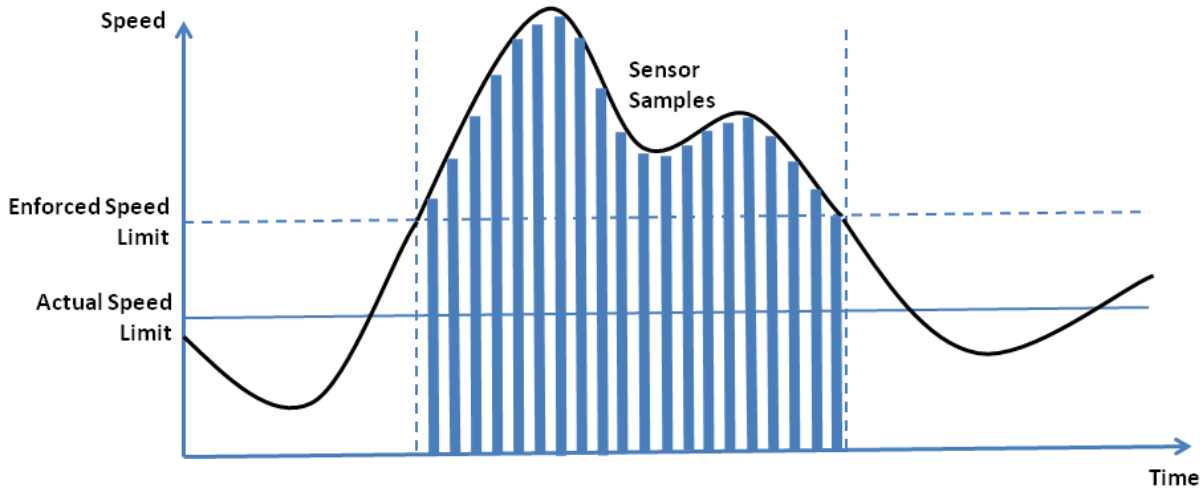

Figure 6: Measurements in the dataset are discontinuous and represent discrete samples, illustrated with blue bars, taken approximately three times per second. Some programmers chose to consider each sample as an opportunity to issue a ticket, but others treated each cluster as a single speeding infraction.

Inter-ticket Time

Whether a ticket is issued based on a single sample or a cluster of samples, a related aspect is the potential minimum time between tickets. Human-based policing does not have the

potential to issue multiple tickets per second as some programmers implemented in their code. For the design specification, we chose a 5 minute minimum inter-ticket time. 100% of the Specification group implemented this threshold. No one in the Intent of the Law group chose to implement inter-ticket time, and 25% of the Letter of the Law group did. Two chose 30 seconds, one chose 30 minutes, and a fourth chose to use a deceleration function to check that the driver was slowing down appropriately. Further discussion occurred surrounding the 30 minute value, with several suggesting that drivers might attempt to game the system by triggering a minor speeding ticket and then driving at excessive speed during the remainder of the 30 minute grace period.

Some programmers suggested that inter-ticket time was equivalent to "time to modify behavior." We disagree. "Time to modify" behavior assumes the driver was notified of the offense, which is not guaranteed. Human-based policing typically provides this feedback, by pulling over the driver and a police officer engagement. Some speed cameras and red light cameras might emit a photographic flash when taking a picture to provide instant feedback, but there could be confusion as to which was the offending vehicle. Some countries, such as Qatar and Saudi Arabia, mandate a loud buzzer in vehicles that triggers when a speed threshold is exceeded, providing instant and annoying feedback.[28] Other existing automated traffic monitoring systems provide notice via the postal service, weeks after the offense, effectively eliminating short-term behavior modification. We did not include an assumption of driver notice in the design specification. We suggest, however, that continuous monitoring is possible and hence allows for continuous feedback to the driver. Figure 7 depicts one such scenario. Here the driver is provided initial notice of a violation and is faced with a choice: ignore the notice and be issued a ticket at a later point (Path A), or decelerate to avoid the ticket (Path B). In order to ensure the "Path B" driver is complying, we depicted a region of acceptable required deceleration rates as a shaded region. To deviate outside this region could also result in a ticket, as the driver did not decelerate fast enough or chose to decelerate too quickly resulting in dangerous driving.

[28] Rabil. "Buzzer When Speed Exceeds 120kph." Hyudai-Forums, 10 May 2006. http://www.hyundai-forums.com/181-nf-2006-2010-sonata/72829-buzzer-when-speed-exceeds-120kph-2.html, last accessed 15 March 2013.

[28]

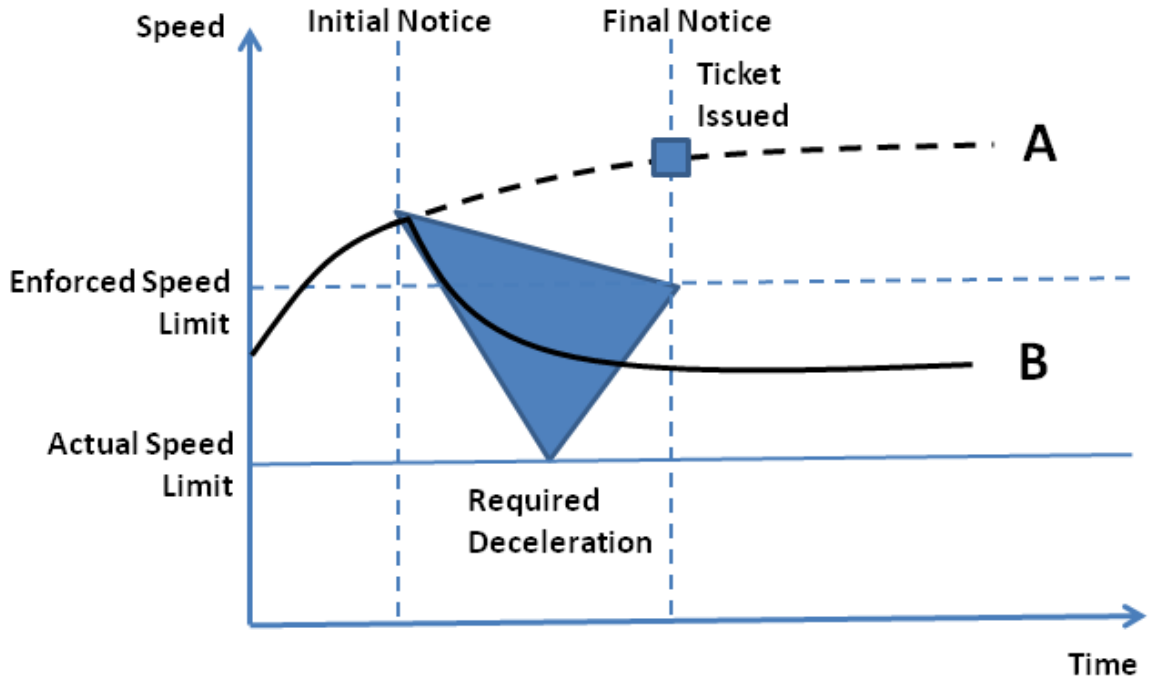

Figure 7: The focus group highlighted the potential need for system notification, either as an initial notice before a ticket was issued (Path B), so that a ticket might be avoided, or notice when a ticket is issued (Path A). Some systems might mandate deceleration within certain minimum and maximum values, as illustrated by the blue triangular region.

Context

Every programmer, across all groups, assumed "ideal" conditions, such as fair weather and open roads. The authors similarly had in mind ideal conditions when crafting the design specification, but the specification did not explicitly state this assumption. During the discussions, several important contextual issues were raised. Time of day, for example, is an easily measurable contextual attribute and common time-based driving scenarios, such as driving during rush hour or early morning Sunday after the mandatory closing time for bars, have long informed human-based law enforcement strategies. One programmer suggested allowing faster speeds late at night while roads were less busy, a technical possibility under automated law enforcement regimes; several others disagreed due to understandable safety concerns. Another suggested that times near the end of a month have historically played a role in the quantity of tickets issued, e.g. due to police ticket quotas or on holidays when police are paid overtime, but added that these inconsistencies should be surmountable in an automated law enforcement system. Location offers similar contextual insight, such as driving near a national border, in a

school zone, in an urban area known for drug sales, or through a zone known for traffic accidents. Weather and road conditions play a key role in driver decision-making, as does traffic volume and speed. For example, many states have minimum speed limits on major highways, but this point is often moot in heavy rain or stop-and-go traffic. Among many potential attributes, time, date, location, weather, road conditions, traffic volume, driver and passenger identities, even fuel levels in vehicles, each provide important contextual information, and are all amenable to automation to varying degrees. In combination, however, they provide significantly increased contextual resolution. On the other hand, what a programmer may define as "context" might be interpreted by another as "bias." That being said, properly integrating such contextual information into automated law enforcement systems, at all but the most basic level, remains largely an open problem.[29]

Scalability

Another aspect drawn out by the discussion was algorithm efficiency and system scalability. The programmers felt it far easier to code an efficient algorithm when evaluating a single vehicle, but another problem arises altogether when attempting to monitor six lanes of dense, high-speed traffic during rush hour. In the design specification, we did not include any guidance as to the performance constraints of the system, but these issues will likely arise in real-world systems. Advances in technology such as processor and networking speed improvements, will serve, nonetheless, to lessen scalability and performance concerns.

Subtle Coding Distinctions

When coding, developers make many subtle, low-level decisions that could ultimately impact an automated law enforcement system's performance, possibly even impacting determination of legal and illegal behavior. The focus group sessions raised several important areas. One notable example was data type choices. Data types define the logical constructs used by programs to hold information. For example, a programmer may decide to employ integer variables to hold numeric values without a decimal (e.g. -7, 0, 2, 16) or employ floating point variables to hold numeric values with a decimal (e.g. -7.09, 0.06, 2.0, 16.5). Some programming languages are "loosely typed" and attempt to infer from context what data type to use, alleviating the need of the programmer to make data type decisions in many situations. These decisions, whether manual or automated, may induce subtle errors in the code. In this experiment, the programmers used the programming language C#, which is a strongly typed language in which

[29] IBM's Watson supercomputer which in 2011 famously beat Jeopardy champion Ken Jennings, initially struggled to deal with context. These challenges were overcome by Watson architects in the narrow problem domain of Jeopardy-type questions, but not in a generalized sense. *See* John Markoff, *Computer Wins on 'Jeopardy!" Trivial, It's Not*, NEW YORK TIMES, (Feb. 16, 2011), http://www.nytimes.com/2011/02/17/science/17jeopardy-watson.html?pagewanted=all&_r=0. Also note that IBM's Deep Blue supercomputer, which beat chess champion Garry Kasparov in 1997, faced fewer contextual issues because of chess's well-defined rules. Rudy Chelminski, *This Time It's Personal*, WIRED (October, 2001), http://www.wired.com/wired/archive/9.10/chess.html.

the programmer must explicitly declare a data type for every variable.[30] Table 1 summarizes the choices made by programmers in each group, as well as the size and precision of data types in C#. Note that the design specification did not provide guidance on data types to be used, although it strongly implied data types capable of decimal place accuracy would be required.

Table 1: Data type decisions made by programmers and the size and precision of each data type. Note that subtle decisions such as these might affect an automated system's determination of legal and illegal behavior.

	Letter of the Law Group	*Intent of the Law Group*	*Specification Group*	*Data Type Size*	*Data Type Precision*
decimal	*68.75%*	*50.00%*	*33.33%*	*128 bits*	*28-29 digits*
double	*25.00%*	*44.44%*	*66.67%*	*64 bits*	*15-16 digits*
float	*6.25%*	*0.00%*	*0.00%*	*32 bits*	*7 digits*
int	*0.00%*	*5.56%*	*0.00%*	*32 bits*	*N/A*

We believe the programmers' decisions were reasonable for this programming task, with the exception of using integer data types, which discard decimal place accuracy in the samples. While seemingly innocuous, however, such decisions in general may impact calculations, particularly those that rely upon multiplication or exponentiation. A key tenet of chaos theory is that nonlinear systems are notoriously reliant on initial conditions, and even small deviations can cause dramatic changes in later state. Lorenz discovered this by noting simple rounding of parameters in a weather analysis program resulted in entirely different forecasts.[31]

Other related effects associated with these and similar programmer choices include unanticipated behavior if a variable exceeds the maximum allowed value. In some programming languages, adding one to the maximum value will result in wrapping around to the minimum allowed value. Math functions supported in programming languages—such as *round(),* which typically returns the whole number nearest the specified value; or conversions between data types, such as a float to an integer, which may truncate rather than round values after the

[30] *Data Types (C# Programmers Guide)*, MICROSOFT DEVELOPER'S NETWORK (MSDN), MICROSOFT CORPORATION, http://msdn.microsoft.com/enus/library/ms173104%28v=vs.80%29.aspx (last accessed 11 March 2013).
[31] Edward Lorenz, *Deterministic Nonperiodic Flow*, 20 (2) JOURNAL OF ATMOSPHERIC SCIENCES 130.

decimal place—may cause subtle errors. Other common errors may also occur, such as failure to initialize variables before first use or inadvertently being "off by one" in programming logic, an error that might inadvertently exclude analysis of the first or last element in the dataset. Errors of logic are likewise possible. For example, computers sequentially make a series of nested binary decisions, the order of which may affect the ultimate outcome. The litany of potential errors underscores the need for automated law enforcement systems to undergo rigorous testing and code reviews before deployment.

Number of Tickets

A strong indicator of the variance in programmer assumptions is manifest in the resultant number of tickets issued. The Letter of the Law group issued a draconian 498.33 (Standard Deviation = 453.42) tickets on average; the Intent of the Law group averaged 1.5 tickets (Standard Deviation = 5.68). The lack of a design specification was evident in both these groups, but application of the Intent of the Law group's personal assumptions resulted in a much tighter distribution and a much more reasonable number of tickets issued. The Specification group did not issue any tickets, which was in line with test code written by the authors to validate their results. Issuing of tickets was particularly dependent on programmer assumptions regarding clustering, tolerance, and inter-offense time. Figure 8 illustrates two examples (inter-offense time of 0 and 5 minutes) against tolerances ranging from 0 to 10 MPH. Note how the number of tickets drops off rapidly. At 8 MPH or higher tolerance, no tickets were issued.

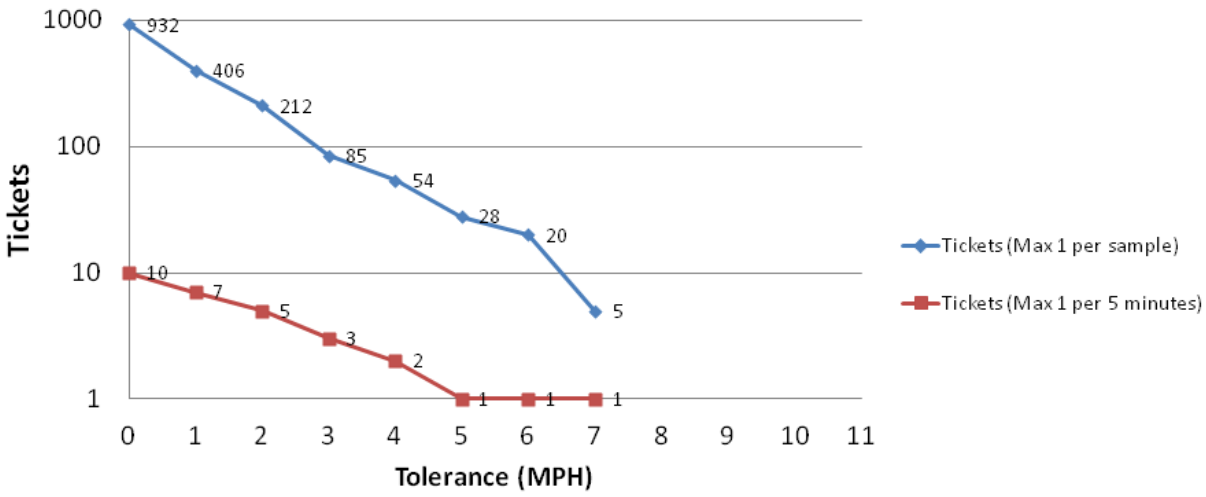

Figure 8: The number of tickets issued by an algorithm depends heavily on tolerance and clustering. By treating each sample as an individual punishable offense, there are a maximum of 932 instances in the dataset. By allowing 8 MPH tolerance, however, no tickets are issued. By treating each cluster of samples as a single offense, there are 10 instances in the dataset. This number also drops off rapidly with increased tolerance.

Concerns about Their Creations

We concluded each focus group session with the following question: "Would you want to drive on a highway that employed the system you built?" The results for the Letter of the Law and Specification groups were very negative, and even the majority of the Intent of the Law group was in disfavor. The low approval rating across all three groups is illuminating. Of the Letter of the Law group, 94% would not want to be subject to the system. The one supportive programmer qualified her vote, saying yes, but only if she could build in a back door that made her exempt from the system.[32] In the Intent of the Law group, 56% would not want to use the system, despite being responsible for many key design decisions. Finally, of the Specification group 95% did not want to be subject to the system. Recall, that the specification was written to follow current policing norms and provided generous allowances for tolerance and sensor error

[32] This offhand comment raises an important point. Those who code the system possess significant power. Some may seek to subvert the system they design. Such subversion is not without precedent. The presence of "Easter Eggs," unofficially embedded functions or sub-programs in a computer program, occur on a regular basis, perhaps the most famous is the flight simulator embedded in Microsoft Excel. See the Easter Egg Archive for an extensive listing, http://www.eeggs.com/ (last accessed 27 March 2013). Code reviews are one strategy for reducing, but likely not eliminating, back doors and Easter Eggs.

(8 MPH total for speeds less than or equal to 30 MPH and 13 MPH total for speeds greater than 30 MPH). All groups expressed concern regarding the continuous surveillance aspects of such a system. The focus group's general consensus was that engineers and programmers should be cautious about the systems they build lest they be used in detrimental ways, but many felt this caution may be overridden by desires for personal financial gain.

To conclude our results section, we provide Table 2, which summarizes the experiment's results.

Table 2: Summary of Experimental Results

	Letter of the Law Group (n=16)	*Intent of the Law Group (n=18)*	*Specification Group (n=18)*
Employed tolerance for speeds above speed limit	*0%*	*100%* *Average 8.14 MPH, Standard Deviation 4.50*	*100%* *5/10 MPH as stated in specification, Standard Deviation 0.00*
Employed varying tolerances depending on speed limit	*0%*	*0%*	*100%* *5/10 MPH as stated in specification*
Assumed sensor error	*37.50%* *All biased in favor of driver*	*11.11%* *All biased in favor of driver*	*100%* *3 MPH, biased in favor of driver, as stated in specification*
Treated each sample as a separate potential offense	*87.50%*	*66.67%*	*0%*
Enforced a minimum time between tickets	*25.00%*	*0%*	*100%* *5 minutes, as stated in specification*
Average number of tickets issued	*498.33 tickets for entire group, Standard Deviation 453.42* *661.33 tickets for subgroup treating each sample as a single offense, Standard Deviation 403.87* *11.25 tickets for subgroup treating multiple samples as a single offense, Standard Deviation 5.68*	*1.50 tickets* *Standard Deviation 2.73*	*No tickets issued*
Want to drive on highway using their algorithm	*6.25%*	*37.5%*	*5.55%*

2. Analysis

The results from our experiment highlighted the significant deviation in number and type of citations issued during the course of the commute, based on legal interpretations and assumptions made by programmers untrained in the law. These deviations were mitigated, but not eliminated, in the group provided with a legally-reviewed software design specification, thus providing insight into strategies to automate the law in the future. Perhaps, future laws intended to be automated could include human-readable and machine-processable components of verifiable equivalency. The results of our experiment and related analysis uncovered numerous important issues surrounding automated law enforcement and compliance.

What Are the Problem Areas for Automated Enforcement or Compliance?

Due to the proliferation of low-cost networked sensors and processors in the home, automobiles, workplace, and community, automated law enforcement is becoming increasingly feasible. Not all laws could or should be enforced in such a fashion, however. Various classes of law will continue to require some level of human interpretation,[33] and others may not be suitable for automation in the foreseeable future, if ever. Based upon issues encountered in the execution and results of the driving experiment, we outline below some considerations useful for critical analysis in considering whether the automated enforcement or compliance of a certain law is feasible. These considerations of feasibility include the kind and degree of culpability required by law, the degree of objectivity required to ascertain the wrongful conduct, the ability to identify the wrongdoer, and the accessibility of the information needed to determine wrongdoing.

Culpability

Strict liability offenses, that is, legal violations that do not require a finding of culpability, may be feasibly subject to automated enforcement or compliance. This includes many motor vehicle moving violations such as speeding or driving while intoxicated as well as other kinds of laws like curfews, noise ordinances, restraining order violations, among others. As discussed above, however, automating even the simplest of such laws is fraught with complications and a high likelihood of error, undesirable results, and significant unintended consequences. Laws that require some form of culpability or scienter, that is guilty knowledge, seem unfeasible to automated enforcement or compliance at this time, given the complexity of such requirements and limits on inferences of state of mind.

[33] A classic example of such human interpretation is U.S. Supreme Court Justice Potter Stewart's "I know it when I see it" test for obscenity. Jacobellis v. Ohio, 84 S.Ct. 1676 (1964).

Objectivity

An individual's wrongful conduct must be to some degree objectively ascertainable in order to be automated. For example, given the right sensors and access, an individual's speed, history of movement, and location are capable of being ascertained with little subjective inference, as are the time they were observed, whether they physically made contact with another person or object, and whether they said particular things. Increased subjectivity and contextual dependency of an inquiry reduces its validity. For example, while an individual's speed can be objectively determined, the same cannot be said for ascertaining whether an individual was driving prudently in inclement weather or unreasonably "interfering with the free and proper use of the public highway" or unreasonably endangering "users of the public highway."[34] Such subjective determinations require significantly more input from the observer or enforcing authority as well as enormous amounts of contextual information, which likely makes automation of such decisions currently unfeasible.

Identification

In order to be punished, those that violate the law must be correctly identified. Thus, any inquiry into automated enforcement should determine the extent to which a wrongdoer's identity can be proven. Traffic cameras, in addition to reading license plates, also typically take several photos of the driver as additional evidence. Various biometric identification mechanisms could also be deployed, such as facial, gait, iris, keystroke, retinal, and voice recognition. These technologies may be aided by large national biometric registry schemes and existing databases of law enforcement, driver's license, or national ID card identity information, and complemented by crowdsourced identification strategies when such records prove insufficient.[35] Yet even biometric identification protocols are fraught with problems.[36]

Issues of identification remain, however, in many currently automated enforcement regimes. For example, copyright owners seeking to automatically enforce their works are forced to rely upon IP addresses. This identifier is highly unreliable given due to multiple users of the same computer, spoofing, and other identity-masking strategies.[37]

---

[34]New York State Vehicle and Traffic Law Art. 30, §§ 1180, 1180-a, 1181 and Art. 20, §510 (2009); Art. 33, s 1212.
[35] One such crowdsourced identification strategy is the "Identify the Rioters" website, which seeks to "help identify the people behind the Vancouver riots" of 15 June 2011, see http://www.identifyrioters.com/.
[36] *See, e.g.,* Prabhakar, Salil, Sharath Pankanti, and Anil K. Jain. Biometric recognition: Security and privacy concerns, 1.2 SECURITY & PRIVACY, IEEE 33 (2003); Lai, Lifeng, Siu-Wai Ho, and H. Vincent Poor, *Privacy-security tradeoffs in biometric security systems,* COMMUNICATION, CONTROL, AND COMPUTING, 2008 46TH ANNUAL ALLERTON CONFERENCE ON. IEEE (2008).
[37] *See, e.g.,* Joshua J. McIntyre, *Balancing Expectations of Online Privacy: Why Internet Protocol (IP) Addresses Should Be Protected as Personally Identifiable Information*, 60 DEPAUL L. REV. 895 (2011).

Accessibility

The ability to observe wrongdoing is entirely dependent upon the placement and ability of sensors as well as the ability to obtain data from other inputs. Despite claims by some of living in a surveillance state due to the mass proliferation of sensors and data inputs, not all information required to ascertain wrongdoing is freely accessible. The ability to obtain information and the amount of friction or transactional cost related to doing so are thus important considerations for enforcement purposes. Information collected in public spaces by government entities, for example, typically requires less effort to collect by automated law enforcement systems than information possessed by private companies or individuals, which may require varying degrees of legal pressure, such as a court order or National Security Letter, to obtain. In some instances, legal or Constitutional safeguards may prevent such access.

Automated compliance with the law faces similar challenges. Those seeking to comply with the law must also ensure that the appropriate information is accessible. While most laws are publicly accessible, some robots might also require information from humans to comply with a law, such as age, location, sobriety, and even presence. Examples include prohibitions against access to pornography by minors, the illegality of online gambling in certain jurisdictions, and mental competence validation for those seeking to purchase firearms.

"Coding" the Law Requires Expertise and Potentially Lawmaking Authority

As demonstrated by our experiment, it is difficult to consistently automate enforcement and compliance with even simple laws that contain quantitative elements.[38] The programmers' assumptions and biases are embodied in the code they write. This problem can be ameliorated through well-constructed software design specifications, as seen in the much tighter grouping of outcomes from the Specification group, but this approach is not a panacea. Perfect specifications that anticipate all possible issues are exceedingly difficult to devise, but can be iteratively improved over time. Gaps in specifications invite programmer assumptions. Automated compliance systems must also understand the law at a deep level; we posit that automated law enforcement specifications should be transparently shared in many instances to aid compliance engineers and programmers. Many laws are ambiguous and frequently contradictory. To be properly instantiated in code, portions of the law must be redesigned or refined with potential automation in mind. We can learn from the work of the Internet Engineering Task Force (IETF), whose policies often include both human readable guidelines and unambiguous algorithms and grammars for implementing protocols and behaviors in

[38] For a seminal discussion of quantitative versus qualitative elements of the law and their applicability to automated vehicles, *see* Bryant Walker Smith, *Automated Cars are Probably Legal in the United States*, STANFORD CENTER FOR INTERNET AND SOCIETY (Nov. 1, 2012), http://cyberlaw.stanford.edu/publications/automated-vehicles-are-probably-legal-united-states. We note however, that while Smith suggests that quantitative guidance in law will be easier than qualitative to implement in automated systems, we tested this assertion in our experiment and found that sound algorithmic encoding of even quantitative law is nonetheless non-trivial.

code.[39] While imperfect, such IETF efforts have allowed myriad diverse computing systems to form the Internet we have today rather than an incompatible and incomprehensible Tower of Babel. We understand that rewriting or augmenting the law to reflect automated law enforcement and compliance is a massive and potentially disruptive undertaking, but to fail to do so invites *ad hoc* technological solutions that embed programmer, contractor, law enforcement, and myriad third-party entities' bias into code, effectively rewriting the law beyond judicial oversight.[40] Groups seeking to automate compliance should take extreme care to test and refine for optimal results. Due to the many assumptions that must be made with respect to the law, groups seeking to automate enforcement must ensure that they have appropriate lawmaking or interpretive authority to comply with due process requirements.[41]

Automated Law Compliance Programs Are Needed

We anticipate that certain robots—including automated vehicles and robotic law enforcement systems of the future—will need to be certified as compliant with certain laws. Robotic vehicles, for example, may require certification that they comply with the national, regional, and local traffic laws. Likewise, automated law enforcement systems need certification that they properly apply and enforce whatever set of laws they are designed to enforce. This creates the need not just for an appropriately balanced law enforcement program, but also an automated law compliance program for systems. These programs demand rigorous testing, transparency, and a routine update mechanism that will ensure systems remain current despite frequent changes in law and myriad jurisdictions, both geographic and virtual. We anticipate that developers, code, labs, and sensors will require a techno-legal testing regime in order to certify. We must also consider standards for robotic and system behavior when the system's intelligence determines that it cannot comply.[42]

[39] As an example, *see* RFC 2426 "vCard MIME Directory Profile" (1998) which contains a formal grammar that rigorously specifies the standard for vCard electronic business cards. The Platform for Privacy Preferences (P3P), http://www.w3.org/P3P/, a protocol for machine to machine communication of privacy preferences provides another useful example. P3P may provide a partial template for communication and negotiation between law enforcement and law compliance automated systems. For a legal critique of the strengths and weaknesses of P3P *see* William McGeveran, *Programmed Privacy Promises: P3P and Web Privacy Law*, 76 N.Y.U. L. REV. 1812 (2001).
[40] *See, e.g.,* Danielle Keats Citron, *Technological Due Process*, 85 WASH. U. L. REV. 1249 (2008).
[41] *Id.*
[42] Consider these two examples. From the end-user compliance perspective a headlight may burn out, a normally trivial failure but nonetheless illegal, and be detected by a self-driving car's on board compliance software. The car, if designed to obey the law and not merely pass along notice to its owner, may then park and sit idle awaiting repair. In an automated law enforcement system, a red-light camera might conduct a periodic self-test, and if it detects an error, reports the error condition to the system operator and then turns itself off until it is repaired and the error condition cleared. Both cases highlight the complex issues arising from only the simplest of circumstances involving automated law.

Even the Best Models will Prove Insufficient at Times

Despite the best intentions of designers, any model of the law and of the physical world is, by definition, a simplification. Environmental variables will fall outside the model and lead to error. Potholes develop, trees fall across roads, and streets become icy. Lack of context as well as absence of the traditional police officer's domain knowledge is likely, and in some cases inevitable, due to lack of appropriate sensor data or inability to process higher level cognitive functions in software. Expect court challenges. These external factors will cause compliance failures in law enforcement and compliance systems. A robotic car, for example, might slide through a stop sign due to snow and possibly record that it did stop because the wheels stopped turning. Or the car might drive 15 MPH on a freeway because a repair crew forgot to take down an RFID-enabled construction zone sign.[43] These realities force us to ask to what extent humans should mediate the complicated processes. Should compliance modules perform actions or just provide "suggestions" to humans? What is the role of due process?[44] Given the inevitability of errors,[45] should automated law enforcement systems and legal compliance systems incorporate grace areas or buffer zones tied to limitations in algorithm or sensor precision, or to take into account norms, both legal and societal? At the other end of the spectrum, the law doesn't deal in trivialities. Rigorous and painfully accurate automated law enforcement will generate excessive violations far beyond anything encountered with manual enforcement regimes of today, likely shockingly so. How should *de minimis* be applied to prevent a society where every driver loses his or her license in a matter of hours? Similarly, systems that rigorously comply with the law may create a correspondingly dystopian environment without human override, awareness of exigent circumstances, or common sense—a virtually-imposed, societal gridlock.

Expect Competition between Automated Enforcement and Compliance Systems

We anticipate that citizens will not wish to sit idly by while being subjected to increasingly automated law enforcement measures. Today we see radar detectors, radar absorbing paint, and license plate covers that become opaque when subject to a photographic flash of light, among myriad other countermeasures.[46] Increasingly automated law enforcement systems, compliance systems, and countermeasures compete with and generate unanticipated

---

[43] Automated systems can be designed to rely on existing infrastructure, such as today's traffic signals or speed limit signs, but new enabling technologies such as these notional RFID-enabled construction zone signs will be likely. *See* Reilly Brennan, *Q&A on the Future of Autonomous Driving*, REVS PROGRAM AT STANFORD (Jan. 17, 2013), http://revs.stanford.edu/blog/689.

[44] *See* Danielle Keats Citron, *Technological Due Process*, 85 WASH. U. L. REV. 1249 (2008).

[45] The popular press contains numerous examples of automated law enforcement system errors. *See, e.g.,* Scott Calvert, *City issued speed camera ticket to motionless car*, THE BALTIMORE SUN (Dec. 12, 2012), http://articles.baltimoresun.com/2012-12-12/news/bs-md-speed-camera-stopped-car-20121212_1_potential-citation-xerox-state-camera-ticket.

[46] Lisa Shay and Greg Conti, *Countermeasures: Proactive Self-Defense Against Ubiquitous Surveillance*, HOPE, New York City, July 2012.

second, third, and higher order effects.[47] We also anticipate a continued game of one-upmanship, as laws and regulations are modified to deter and combat countermeasure usage, while users respond concomitantly.[48]

There is an increasing need for law in algorithmic forms, and with it, myriad challenges. Red-light cameras now populate our intersections, and the Google Car is street-legal.[49] Moving violations are just some of the first offenses that can be complied with, processed, and adjudicated through automation. But the current implementation process is ad-hoc. The future portends widespread adoption of automated law enforcement regimes and necessity for certified legal compliance by robotics systems. This paper outlined challenges, open questions, and promising directions for future work to help both the legal and technologist communities move forward in a sensible and legal fashion.

We should expect law enforcement algorithms to be gamed or exploited, during design and coding, while in use, and after the fact in legal challenges. During design and coding, programmers or designers may seek to embed—either overtly or covertly—logic that biases the algorithm toward a desired goal. Examples include backdoors emplaced by programmers or law enforcement officials mandating exceptions for privileged classes such as law enforcement vehicles' being immune to detection by certain sensor systems. During use, we anticipate that those subject to automated law enforcement systems will learn the strengths and weaknesses of a system to operate outside its detection and recognition capabilities, when possible. This fact was highlighted in our focus group discussions by the programmers who suggested that a 30 minute grace period after issuing a speeding ticket provides opportunity for 29 minutes of unconstrained speeding. After use, we anticipate virtually every weakness of an automated law enforcement system will be dissected in court by defense attorneys seeking to prove their clients "not guilty." Knowledge gained during these before, during, and after analyses will be shared via social networks, specialized applications (such as mobile phone applications that share locations of speed traps), and via legal and human rights communities. Transparency of the algorithm, a common practice in the open source and cryptographic communities, will likely provide initial consternation to law enforcement officials as experts point out inevitable flaws, but iterative refinement will help address weaknesses and improve trust among those surveilled. Automated compliance systems could theoretically be designed with knowledge of the law alone, but would

---

[47] An excellent example of unanticipated effects of competing algorithms was the 2010 "Flash Crash" impacting U.S. stock markets. *See* U.S. Securities and Exchange Commission and the Commodity Futures Trading Commission (September 30, 2010). "Findings Regarding the Market Events of May 6, 2010".

[48] One example today is New York State Vehicle and Traffic Law Art. 30, § 1180 (g)(i) which makes radar detectors illegal to operate in vehicles exceeding fifty-five miles per hour in most circumstances, to deter radar detector usage by speeders.

[49] Bryant Walker Smith, *Automated Cars are Probably Legal in the United States*, STANFORD CENTER FOR INTERNET AND SOCIETY (Nov. 1, 2012), http://cyberlaw.stanford.edu/publications/automated-vehicles-are-probably-legal-united-states; Ryan Calo, *Nevada Governor Signs Driverless Car Bill Into Law*, STANFORD CENTER FOR INTERNET AND SOCIETY (June 22, 2011), . *See also* Gary Marchus, *Moral Machines*, THE NEW YORKER (Nov. 27, 2012).

be far more effective if compliance system designers were aware of the algorithmic details of how the law is enforced.

Beware Second and Third Order Effects

Automated enforcement will have unintended second order and third order effects. One example is influence on traffic flow. It is possible that once individuals learn where enforcement sensors such as traffic cameras are, they will choose driving routes around heavily-instrumented stretches of highway in the same way a long-haul trucker carrying a heavy load might avoid weigh stations.[50] If large numbers of drivers choose to avoid instrumented roads, what will be the impact on businesses, as well as safety and traffic congestion on secondary roadway systems?

New Paradigms in Punishment Are Now Possible

Automated law enforcement systems provide such high resolution into some classes of legal transgressions that new paradigms in punishment, tolerance, and forgiveness will likely be necessary. As our programmers illustrated, an experienced and responsible driver who drove safely and only modestly exceeded the speed limit would potentially be subject to dozens of speeding tickets per hour, if not more. Given that many jurisdictions suspend licenses after only a few tickets, left unchecked we may find that the asymptotically perfect nature of automated law enforcement systems may result in most drivers losing their licenses or becoming uninsurable. Elderly and young drivers may be particularly vulnerable. An interesting—and telling—future analysis on the datasets we provide may be calculation of drivers license points issued and measuring time to loss of one's drivers license. One potential solution, albeit culturally challenging, is the development of new means to address offenses, such as proportional tickets. Consider Figure 9. This figure includes an "enforced speed limit" that takes into account sensor error and a tolerance for exceeding the speed limit in accordance with social norms. The violation above this threshold is indicated in the gray shaded area. Perhaps rather than issue tickets based on each sample, future offenses could be measured and punished based on this area under the curve. Taking a holistic view of automated enforcement and punishment, some religious faith systems posit a God who keeps perfect tally of all transgressions while withholding judgment until death. It is fair to question the wisdom of continuous judgments and punishments throughout our existence on earth.

[50] *Truckers Avoid Weight Stations By 'Dodging the Scales*, KXAN-TV, Austin, Texas (Feb. 18, 2008). While this story appears to be no longer available online, numerous discussions it generated on truck driver forums remain available. For example, see http://www.thetruckersreport.com/truckingindustryforum/truckers-news/37068-truckers-avoid-weigh-stations-dodging-scales.html

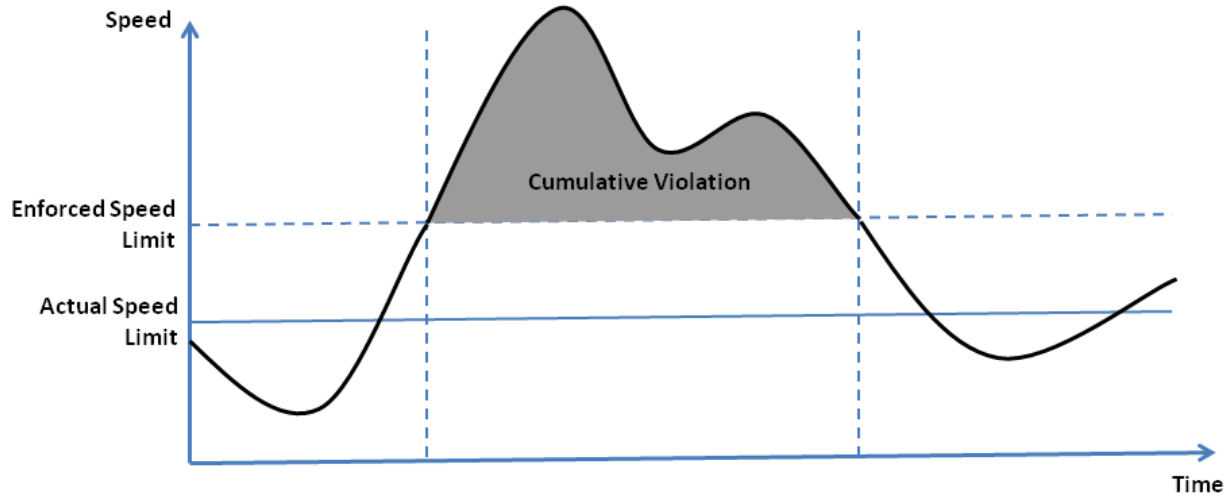

Figure 9: Automated law enforcement systems bring new capabilities. Rather than a police officer determining a specific instance of a violation more complete details about the behavior will be possible. This figure depicts a speeding driver's cumulative violation. New paradigms in punishment, that take into account the full extent of violations, may be possible under automated law enforcement regimes.

Blind Justice: The Social Cost of Robotic Law Enforcement and Compliance

The question arises, then: What is the societal cost of automated law enforcement, particularly when involving artificially-intelligent robotic systems unmediated by human judgment?[51] Our tradition of jurisprudence rests, in large part, on the indispensable notion of human observation and consideration of those attendant circumstances that might call for—or even mandate—mitigation, extenuation, or aggravation. When robots mediate in our stead—either on the side of law enforcement or the defendant, whether for reasons of frugality, impartiality, or convenience—an essential component of our judicial system is, in essence, stymied. Synecdochically embodied by the judge, the jury, the court functionary, etc., the human component provides that necessary element of sensibility and empathy for a system that always, unfortunately, carries with it the potential of rote application, a lady justice whose blindfold ensures not noble objectivity but compassionless indifference.

This conscious need for jurisprudence's flexibility is not only an integral component of our legal heritage, but it is also deeply interwoven into our cultural fabric. Consider Portia's eloquent plea to Shylock's humanity in William Shakespeare's *The Merchant of Venice*: "The quality of mercy is not strained; / It droppeth as the gentle rain from heaven / Upon the place

[51] *See, e.g.,* Michael Rich, *Limits on the Perfect Preventative State*, http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2231453.

beneath. It is twice blest; / It blesseth him that gives and him that takes: / 'Tis mightiest in the mightiest; it becomes / The throned monarch better than his crown." In a judicial system rigorously and efficiently enforced by robotics, either intelligent or not, it is unlikely that Portia's "gentle rain" will fall without human (or divine) intervention. The pound of flesh Shylock cruelly demands might be asked for merely in the guise of a speeding fine (or several, depending on the programmer's pre-coded level of tolerance), but the extreme potentialities envisioned in such a justice system, however remote, are frightening indeed. A living governor is needed.

An early antecedent to our judicial tradition—*judicium Dei*, or trial by ordeal—called for a defendant to be subjected to a dangerous test or series of dangerous tests, the results of which determined guilt or innocence of a crime—ordeals of fire, poison, water, etc. The possibility of human error was thus believed to be removed from the judicial process, which relied instead on divine judgment being manifested through the defendant's success, or lack thereof, during his arduous and usually gruesome ordeal. If the defendant was innocent, a sort of *deus ex machina* intervened and empowered the defendant-victim to endure the test and thus prove his guiltlessness due to divine intercession, protection, and thus validation. While robotic law enforcement is, obviously, a more humane means of meting out justice than submersion in a vat of boiling water, the absence of human interface—that ability to intuit deserved mitigation, extenuation, or even aggravation—carries with it, in its extreme, the disturbing potential of justice gone awry.

While automated enforcement certainly comes with a social cost, so too does automated compliance, which erodes, to a degree, some necessary components of a free society: individual agency and freewill. For instance, if a vehicle is equipped with an automatic speed governor that disallows a vehicle from exceeding the posted speed limit, then the driver loses the freedom of choice to obey or disobey the law. Of course, if our law-abiding driver operates within the legal limit, automated compliance will never activate, and the driver will appear to have the freedom of action. But it is essentially a deceptive freedom. The situation becomes more complicated when the automated compliance mechanisms are pre-programmed to be triggered by road variables: weather, sun glare, traffic, accidents, etc. The agency of the human driver is then replaced by that of an absent programmer and system designer, the authoritative gaze continually resting on the driver and her actions. While certainly this surveillance and control of our roadways and their safety has distinct social benefits, what might occur within the social body as our actions become increasingly articulated and then mediated by automated algorithms? What happens when our decision to act freely outside pre-programmed parameters is stymied by state-mandated or industry-installed machinery? What if a compelling need to violate the law arises due to an emergency or other unforeseen circumstance? Is there a concomitant degradation in respect for the law and authority if our agency to resist or challenge slowly fades? Indeed, does the law lose its mandate from the people if automation restricts our ability to act freely within and beyond the legal limits?

These are questions worth pondering as automated law enforcement and compliance are increasingly integrated into our quotidian activities. Certainly, versions of these enforcement and compliance systems are already functioning on our roadways and in our neighborhoods and workplaces, as alluded to above. That said, as technology rapidly progresses and cost-savings, efficiency, and accuracy are increasingly valued by our legislatures and law enforcement agencies, we see an immediate need to consider the societal cost of such systems while simultaneously assessing their feasibility for surrogating the police officer, judge, and jury.

CONCLUSIONS AND FUTURE WORK

We envision several important areas open for future research. We suggest exploration of re-writing several amenable laws to included machine-processable extensions, similar to those used by the Internet Engineering Task Force to describe network protocols, in both human and machine-processable forms. These modified laws could then be used in experiments that implement the law in code with the results—both the modified laws and resultant code—undergoing critical analysis by legal and technology experts.

Future work is also necessary to refine appropriate software engineering strategies for encoding the law in software. For example, we believe it necessary to formally identify potential degrees of freedom encountered by programmers and iteratively refine associated software design methodologies to explicitly limit these degrees of freedom and validate the resultant software artifacts as being legally compliant. In addition, our experiment dealt with three aspects: time, speed, and speed limit; however, we recommend exploring collection, aggregation, and disambiguation of other types of data to better inform higher level automated legal decision making. For example, we believe attempts at integration of time of day, date, location, weather, road condition, and identity data, as well as incorporation of higher precision semantic information, such as school zones, construction zones, and accidents, would provide valuable insights.

We also suggest creating experiments that pair automated enforcement schemes with automated compliance technology. For instance, in the context of the data we provided, what compliance technologies would be required, and how would they be algorithmically implemented to avoid violating speeding laws? We anticipate many interesting unanticipated aspects, insights, and issues would arise from such analysis. Finally, we recommend analysis and experimentation beyond the context of our driving scenarios; promising areas include curfew, rioting, and restraining order enforcement, among numerous others.

Automated law enforcement and automated compliance are in their early stages, but are rising in adoption, applicability, and importance. Our experiment illustrated that even apparently quantitative laws are difficult, but not necessarily impossible, to algorithmically encode in automated systems. However, care must be taken to carefully specify requirements to programmers, lest programmers make legal assumptions outside of judicial oversight. To do otherwise, risks simply shifting undesirable bias from humans to machines. The resultant

software systems must undergo rigorous testing and code reviews to validate appropriate behaviors. Ultimately, transparency of the underlying algorithms may prove necessary to ensure validity and acceptance of both automated law enforcement and compliance systems. Despite many technological and policy challenges, finding a sensible way ahead is necessary. Poorly conceived or executed automated law enforcement and compliance systems can extract a painful social cost and threaten the acceptance of the law itself by the very populace it seeks to protect.

***

BIOGRAPHIES

LISA A. SHAY is an Assistant Professor in the Department of Electrical Engineering and Computer Science at the US Military Academy at West Point. She holds a B.Sc. from the US Military Academy, an M.Sc. from Cambridge University, and a Ph.D. from Rensselaer Polytechnic Institute, all in Electrical Engineering. She is a Senior Member of the Institute of Electrical and Electronic Engineers.

WOODROW HARTZOG is an Assistant Professor at the Cumberland School of Law at Samford University. He is also an Affiliate Scholar at the Center for Internet and Society at Stanford Law School. He holds a Ph.D. in mass communication from the University of North Carolina at Chapel Hill, an LL.M. in intellectual property from the George Washington University Law School and a J.D. from Samford University.

JOHN NELSON is an Assistant Professor in the Department of English and Philosophy at the US Military Academy at West Point. He holds a B.S. from the US Military Academy, a M.A. from Oregon State University, and a Ph.D. in Comparative Literature from University of Washington.

GREGORY CONTI is an Associate Professor in the Department of Electrical Engineering and Computer Science at the US Military Academy at West Point. He holds a B.S. from the US Military Academy, an M.S. from Johns Hopkins University, and a Ph.D. from the Georgia Institute of Technology, all in Computer Science. He is a Senior Member of the Association for Computing Machinery.

APPENDIX A – CODING THE LAW ASSIGNMENT

Coding the Law

Homework Assignment

Instructions

You are designing an automated law enforcement system that will issue tickets to speeding drivers on a given stretch of "smart highway." This road contains sensors that determine the speed of all vehicles it carries and license plate readers that can uniquely identify all vehicles.

In this assignment you are to develop a C# program that determines the time and number of speeding ticket(s) to be issued to a driver. You will be given two files. The first file (driving_data.csv) contains a Comma Separated Value (CSV) series of time stamps and speeds, one sample per line (time in minutes, speed in MPH), taken from an automobile's on-board computer. The second file (speedlimit.csv) contains a matching series of timestamps and speed limits. The timestamp in this file indicates the time in minutes at which the vehicle encountered the given speed limit. You will need to load each file and construct appropriate logic to determine when violations occurred. Below are extracts from the laws you are enforcing:

http://www.safeny.ny.gov/spee-ndx.htm

http://www.safeny.ny.gov/spee-vt.htm#sec1180

http://www.safeny.ny.gov/spee-vt.htm#sec1181

Your approach to coding the law and determining violations will vary by class section.

Group A - Code what you believe to be the "letter of the law," e.g. a strict interpretation of the law.

Group B - Code what you believe to be the "intent of the law." e.g. an interpretation based on normal police law enforcement behaviors.

Group C - Code the law based on the design specification given to you separately.

Your program should output in a textbox the time that your software issued any ticket(s) and a total number of tickets. Your program should also save a CSV file (time stamp in minutes when violation occurred, speed travelling in MPH, speed limit in MPH) one row per ticket. The last line of the CSV should state the total number of tickets issued. For example

3.4, 62.0, 55.0

12.0, 65.0, 55.0

37.3, 112.0, 55.0

Total tickets = 3

There is no right answer for the number of ticket(s) to be issued to the driver, but your code should be based on the law and the per section guidance provided above. However, please provide a list of at least 10 assumptions you made during the design and coding process. These assumptions should be based on your interpretation of the law and how you implemented it in code.

Deliverables

1. An electronic copy of your complete C# Project.
2. An electronic copy of your program's CSV output.
3. An electronic copy of your ten or more assumptions in Word (.doc) format

APPENDIX B – DESIGN SPECIFICATION

Group C

Design Specification

Your program should issue traffic tickets based on the following logic. You are to assume that the sensor is properly calibrated and the results are accurate within +-3 MPH, the maximum capability of the sensor. Only a single sample of exceeding the speed limit is required to issue a ticket, but tickets will only be issued unless the driver has exceeded the speed limit by 13.0 MPH (for speed limits greater than 30.0 MPH) and 8.0 MPH (for speed limits of 30.0 MPH or less). This allowance is based on a 10.0 MPH and 5.0 MPH allowance, respectively, due to driver norms on the roadways where the data was collected and an assumption that the accuracy of the sensor falls 3.0 MPH against the driver, the worst case. At least 5.0 minutes must transpire before another ticket may be issued. Tickets may be issued immediately upon the start of the sensor data. As long as a ticket meets the above criteria there is no limit to the total number of tickets that may be given.

In addition, for your homework submission, I will assume you've followed the above guidance. In the written portion of the assignment, you should only list any _additional_ assumptions you've made (i.e. assumptions due to gaps or vagueness in the above guidance). This means there may be zero or more assumptions on your list. For grading, less than ten, even zero is fine, as long as that matches your actual code. That being said, it is helpful to us if you identify gaps in the specification logic.