On Analyzing Self-Driving Networks: A Systems Thinking Approach

Touseef Yaqoob
Information Technology University Lahore, Pakistan
msee17010@itu.edu.pk

Muhammad Usama
Information Technology University Lahore, Pakistan
muhammad.usama@itu.edu.pk

Junaid Qadir
Information Technology University Lahore, Pakistan
junaid.qadir@itu.edu.pk

Gareth Tyson
Queen Mary University of London, United Kingdom
g.tyson@qmul.ac.uk

ABSTRACT

Along with recent networking advances (such as software-defined networks, network functions virtualization, and programmable data planes), the networking field, in a bid to construct highly optimized self-driving and self-organizing networks, is increasingly embracing artificial intelligence and machine learning. It is worth remembering that the modern Internet that interconnects millions of networks is a 'complex adaptive social system', in which interventions not only cause effects but the effects have further knock-on consequences (not all of which are desirable or anticipated). We believe that self-driving networks will likely raise new unanticipated challenges (particularly in the human-facing domains of ethics, privacy, and security). In this paper, we propose the use of insights and tools from the field of "systems thinking"—a rich discipline developing for more than half a century, which encompasses more realistic models of complex social systems—and highlight their relevance for studying the long-term effects of network architectural interventions, particularly for self-driving networks. We show that these tools complement existing simulation and modeling tools and provide new insights and capabilities. To the best of our knowledge, this is the first study that has considered the relevance of formal systems thinking tools for the analysis of self-driving networks.

CCS CONCEPTS

• **Networks** → *Network design principles*; *Network dynamics*;

ACM Reference format:
Touseef Yaqoob, Muhammad Usama, Junaid Qadir, and Gareth Tyson. 2018. On Analyzing Self-Driving Networks: A Systems Thinking Approach. In *Proceedings of ACM SIGCOMM 2018 Afternoon Workshop on Self-Driving Networks, Budapest, Hungary, August 24, 2018 (SelfDN 2018),* 7 pages.
https://doi.org/10.1145/3229584.3229588

1 INTRODUCTION

The exponential growth in the number of connected devices and users in networks is placing significant stress on current human-in-the-loop network management architectures. There is now interest in equipping networks with autonomous run-time decision-making capability through the incorporation of artificial intelligence (AI), machine learning (ML), big data network analytics, and network telemetry to allow networks to configure, manage, and heal themselves. The idea that networks should learn to "drive themselves" is gaining traction, and there is a lot of interest in the networking community to develop *self-driving networks* [10]. The idea itself is not entirely new and reflects a recurring motif seen in various guises such as cognitive networking [48], self-organized networks [1], knowledge defined networks [33], and most recently, data-driven networking [19] and *self-driving networks* [10, 28].

The vision of self-driving networks is promising and finds much encouragement from recent advances in ML (such as deep learning) and networking (such as software-defined networks, programmable data planes, and edge computing). However, there are many challenges that remain. Most notably, modern networks, and their integration into the global Internet, yields a *complex adaptive social system* that encompasses the interaction of a vast diversity of autonomous devices, human users, applications, and service providers. Complex adaptive systems are characterized by their dynamic complexity and nonlinearity due to which, "*the act of playing the game has a way of changing the rules*" [15]. Any self-driving network must acknowledge and address this complexity. Hence, the real concern is not to only see the potential benefits of the approach to the optimizing entity itself, but to also critically understand potential downsides and unintended consequences on other subsystems. In this work, we seek to investigate the pros and cons of self-driving networks using *systems thinking* techniques.

1.1 What is Systems Thinking?

Although many different definitions of *systems thinking* have been proposed [3], all of them share an emphasis on interconnectedness and interdependency. They focus on understanding how various system entities influence each other and, in turn, themselves through feedback loops. The goal is to facilitate users in seeing the proverbial *"forest for the trees"* [45]. Systems thinking is characterized by three important features: *firstly*, the ability to think dynamically; *secondly*, to think causally through feedback loops; *thirdly*, to think more deeply about endogenous influences (where the system itself is the cause of the observed problems) [42]. Systems thinking is different from conventional thinking in many ways (see Table 1), but most prominently in modeling complex systems nonlinear, closed-looped, and multi-causal with delayed feedback rather than linear, open-looped, causal with immediate feedback.

Systems thinking has made a big impact in research on *complex adaptive systems*. In such systems, researchers have noted that

SelfDN 2018, August 24, 2018, Budapest, Hungary

ACM ISBN 978-1-4503-5914-6/18/08…$15.00
https://doi.org/10.1145/3229584.3229588

Table 1: Comparing Conventional vs. Systems Thinking (Details: [44][47])

	Conventional Thinking	*Systems Thinking*
Model of thinking	Linear, causal, open-looped, immediate feedback	Nonlinear, multi-causal, closed-looped with delayed feedback
Determining a problem's cause	Obvious and easy to trace	Indirect and non-obvious
Cause of problems	External to the system	Internal (System-as-a-cause thinking)
How to optimize?	By optimizing the parts	By optimizing relationships among the parts
Where to intervene?	Aggressive use of "obvious" solutions	Careful change applied at the "leverage points"
How to resolve problems?	Cure the symptoms	Fix the systemic causes

hardly anything is influenced linearly in just one direction and the presence of multiple intertwined nonlinear feedback loops have made social systems notorious for being counterintuitive [11]. In terms of self-driving networks, this implies that it is not sufficient to optimize only a protocol, an architecture or a network, without reasoning about how this will influence the other parts (technical as well as socio-cultural aspects) of the larger Internet system. Systems thinking the right tool for understanding complex adaptive social systems since it is considered axiom in systems thinking that every influence is both a cause and an effect [44, 47] and that interactions between (sub-)systems are modeled by *circular loops* rather than *directed arrows*.

We can define a system as, "*an interconnected set of elements that is coherently organized in a way that achieves something*" — a definition given by Donella Meadows, a highly influential system thinker and the lead author of the best-selling "Limits to Growth" [32]. The "something" in the definition may however be quite different from what the designer intended. If we find some stubborn problems with a system that refuse to go away despite efforts and best intentions (some authors call these *wicked* problems [6]), it is likely that these problems are *systemic* (i.e., the problems follow from how the system is built and from its architectural choices, goals, and constraints). Systems thinking opens us up to the fact that "wicked problems" may not be resolvable through further interventions and that using more advanced technology is not always good or neutral [49]. Thus, systems thinking may be defined as "*the ability to understand the systemic interconnections in such a way as to achieve a desired purpose*" [47]. For more information about systems thinking, we refer to various primers [27][26] and books [47] [6] on this topic.

1.2 Contributions of This Paper

In this paper, we aim to highlight that the Internet and self-driving networks should be envisioned as complex adaptive systems in which we should be wary of easy solutions and quick fixes. As pointed out by H. L. Mencken, there's always an easy solution to every problem that is neat, plausible, *but* wrong. In a similar vein, systems thinking research informs us that most well-intentioned solutions fail to sustainably solve their addressed problems and may actually create more problems than they solve. However, not all solutions are doomed in this manner—some "high-leverage" solutions exist that can provide sustainable long-run benefits with minimum effort and these can be uncovered by systems thinking. We propose the use of tools and insights from systems thinking in self-driving networks for managing the unintended consequences of policies and for devising high-leverage effective solutions. To the best of our knowledge, this is the first proposal to use systems thinking insights and tools for the study of self-driving networks and possibly also for the Internet.

2 WHY USE SYSTEMS THINKING FOR SELF-DRIVING NETWORKS?

2.1 Leveraging a rich set of theory and tools

Systems thinking has been successfully used as a management tool to study policy-making in domains such as healthcare, education, management [46] and looks promising for self-driving networks as well. The field of systems science is a highly-developed discipline with many schools of thought (including system dynamics, complexity theory, general systems theory, human system dynamics [34] [47] [3])[1]. We can leverage tools from a vast library of qualitative as well as quantitative tools (e.g., visualisations, domain specific languages) developed by the systems thinking community, which has been active since its genesis at MIT in the 1950s [44] [26].

As an example of a qualitative system thinking tool, consider *causal loop diagram* (CLD), which is an aid to visualize and easily communicate how different system entities connect to each other and influence each other possibly with a delay through reinforcing (positive) or balancing (negative) feedback loops. This could be used to capture various aspects of decision making within modern networks. For instance, Content Delivery Networks (CDNs) exert significant influence on the traffic generated within ISPs by dynamically setting the destinations of a large fraction of flows. This has long-term impacts on ISP decision making, as well as transient effects on relevant content producers, transit providers and exchange points. It has been hypothesized that integrating the control loops of these parties could have significant mutual benefits [13]. Self-driving networks offers a means to attain this, but a vital precursor would be formalizing the influences and dependencies between these stakeholders. CLDs offer a perfect tool.

In contrast to CLDs, the *stock and flow diagram* is a quantitative system thinking tool for understanding systemic structure. Stocks (or accumulators) are things that accumulate and can be measured (e.g., population; bits transferred; energy spent) while flows (or rates) represents things that change over time (e.g., transmission rate). Unlike CLDs, stock and flow diagrams can provide information about *rates* of change. Due to the lack of space, we limit our discussion to these two tools only but highlight that the field has a number of other tools, details of which can be seen in [26] [47].

2.2 Support for rigorous big picture thinking

Systems thinking also affords us the ability to see the big picture by expanding our *time* and *thought* horizons. Using system thinking tools, we can take better policy decisions regarding self-driving networks and avoid an exclusive reliance on implicit mental models, which are ill-suited for this task since they are simplistic (since they inadvertently substitute a higher-order nonlinear system for a

[1] In this paper, we consider systems thinking to be synonymous with system dynamics [43] and to encompass it [6], although not everyone agrees [12].

linear causal one); *narrow* (i.e., not broad enough to see the big picture); and *myopic* (since they tend to discount the future and focus predominantly on the short-term) [11]. Systems thinking can also be used to better understand the connections between the various subsystems. In particular, it helps us identify non-obvious connections between effects and causes; and also find missing connections, which if they had existed, would have improved the system performance of our self-driving networks.

2.3 Finding high-leverage interventions

In systems thinking, systems respond to interventions according to the *principle of leverage* [47]. Previous research in system dynamics has shown that most intuitively obvious policy interventions in complex social systems are *low-leverage* (i.e., they do not produce significant long-run change and will likely also create other problems) and only a few policy interventions are *high-leverage* (i.e., capable of producing substantial long-run change). System dynamics research has consistently highlighted the counterintuitive nature of complex social systems in that the high-leverage points are not where most people expect, and if even these points are identified, they are prone to be altered in the wrong direction by people [11]. In an influential essay on leverage [31], Donella Meadows stated that interventions that rely only on parameter optimization are typically low leverage, and higher leverage can be attained through deeper interventions that for example optimize information flows (e.g., by minimizing information sharing delays) or change the system rules (i.e., the incentives and the constraints); the most powerful way to change a system, Meadows notes is to change the system goals and paradigm, out of which its goals, rules, and culture emerge. Although these ideas are abstract, we can use insights about leverage points to unearth the few sensitive influence points in self-driving networks, thereby avoiding some of the problems that have plagued traditional networks.

2.4 Facilitating "system-as-cause" thinking

In systems thinking, it is considered an axiom that every influence is both a cause and an effect—i.e., it is possible that when A causes B, B also causes A through a feedback loop—in such doubly looped system, the systems are said to cause their own behavior *endogenously*. We can use the systems thinking concept of *system-as-a-cause* to explain how the perennial Internet nuisances (such as spam and lack of privacy, security and QoS) are not isolated problems but, as noted by Keshav [25] follow endogenously as the byproducts of the Internet's design preferences. This work points out that paradoxically the Internet's architectural elements most responsible for its success are also responsible for its most vexing problems. It is clear that if we want to fix these ancillary problems, this cannot be achieved superficially without changing the systemic causes. We can use this *system-as-cause* understanding in self-driving networks to ensure that the purposes achieved by the self-driving network are congruous to our stated goals.

2.5 Management of unintended consequences

Unintended consequences are the staple of complex social systems, which follow unexpectedly from the nonlinear interactions between subsystems [11] and our propensity to intervene in systems with our "solutions". Unfortunately, our problem-solving instinct also creates a number of followup problems and networking systems (including future self-driving networks) are not immune to this tendency [40]. Systems thinking can help us anticipate and avoid the negative consequences of well-intentioned solutions. This can be done both *prospectively* by anticipating unintended consequences during strategic planning or *retrospectively* by understanding more deeply the non-obvious causes of existing chronic complex social problems.

3 SYSTEM (MISBEHAVIOR) ARCHETYPES

System dynamics literature is rife with examples of fixes gone wrong—in which well-intentioned common-sense interventions to mitigate a particular problem has gone on to aggravate it (not to mention the creation of other problems) [44]. Peter Senge, the author of the best-selling systems thinking book [44], devised a list of system laws generalized from manifestations of commonly observed system behavior in diverse settings and many of these laws are somber reminders of how systems can misbehave, and how solutions can themselves create new problems.

Some of Senge's laws most pertinent to our work are: 1) *today's problems come from yesterday's "solutions"*; (2) *behavior grows better before it grows worse* (i.e., benefits of quick-fix interventions accrue in the short-time, only to neutralize and worsen off in the long-run); (3) *the easy way out usually leads back in*; (4) *the cure can be worse than the disease* (i.e., short-term improvements can lead to long-term dependencies); (5) *cause and effect are not closely related in time and space*; and (6) *small changes can produce big results—but the areas of highest leverage are often the least obvious*[2].

We term these generalizable pitfalls as "archetypes". This section will detail some of the broadly applicable system archetypes and discuss how they might apply to networking in general and to self-driving networks in particular. These archetypes are easily understood, and once internalized, can help designers and stakeholders in identifying the rut they are in and to identify recognizable paths (the leverage points) that can adopted for a resolution. These system failure archetypes are listed in Table 2, along with some networking examples (due to the shortage of space, these are not always elaborated upon in text).

3.1 Fixes that Backfire

This system archetype is associated with the concept of unintended consequences. Fixes that backfire are characterized by the use of a quick fix to reduce a problem symptom that works in the short run but at the cost of long-term consequences (which people often fail to see due to long system delays). This is a common pitfall in networks with some networking examples including (1) increasing queue buffers to decrease packet loss but instead causing bufferbloat [14], and (2) introducing additional links to an existing system only to see overall performance drop (Braess' paradox) [24].

3.2 Shifting the Burden

This archetype is associated with the concept of unintended dependence. This arises from dependence on a quick fix, which is resorted to when the more fundamental solution is too expensive or too difficult to be implemented. This archetype differs from "fixes that backfire" since the fundamental solution may not be apparent or applicable in the latter. With the "shifting the burden"

[2] Due to the lack of space, we omit explanations of these laws and refer interested readers to [44].

Table 2: System archetypes identified in system dynamics [44] with networking examples

No.	Archetype Name	Description	Networking Examples
1	Fixes That Backfire	A quick solution with unexpected long-term consequences	[23] [25] [14] [24]; IP NAT
2	Limits to Growth	Improvement accelerates and then suddenly stalls	IPv4; [41] [18]
3	Success to the Successful	Things get better for “winners” and worse for “losers”	[4] [16]
4	Shifting the Burden	Systems unconsciously favor short-term, addictive solutions	[23]
5	Tragedy of the Commons	Shared unmanaged resource collapses due to overconsumption	[8] [17]
6	Escalation	Different parties take actions to counter a perceived threat	[35] [7]
7	Eroding Goals	Short-term solutions lead to the deterioration of long-term goals	[7]

archetype, the quick fix produces temporary relief by treating the symptoms, which tends to reduce the motivation to implement the more fundamental solution. The best example is seen in network capacity planning, where operators would rather over-dimension the network than implement more complex long-term solutions.

3.3 Limits to Growth

This archetype describes the concept of unanticipated constraints, based on the insight that no physical system can sustain growth indefinitely. Any engine of growth, however successful, will inevitably be constrained by internal and external bottlenecks and constraints, e.g., Meadows [32] showed that we cannot sustainably support perpetual growth in a finite world. This is a long-standing worry in communications networks. For example, researchers are now exploring how a permanent energy crisis scenarios may fundamentally limit our ability to maintain the current-day Internet architecture and what our response should be in such an eventuality [41] [39].

3.4 Success to the Successful

This archetype is associated with the concept of the winner takes it all. It refers to the common tendency in social systems for the privileged to accumulate more of the benefits than the underprivileged. This archetype commonly occurs in system dynamics and helps to make differences in privileges more pronounced over time. For the purpose of self-driving networks, this archetype has implications for policy making for network neutrality and for ensuring fair resource allocation. For example, as major corporations (e.g., Google, Facebook) control increasing portions of the Internet’s content and infrastructure, it seems likely that long-term competition might be crowded out. This would potentially lead to an Internet designed around the few, rather than the many.

3.5 Eroding Goals

This archetype, also called *“Drifting goals”*, is another easily recognized system archetype. It is a special case of “shifting the burden”, where the preferred quick fix is to repeatedly lower the system goals. This continuous adjustment then turns out to be fatal for the system, as it fails to fulfill its original design purpose. In networks, this sometimes happens in emerging markets where initial deployment expectations are curtailed by economic considerations [9].

3.6 Escalation

This system archetype describes the story of unintended proliferation in a sort of an arms race in which the harder you push, the harder the adversary pushes back. Many examples can be taken from network security, where the attempts by applications to encrypt their traffic is responded to by adversaries upskilling their monitoring technologies (e.g., fingerprinting encrypted webpages).

3.7 Tragedy of the Commons

Perhaps the most famous archetype, this refers to the concept of a depleting shared resource that all parties are interested in exploiting but none feel responsible for maintaining. For networking, this is applicable for unlicensed use of natural shared limited resources such as radio spectrum—e.g., the problem of interference in unlicensed wireless commons [30]. It was also commonly reported in peer-to-peer file sharing applications, where users would avoid sharing upload capacity [22].

4 APPLYING SYSTEMS THINKING IN SELF-DRIVING NETWORKS

In this section, we begin to explore how systems thinking may be applied to self-driving networks. In particular, we propose a system diagram (or *systemigram*) of self-driving networks in Section 4.1 and discuss various considerations for improving system structure in Section 4.2.

4.1 Systemigram of Self-driving Networks

To illustrate how system thinking concepts may be applied in the context of self-driving networks, we use a systemigram (a portmanteau combining the two words *system* and *diagram*) shown in Figure 1. The ultimate goal of systems thinking is to improve the understanding of systems, predicting their behaviors, and devising modifications in order to produce desired effects.

The journey towards this goal starts with understanding the system structure *(Oval 1* in Figure 1) which includes recognizing the interconnects between system components, identifying the feedback between the various entities, and identifying all the stakeholders. This understanding of the structure will help us understand the dynamic behaviors of the underlying system *(Oval 2)*.

The complex system dynamics can be modeled *(Oval 3)* using tools such as stock and flow diagrams and behavior over time graphs [26]. The system’s non-linear behavior can be also understood by modeling the cause-effect relationship among the variables and fixed entities in the network system using tools such as causal loop diagrams. System dynamics tool allow us to simulate system models, which can be used to test out the efficiency of various policies or interventions *(Oval 4)*. It is worth noting that self-driving networks do not capture a single system but can be thought of as multiple interacting systems, in which one system may be a subsystem of a larger system.

Systems thinking is all about expanding horizons and seeing the big picture of how the system interacts with other systems, which mutually influence each other. Therefore the next stage is emphasizing an understanding of the system at various scales *(Oval 5)*. Following these steps will improve our capability of identifying systems, predicting their behaviors, identifying relevant system

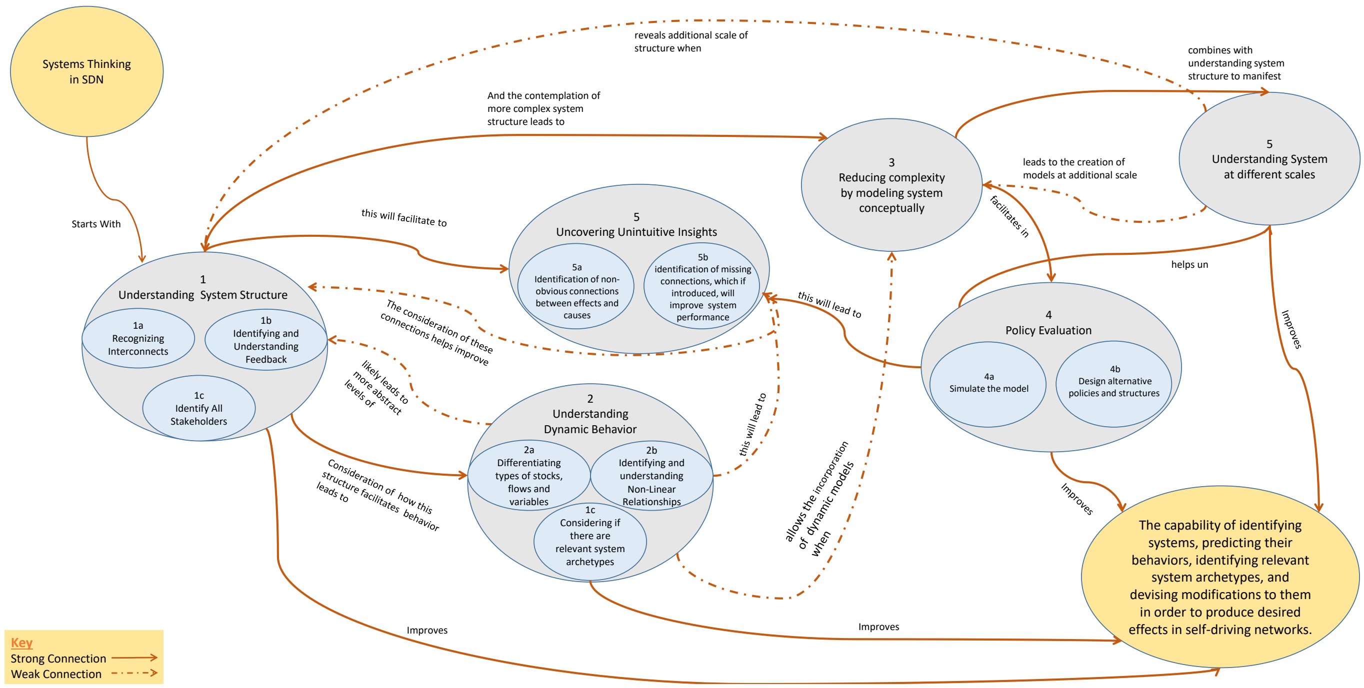

Figure 1: Systemigram of Self-driving Networks

archetypes, and devising modifications to them in order to produce desired effects in self-driving networks.

4.2 Improving System Structure

This section will briefly discuss certain key things that should be understood when designing self-driving networks. Again, we take inspiration from systems thinking in identifying these considerations.

4.2.1 Tussles, Conflicts, and Dilemmas. First, it must be kept in mind that different stakeholders on the Internet ecosystem have different, often conflicting, interests, which when independently pursued create "tussles" of various types. For example, some people wish for privacy on the Internet, others prefer accountability and the ability to identify behaviors. Some protocols aim to implement a functionality in an end-to-end manner; others may prefer an in-network mechanism. The functionality implemented at various layers may be neutralized or may even conflict. When designing a self-driving network, we must be clear on which conflicts are being managed within each design decision.

Thus, there is "*not a single happy family of people*" on the Internet with aligned goals [7]. Apart from tussles and conflicts, Internet protocols and applications also often face dilemmas in which the goals of the subsystem and the overall system conflict. One of the major insights of systems thinking is that the best way to optimize a system is not to independently optimize each subsystem but to optimize the relationships among the parts (which often is the bottleneck). An important implication for self-driving networks is that we cannot be everything to everyone—it becomes important to clearly articulate our goals while keeping in view that different subsystems do not have homogeneous interests or points-of-view. We can also use systems thinking tools to anticipate the non-obvious interactions between the subsystems and use insights therefrom to minimize tussles and bottlenecks.

4.2.2 On Architecting Goals for Networks. Interventions that aim to optimize parameters are nowhere as powerful as interventions that aim at changing the system's goals and paradigms [31]. To ensure better performance, we need clearer articulation of what the goals of our self-driving networks are. We argue that these goals should be enshrined in formal declarative languages, that are common across diverse stakeholders. Through these, it becomes possible for systems engineers to reason over their design decisions. For example, it makes it possible to identify conflicting goals as well as key trade-offs between them.

4.2.3 Focusing on System Bottlenecks. A system's performance is never the sum of the performance of its parts, but the product of their interactions. To improve system performance, bottlenecks should be identified and efforts should be invested in alleviating these bottlenecks rather than on optimizing subsystems separately. For example, it is well known that control loops within content delivery networks and Internet service providers can conflict (e.g., redirection strategies within a content delivery network may negatively impact the load balancing strategies of the network). Optimizing these two subsystems separately is far less beneficial than improving their interactions [38]. In addition to identifying the problematic connections (i.e., bottlenecks), self-driving networks could also leverage systems thinking to determine new types of interactions that could mitigate bottlenecks through more efficient information sharing [13]. Doing so in an integrated way might address problems faced when deploying past collaborative mechanisms, e.g., ALTO [5].

Unfortunately, these control loops can suffer from significant timing delays. In most networks, sharing of information is limited to standard protocol exchange (e.g., BGP). Less conventional data sharing occurs within a period of days (typically manually). This means that relevant information may not be available to the decision maker when required. This is particularly problematic in self-driving, which may rely on second-by-second updates. To facilitate the required timely sharing of information, new architectures and strategies (such as split control architectures) are needed [19].

5 CHALLENGES IN DEVISING SELF-DRIVING NETWORKS

The previous section has identified key opportunities and approaches to designing self-driving networks. Finally, we now look at key open questions and challenges that remain as ripe areas of research.

5.1 Finding the Right Functional Split

Despite the moniker of self-driving networks, humans will not be removed completely from the management of networks. There will inevitably be a functional split between humans and computers for network management. It is true that algorithms can prevent many trivial manual mistakes, but it is worth keeping in mind that algorithms are also not impervious to blunders [29] (since algorithms do not have the common sense and can only learn from the given instructions or data). With it being well known that human intuition is sometimes marvelous and sometimes flawed [21], an important (and not entirely technical) exercise is to map the boundary conditions for the management of self-driving networks where we can safely relegate matters and operations to algorithms and where we will like to have human oversight (e.g., in crafting policies related to matters pertaining to ethics and human subjects). There will likely be many configurations of self-driving networks and more debate is needed on the right functional split—especially to avoid reliability, security, and ethics related problems.

5.2 Ethical Challenges

Giving away the agency of decision-making to algorithms in self-driving networks opens up a plethora of ethical challenges. Despite the many successes of machine learning (ML), experts have pointed out that many modern ML techniques work like a black-box and may make predictions without really knowing why. The harmful effects of opaque unregulated ML-based algorithms described by O'Neil in [37] represent a significant concern for self-driving networks. In [10], an example of an ML-based spam filter was proposed using features such as the autonomous system number of the sender. Although very useful, one should reason ahead about the potential of "false positives" and take steps to ensure that we do not inadvertently create "weapons of math destruction" or strengthen existing stereotypes [37] [50]. We believe that systems thinking can help us perform higher order thinking and determine unintended consequences of relying on opaque ML algorithms and potentially biased datasets.

The question of *agency*—i.e., "*who will take the ethical decision?*—also looms large for self-driving networks. It is not clear if network operators and managers should make ethical decisions on behalf of the uses and if so then how. These ethical questions may not have an objectively straightforward resolution and present dilemmas (e.g., self-driving network version of trolley problems [36] may arise in which the interest of many might be vying with the actions of a limited few and one has to decide how this conflict is to be addressed). An example would be where a self-configured network chooses to block certain IP ranges to curtail a perceived DoS attack, whilst also impacting regular users.

The ethical decisions adopted may also have strong social and economic implications, as the policy may be beneficial for some stakeholders but not for others. Furthermore, changes in incentives may trigger changes in the services and products the clients will use. We believe that systems thinking can allow us to rigorously study these ripple effects in self-driving networks. Ethical concerns related to networking research are now being documented and guiding principles articulated [2, 20], but specific ethical concerns around self-driving networks require more deliberations.

5.3 Security Challenges

As remarked tellingly by Russell Ackoff, "*no problem stays solved in a dynamic environment.*" Since algorithms are trained using historical datasets, self-driving networks are always vulnerable to future evolved adversarial attack. We argue that we should use systems thinking tools to anticipate the various kinds of crippling attacks that adversarial attackers can launch on self-driving networks. Relying on algorithmic measures also opens up an opportunity for malicious applications/users to game the system. Since self-driving networks and adversaries, both will be using ML techniques this will produce an arms race in the network, which is related to the *escalation* system archetype which can be rigorously modeled using systems dynamic tools to preemptively discover the unintended consequences of this adversarial situation.

6 CONCLUSIONS

Our technological interventions in the Internet have wide-ranging implications since Internet technologies are deeply embedded in a larger social, political and cultural context. With the rise of interest in self-driving networks, which will become part of the larger Internet, there is a need to rigorously look at how these technologies will affect—positively as well as negatively—all the stakeholders. In order to devise appropriate policies for future self-driving networks, it is essential that we not only use traditional machine learning (ML) and analytic tools but also complement these with systems thinking tools to study the dynamics of interaction within self-driving networks and between it and other interacting systems. We believe that system thinking complements traditional methods (e.g., mathematical/statistical/ML models as well as discrete-event simulators) to bring unique insights not attainable through traditional methodologies. Our work applies for the first time powerful insights from systems thinking and demonstrates their relevance for studying the broad implications of self-driving networks. Although principally applicable to all networks, systems thinking tools are especially relevant for self-driving networks that will rely on ML-based data-driven algorithms to autonomously drive networks—which can suffer from problems such as bias, noise, and unintended consequences—to help troubleshoot chronic problems and to ensure that no significant unintended consequences are ignored during design.

Acknowledgments. We would like to thank the anonymous reviewers, and our paper's shepherd Jennifer Rexford, for their insightful feedback on the paper.

REFERENCES

[1] Osianoh Glenn Aliu, Ali Imran, Muhammad Ali Imran, and Barry Evans. 2013. A survey of self organisation in future cellular networks. *IEEE Communications Surveys & Tutorials* 15, 1 (2013), 336–361.
[2] Mark Alllman and Vern Paxson. 2007. Issues and etiquette concerning use of shared measurement data. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 135–140.
[3] Ross D Arnold and Jon P Wade. 2015. A definition of systems thinking: A systems approach. *Procedia Computer Science* 44 (2015), 669–678.
[4] Yochai Benkler. 2006. *The wealth of networks: How social production transforms markets and freedom*. Yale University Press.
[5] Eric W Burger and Jan Seedorf. 2009. Application-layer traffic optimization (ALTO) problem statement. (2009).
[6] Derek Cabrera and Laura Cabrera. 2015. *Systems thinking made simple: New hope for solving wicked problems*. Odyssean Press.
[7] David D Clark, John Wroclawski, Karen R Sollins, and Robert Braden. 2005. Tussle in cyberspace: defining tomorrow's internet. *IEEE/ACM Transactions on Networking (ToN)* 13, 3 (2005), 462–475.
[8] Jan Damsgaard, Mihir A Parikh, and Bharat Rao. 2006. Wireless commons perils in the common good. *Commun. ACM* 49, 2 (2006), 104–109.
[9] Rodérick Fanou, Gareth Tyson, Pierre Francois, and Arjuna Sathiaseelan. 2016. Pushing the frontier: Exploring the african web ecosystem. In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 435–445.
[10] Nick Feamster and Jennifer Rexford. 2017. Why (and How) Networks Should Run Themselves. *arXiv preprint arXiv:1710.11583* abs/1710.11583 (2017).
[11] Jay W Forrester. 1971. Counterintuitive behavior of social systems. *Theory and Decision* 2 (1971), 109–140.
[12] Jay W Forrester. 1994. System dynamics, systems thinking, and soft OR. *System dynamics review* 10, 2-3 (1994), 245–256.
[13] Benjamin Frank, Ingmar Poese, Yin Lin, Georgios Smaragdakis, Anja Feldmann, Bruce Maggs, Jannis Rake, Steve Uhlig, and Rick Weber. 2013. Pushing CDN-ISP collaboration to the limit. *ACM SIGCOMM Computer Communication Review* 43, 3 (2013), 34–44.
[14] Jim Gettys and Kathleen Nichols. 2012. Bufferbloat: Dark buffers in the internet. *Commun. ACM* 55, 1 (2012), 57–65.
[15] James Gleick. 2011. *Chaos: Making a new science*. Open Road Media.
[16] Ralf Grötker. 2015. The Citizens' Internet: The Many Threats to Neutrality. (2015).
[17] Alok Gupta, Dale O Stahl, and Andrew B Whinston. 1997. The Internet: A future tragedy of the commons? In *Computational approaches to economic problems*. Springer, 347–361.
[18] Mark Handley. 2006. Why the Internet only just works. *BT Technology Journal* 24, 3 (2006), 119–129.
[19] Junchen Jiang, Vyas Sekar, Ion Stoica, and Hui Zhang. 2017. Unleashing the potential of data-driven networking. In *International Conference on Communication Systems and Networks*. Springer, 110–126.
[20] Ben Jones, Roya Ensafi, Nick Feamster, Vern Paxson, and Nick Weaver. 2015. Ethical concerns for censorship measurement. In *Proceedings of the 2015 ACM SIGCOMM Workshop on Ethics in Networked Systems Research*. ACM, 17–19.
[21] Daniel Kahneman and Gary Klein. 2009. Conditions for intuitive expertise: a failure to disagree. *American psychologist* 64, 6 (2009), 515.
[22] Sebastian Kaune, Ruben Cuevas Rumin, Gareth Tyson, Andreas Mauthe, Carmen Guerrero, and Ralf Steinmetz. 2010. Unraveling bittorrent's file unavailability: Measurements and analysis. In *IEEE Conference on P2P Computing*.
[23] Vikas Kawadia and Panganamala Ramana Kumar. 2005. A cautionary perspective on cross-layer design. *IEEE Wireless Communications* 12, 1 (2005), 3–11.
[24] Frank Kelly. 2001. Mathematical modelling of the Internet. In *Mathematics unlimited—2001 and beyond*. Springer, 685–702.
[25] Srinivasan Keshav. 2018. Paradoxes of Internet Architecture. *IEEE Internet Computing* 22, 1 (2018), 96–102.
[26] Daniel H Kim. 1995. *Systems thinking tools: a user's reference guide*. Pegasus Communications.
[27] Daniel H Kim. 1999. *Introduction to systems thinking*. Vol. 16. Pegasus Communications Waltham, MA.
[28] K. Kompella. June 2017. The Self-Driving Network: How to Realize It. *North American Network Operators Group (NANOG)* (June 2017). https://www.nanog.org/sites/default/files/1_Kompella_The_Networking_Grand_Challenge.pdf
[29] Logan Kugler. 2016. What happens when big data blunders? *Commun. ACM* 59, 6 (2016), 15–16.
[30] R. W. Lucky. 2006. Tragedy of the commons [Reflections]. *IEEE Spectrum* 43, 1 (Jan 2006), 88–88. https://doi.org/10.1109/MSPEC.2006.1572368
[31] Donella Meadows. 2007. *Leverage points: Places to intervene in a system*. Hartland: The Sustainability Institute.
[32] Donella Meadows, Jorgen Randers, and Dennis Meadows. 2004. *Limits to growth: The 30-year update*. Chelsea Green Publishing.
[33] Albert Mestres, Alberto Rodriguez-Natal, Josep Carner, Pere Barlet-Ros, Eduard Alarcón, Marc Solé, Victor Muntés-Mulero, David Meyer, Sharon Barkai, Mike J Hibbett, et al. 2017. Knowledge-defined networking. *ACM SIGCOMM Computer Communication Review* 47, 3 (2017), 2–10.
[34] George E Mobus, Michael C Kalton, et al. 2016. *Principles of systems science*. Springer.
[35] Rishab Nithyanand, Sheharbano Khattak, Mobin Javed, Narseo Vallina-Rodriguez, Marjan Falahrastegar, Julia E Powles, ED Cristofaro, Hamed Haddadi, and Steven J Murdoch. 2016. Adblocking and counter blocking: A slice of the arms race. In *CoRR*, Vol. 16. USENIX.
[36] Sven Nyholm and Jilles Smids. 2016. The ethics of accident-algorithms for self-driving cars: an applied trolley problem? *Ethical theory and moral practice* 19, 5 (2016), 1275–1289.
[37] Cathy O'Neil. 2017. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books.
[38] Ingmar Poese, Benjamin Frank, Bernhard Ager, Georgios Smaragdakis, Steve Uhlig, and Anja Feldmann. 2012. Improving content delivery with PaDIS. *IEEE Internet Computing* 16, 3 (2012), 46–52.
[39] Junaid Qadir, Arjuna Sathiaseelan, Liang Wang, and Jon Crowcroft. 2016. Taming limits with approximate networking. In *Proceedings of the Second Workshop on Computing within Limits*. ACM, 9.
[40] Barath Raghavan. 2015. Abstraction, indirection, and Sevareid's Law: Towards benign computing. *First Monday* 20, 8 (2015).
[41] Barath Raghavan and Justin Ma. 2011. Networking in the long emergency. In *Proceedings of the 2nd ACM SIGCOMM workshop on Green networking*. ACM, 37–42.
[42] George P Richardson. 2011. Reflections on the foundations of system dynamics. *System Dynamics Review* 27, 3 (2011), 219–243.
[43] Barry Richmond. 1994. Systems thinking/system dynamics: let's just get on with it. *System Dynamics Review* 10, 2-3 (1994), 135–157.
[44] Peter M Senge. 2006. *The fifth discipline: The art and practice of the learning organization*. Broadway Business.
[45] Dennis Sherwood. 2011. *Seeing the forest for the trees: a manager's guide to applying systems thinking*. Nicholas Brealey Publishing.
[46] D Sterman John. 2000. Systems thinking and modeling for a complex world. (2000).
[47] David Peter Stroh. 2015. Systems thinking for social change. *Vermont: Chelsea Green Publishing* (2015).
[48] Ryan W Thomas, Daniel H Friend, Luiz A Dasilva, and Allen B Mackenzie. 2006. Cognitive networks: Adaptation and learning to achieve end-to-end performance objectives. *IEEE Communications Magazine* 44, 12 (2006), 51–57.
[49] Kentaro Toyama. 2015. *Geek heresy: Rescuing social change from the cult of technology*. PublicAffairs.
[50] Adrienne Yapo and Joseph Weiss. 2018. Ethical Implications of Bias in Machine Learning. In *Proceedings of the 51st Hawaii International Conference on System Sciences*.