

System Dynamics in Action¹

Ignacio J. Martinez-Moyano^{a,@,b,@}

a: Argonne National Laboratory

@: martinez-moyano@anl.gov

b: The University of Chicago

@: martinez-moyano@uchicago.edu

Introduction

Every day, decisions are made that change the world. These decisions then create new information that affects future decisions in an iterative, continuous, set of feedback processes. Feedback is at the core of the system dynamics (SD) approach [1]. The SD approach was invented by Jay Forrester in the late 1950s at the Massachusetts Institute of Technology in the United States [2,3]. The SD approach, part of the operations research field and identified by systems philosophers as part of the first three waves of systems thinking [4], emerged out of servomechanisms engineering and control theory and has been applied to manage a wide range of complex problems in natural, engineered, and social systems. The SD approach combines the theory of information feedback, human decision making, and computer simulation to create a “computer-aided approach to policy analysis and design” [5]. SD modelers, in order to make better decisions and solve complex problems, are concerned with both *how* and *why* systems change over time. The SD approach favors a point of view that originates from within—an endogenous approach—to explain and propose alternatives for change. SD uses a broad system boundary to provide such endogenous structural explanations of system behavior.

SD Modeling Process

The SD modeling approach is based on the scientific method, in which hypotheses are postulated and data are collected to test those hypotheses. In SD, as in any other rigorous modeling and simulation approach, “adherence to the scientific method is essential for developing models that will stand the test of time” ([6], p. 16). There are many ways of developing SD models (see Table 1). Approaches may vary, but the main goal is always the same: to increase the understanding of the problem being studied in order to intervene in the system and change reality.

¹ This paper was originally written in 2019 and is Chapter 2 of The Handbook of Systems Thinking.

Randers ([7], p. 119)	Richardson and Pugh ([8], p. 16)	Sterman ([9], p. 86)	Martinez-Moyano and Richardson [10]
• Conceptualization • Formulation • Testing • Implementation	• Problem identification and definition • System conceptualization • Model formulation • Analysis of behavior • Model evaluation • Policy analysis • Model use or implementation	• Problem articulation (boundary selection) • Formulation of dynamic hypothesis • Formulation of a simulation model • Testing • Policy design and evaluation	• Problem identification and definition • System conceptualization • Model formulation • Model testing and evaluation • Model use, implementation, and dissemination • Design of learning strategy/infrastructure

Table 1: Some approaches to the system dynamics (SD) modeling process.

Recognizing the interactive nature of complex systems, the SD modeling process is “never a linear sequence of steps” ([9], p. 87); it is always iterative. The iterative nature of the modeling process is designed to maximize learning about the drivers of problems and to produce the most actionable set of recommendations possible. The SD approach pursues two important outcomes: *understanding of the problem and the system* and *understanding of the model* (see Figure 1) because the process “begins and ends with understanding of a system and its problems” ([8], p. 16).

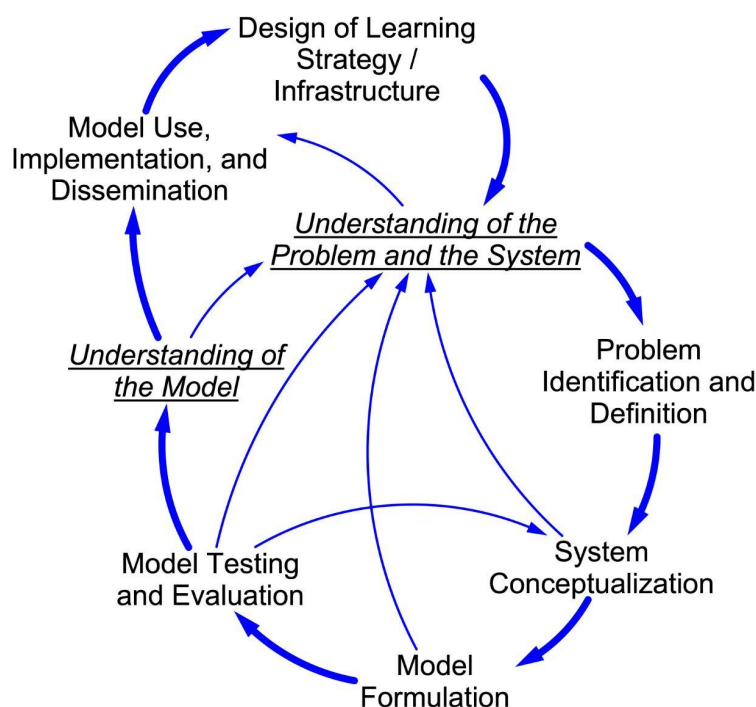


Figure 1: SD modeling process (Martinez-Moyano and Richardson 2013).

In SD, clarifying a single problem of interest increases the likelihood of having a clear purpose for the modeling process. A critical first action in the modeling process is to converge on a useful, and shared, identification and definition of the problem of interest. In practice, before settling in on a useful definition, some model formulation and testing are developed. It is acceptable, and desirable, to circle back to produce a more useful and nuanced definition of the problem under study after some initial work has been completed. In addition, model formulation and reformulation “usually continues nearly to the end of the project” ([6], p. 16). Clarifying the modeling purpose is paramount because “the art of model building is knowing what to cut out, and the purpose of the model acts as the logical knife” ([9], p. 89). Without purpose, there is no guideline for when to stop adding elements to the model (i.e., where to draw the boundary). A clear statement of the model’s purpose gives the modeler “the imaginary line separating what is considered (for modeling purposes) to be inside the system and what is considered to be outside” the system boundary ([8], p.48). The model boundary should be broad enough to include all the components whose interactions generate the problematic behavior of interest. A broad model boundary “that captures important feedback is more important than a lot of detail in the specification of individual components” ([9], p. 96), because the focus is on explaining the dynamics of interest, not on explaining the components of the system.

After problem identification and definition, the system conceptualization process follows. This is where decisions are made on what parts of reality to include and how to describe them [7]. In this phase, the modeler poses a theory—a dynamic hypothesis of the problem—of what creates the observed problematic behavior. It is a hypothesis because it is a best guess, always tentative and subject to change and testing. The dynamic hypothesis is “an initial concise overview of the feedback structures believed to be responsible for the problem behavior” ([10], p.111), which, connected with the reference mode of behavior, becomes the guiding structure for further development of the feedback mechanisms responsible for the problem of interest. A dynamic hypothesis is an effort to create an endogenous explanation of the problem of interest. Endogenous factors—those internal to the system—are the only ones accessible to system actors to accomplish change. Exogenous factors—outside the system, although sometimes illustrative and attractive as likely explanations of behavior—are not useful for change, because they are not within the realm of control of system actors [11]. If the modeling effort proposes that the solution to the problem of interest is to change something outside of the system actors’ control, then the solution is not a true solution.

In complex problems, many different stakeholders have varying degrees of interest and different points of view about what constitutes the problem and a solution (for an example of a large-scale application of the system dynamics approach [12]). The dynamic hypothesis, and the model built afterwards, needs to include as many relevant stakeholders’ points of view as possible. In addition, the modeler needs to ensure that the correct level of aggregation is used to capture the relevant dynamics of interest. The level of aggregation can vary from highly aggregated—represented with aggregated stocks—to the completely disaggregated—represented with individual agents. As long as the choice adequately matches the problem characteristics and available data (or additional, newly collected data), a broad model boundary that captures important feedback mechanisms is key to modelers and stakeholders producing an endogenous explanation.

The operational nature of SD requires that formal models start small and simple, very close to the dynamic hypothesis. For example, a good SD model of milk production should explicitly represent cows [13]. This allows the modeler to more easily add complexity to the model as needed, with greater trust and understanding on the part of stakeholders, which can generate early, frequent conversations in the model-building process. The result is enhanced understanding and structure refinement. Other defining characteristics of a good SD approach include equations written during formulation that make sense and follow real-world processes; models that are representations of how the real world operates (stakeholders should recognize real-life meaning in every parameter and variable); explicitly stated units of measurement in all parameters that correspond to those used in the real world and dimensional consistency tests that ensure logical coherence of model equations; and exclusion of information not available to decision makers in the real system (unless the modeler would like to identify the value or the effect that perfect or complete information would have on a process). In SD models, there is no room for factors that merge several components into unrecognizable items that magically modify the world, because such factors are not truly available to decision makers in the real system.

Model formulation and model testing are two highly interrelated steps in the model-building process because, although “validation and verification of models is impossible” ([9], p. 846), model testing should start when the modeler writes the first equation. If the modeler waits until the model is fully formulated to start the testing process, the modeling effort is unlikely to succeed. Following Popper’s [14] falsification theory of scientific discovery, all tests should aim to falsify the theories we pose, not to confirm such theories. The more tests we conduct, the more likely it is that others will be confident that the model is robust and that the modeler did all the necessary work to adequately represent the problematic behavior. Modelers interested in SD have developed specific tests to uncover flaws and improve many aspects of dynamic models [9,15,16]. According to Sterman ([9], p. 858–861), general areas of assessment include tests related to boundary adequacy, structure assessment, dimensional consistency, parameter assessment, extreme conditions, integration error, behavior reproduction, behavior anomaly, family member (class of system), surprise behavior, sensitivity analysis, and system improvement. Dimensional consistency, for instance, is an extremely important and basic test of model confidence. If a model is not dimensionally consistent, this is evidence that the modeler does not understand the problem’s basic mechanisms of operation or is not careful enough to clearly state the units of measurement of the parameters used in the model. In either case, the resulting model loses credibility. Once all variables have clearly identifiable, real-world counterparts (and units of measurement), the modeler should test for the effects of extreme values because “the model should behave in a realistic fashion no matter how extreme the inputs or policies imposed on it may be” ([9], p. 869). In a milk-production model, for example, no milk should be produced when there are no cows.

Modeling should not “stop at comprehensive analysis and reporting” ([17], p. 134); after extensive modeling and evaluation, a thoroughly tested running model should be used to explore the problem of interest with the stakeholders and other interested parties, keeping in mind implementation and system change. When using the model, it is important to tell clear, coherent stories based on model results, using language that the stakeholders can relate to and pictures (representations) of model structure that they can recognize in the real system. Model-based system stories—illustrated using simplified causal diagrams—can be used with stakeholders as powerful mechanisms to create a long-term learning process for change. A small, simplified model—a boundary object that enables knowledge to be shared and

transferred between and among stakeholders [18]—can be developed and used to explain counterintuitive model behavior that gives rise to real system insights and interesting patterns of behavior.

The modeling process is iterative; it builds on knowledge and insights developed throughout the iterations. This allows for enhanced understanding, additional knowledge, and more powerful insights every time the process is completed. The model becomes a powerful embodiment of knowledge about the problem and system. In addition, the modeling process itself is a source of learning, knowledge, and insight for stakeholders and modelers [19].

Tools for Systems Thinking and System Dynamics

Causal-loop diagrams and stock-and-flow diagrams are the two tools most commonly used to map the feedback structure of systems in both system dynamics and systems thinking. There is no general agreement about which should be used first. Although some SD scholars advocate for the use of causal-loop diagrams first, due to their simplicity (see Table 11 in [10]), this approach is not necessarily problem-free [20] or straightforward [21]. In complex systems, relevant stocks are relatively few compared to feedback loops, so stock-and-flow diagrams allow modelers to focus their attention on a few key variables of the system instead of on several feedback loops that could influence the problem. However, focusing on only the stocks of the system can cause a modeler to lose endogenous sight of the problem. A mixed approach is ideal.

Causal Links and Causal Loops

Causal links are used to clarify how one variable influences another within the structure of complex systems. A causal link is represented with a single-headed arrow that starts at the “cause” and ends at (or points to) the “effect.” The arrow is always single headed because it represents a clear directional effect (e.g., X influences Y) that is not necessarily true when/if reversed. In addition, in order to represent the type of influence, a sign is added to the arrow. A positive (“+”) sign represents a positive causal link (positive polarity) and means that, all else equal, the two variables will follow the same trajectory over time; increases (or decreases) in the variable at the beginning of the arrow will result in increases (or decreases) in the variable at the end of the arrow. Similarly, a negative causal link (“−”) means that, all else equal, the two variables will follow opposite trajectories over time; increases (or decreases) in the variable at the beginning of the arrow will result in decreases (or increases) in the variable at the end of the arrow. For example, elaborating on an example in Sterman ([9], p. 13), “eggs” and “chickens” are connected with a positive causal link (See Figure 2). Eggs and chickens follow the same trajectory of behavior over time; when eggs increase (or decrease), chickens increase (or decrease).



Figure 2: Positive causal link.

Causal loops are closed causal connections that form circular relationships among variables; the simplest causal loop emerges when two variables are interconnected with causal links. Once a circular relationship is identified, linear thinking is replaced with feedback thinking, and a richer picture of the operation of the system emerges. In order to analyze complex causal maps, the unit of analysis is always one single causal link, assuming everything else remains constant (*ceteris paribus*), so the effect of each cause is isolated when multiple causes and multiple causal loops coexist. The shortest possible loop consists of two variables (“vertices” or “nodes” in network and graph theory) connected with two directional causal links (“edges”) represented by directional arrows. The length of a loop is a function of the number of variables in it. Shorter loops tend to operate faster than longer loops. In Figure 2, an initial theory connecting eggs and chickens is advanced. By connecting the variables causally in both directions to create a closed feedback mechanism (see Figure 3), we can say that as chickens increase, eggs increase. After a full cycle of the loop demonstrates an initial increase in eggs, the model generates further increase in eggs. This is called a “reinforcing” loop (identified with an “R”), or positive feedback loop, because it reinforces the original change experienced. It is worth mentioning that a reinforcing loop reinforces both growth and decline tendencies. When the original change goes down after a full cycle of the loop, the behavior of the variable of interest is reinforced to go further down. Reinforcing cycles can act as engines of growth or engines of deterioration (depending on the direction of the behaviors they reinforce).

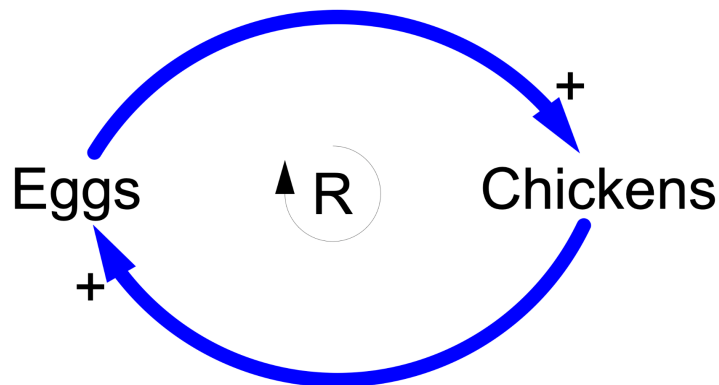


Figure 3: Positive feedback loop.

Positive feedback loops reinforce the original behavior of the variable of interest and negative feedback loops (sometimes called “balancing” or “goal seeking”) tend to counter the original tendency and balance the result. In Figure 4, the chicken and egg theory is extended to include the effect of road crossings. As

chickens increase, road crossings (by chickens) also increase, because a higher number of chickens is available. At the same time, as road crossings increase, chickens decrease. Therefore, after one full cycle, an original increase in chickens has the final effect of a decrease in chickens due to the effect of increasing road crossings, which in turn decreases chickens. The original tendency (or behavior) has been “balanced.” In this example, we created a simple representation of a more elaborate theory about the relationship between chickens, eggs, and road crossings. In order to simplify the representation, many assumptions were made implicit. For example, why do chickens decrease when road crossings increase? That question may have many answers. Depending on the answer, the representation could become confusing and could hold—or not. In order to avoid confusion about assumptions and to increase model transparency, it is important to thoroughly document all aspects of the diagrams and the assumptions used when creating them. Transparency “is an important attribute of useful models because it enables users to identify and understand the assumptions, relationships, and data used” ([22], p. 199).

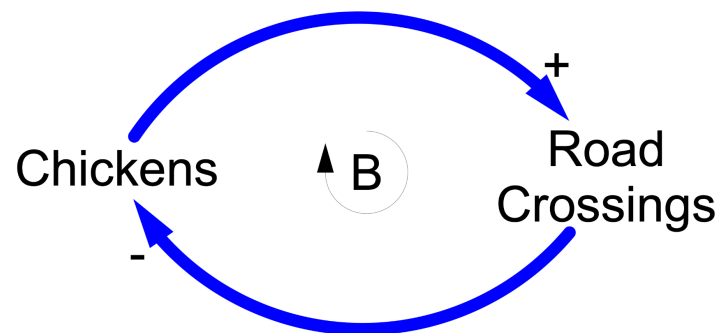


Figure 4: Negative feedback loop.

When creating causal-loop diagrams, be careful the stories that the causal paths allow you to tell make sense throughout the whole causal chain. In addition, it is critical to test how the system story is told when the initial change is reversed (instead of starting with an increase in the variable of interest, start with a decrease). Sometimes, a different initial change reveals important omissions that become problematic for stakeholders and/or people who are unfamiliar with the problem under study. For example, in Figure 4 we advanced the balancing loop connecting chickens and road crossings. We followed the causal chain, starting with an increase in chickens, finding that after a full cycle of the loop, chickens decrease. The story makes sense and is illustrative of what could happen to the chicken population. However, we did not answer why chickens decrease when road crossings increase. If we start exploring the loop by stating that the chicken population decreases, this would lead to a decrease in road crossings, which would lead to an increase in chickens (road crossings and chickens are connected with a negative causal link). This story implies that decreasing road crossings somehow creates chickens. Although the fuller logical path resolves the seemingly illogical statement, this should not happen. Figure 5 provides an alternative to resolve the issue. In Figure 5, as road crossings increase, the chicken death rate increases, which, in turn, decreases the chicken population. This makes the causal path clearer regardless of initial increase or decrease in chickens. In addition, adding “chickens death rate” to the causal path clarifies why an increase

in road crossings decreases the population of chickens (i.e., the chickens die when they cross the road), ruling out alternative plausible explanations (e.g., the chickens are kidnapped after they cross the road, the chickens fall off the edge of the world after they cross the road). Although creating the simplest possible causal diagrams is always a goal, it is important to avoid not fully specifying the causal paths (in order to avoid confusion and/or logical inconsistencies).

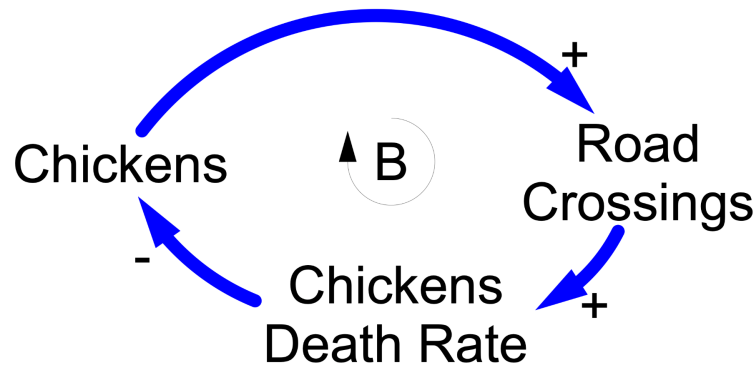


Figure 5: Extended negative feedback loop.

When diagramming the feedback structure of real complex systems, it is common to have multi-loop structures because the real world is inherently a multi-loop environment. Multi-loop diagrams emerge when variables are connected to multiple loops, acting as a bridge among feedback loops. For example, in Figure 6, a more comprehensive theory of chicken dynamics is presented. In this more elaborate theory, chicken population increases as eggs increase (via the reinforcing loop) and, at the same time, it decreases as road crossings increase and the death rate increases (via the balancing loop). The behavior of the two interacting causal loops depends on several factors that are not explicit in the diagram and that will determine the strength of each loop's gain. The gain of a feedback loop is "the product of the gains of its constituent links" ([23], p. 30) or "the strength of the signal returned by the loop" ([9], p. 145). The gain of the link between two variables is essentially how much the "effect" variable changes given a change in the "cause" variable (ratio of the output to the input). If, in the case of Figure 6, the reinforcing cycle has higher gain than the balancing cycle, the type of behavior expected will be different than if the two loops have identical gain. In causal-loop diagrams, the time dimension is not represented explicitly; the variables within the causal loop change over time as one variable influences the next in the chain.

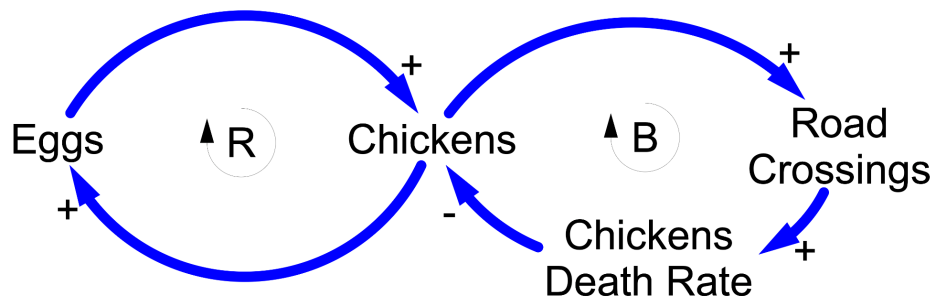


Figure 6: Multi-loop diagram.

Stocks and Flows

Causal loops are very effective at describing the structures of complex systems. However, focusing only on the loops shows only a partial view of the structure. In complex systems, the stocks (or “levels”) represent the accumulations that provide inertia to the system and “characterize the state of the system and generate the information upon which decisions and actions are based” ([9], p. 192). Accumulations, or stocks, are filled and depleted over time just like bathtubs are. In the same way water accumulates in a bathtub based on how much water flows in and drains out, in real systems, stocks accumulate material and nonmaterial things over time through changes in their inflows and outflows. Mathematically, a stock is the integral of the difference between the inflow(s) and the outflow(s), plus the initial condition of the stock. Examples of stocks include world population, money in a bank account, CO₂ concentration in the atmosphere, the U.S. national debt, etc. The stocks can only be changed by changes to inflows and or outflows connected to them. There is no other option. Therefore, if the problematic behavior under study is related to a stock, then the solution should be directed to its flows. In order for a stock to decrease, the net flow should be negative (inflow < outflow). As long as the inflow > outflow, the stock will continue to rise (even at a small pace). This is why, even when nations manage to reduce the deficit (flow) in the budget, the national debt (stock) continues to rise. The only way to reduce the debt (stock), is to run a surplus (flow) in the budget. Stocks persist, flows do not.

In SD, stocks are represented with rectangles and the flows that change them are represented with double arrows (representing pipes) and faucet-like icons (representing valves that control the flows) going into and out of the stocks (see Figure 7). Cloud-like icons represent both sources and sinks for the flows. These sources or sinks represent infinite stocks; these feed the flows and are outside the boundaries of the model, and their associated causal mechanisms are not of interest to the modeler. They can be ignored—or collapsed into the source/sink. Continuing with the eggs and chickens theory, the eggs stock is modified by egg laying (inflow) and egg hatching (outflow). In order for the eggs stock to increase, either the egg-laying rate must increase (*ceteris paribus*) or the egg-hatching rate must decrease (*ceteris paribus*).

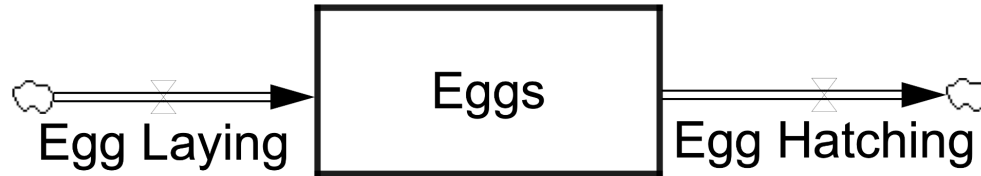


Figure 7: Egg stock and related flows.

Connecting Structure and Behavior

The SD approach is designed to help people understand the mechanisms present in complex systems that create the behavior of such systems. Via computer simulation, it can connect a set of underlying feedback structures to a behavior of interest as observed in the real world and intervene in the system to change the behavior. Policy insights and policy implementation can be identified and tested in coordination with stakeholders and other system actors. The development of a model of the criminal justice system exemplifies how the SD approach was used to identify the connection between structure and behavior, resulting in change to public policy.

On February 11, 2015, Executive Order 15-14 was signed, establishing the Illinois State Commission on Criminal Justice and Sentencing Reform (“Commission”). The Commission recognized that overpopulation of the Illinois prison system had reached critical levels, with dangerously high recidivism, and an alarmingly high cost to taxpayers (~\$1.3B for the Department of Corrections and ~\$131M for the Department of Juvenile Justice every year). The Commission’s goal is to study the criminal justice system and to make recommendations for amending state laws, policies, and procedures to reduce the State’s prison population by 25% by 2025.

A year prior, the Office of Research of the Cook County Sheriff’s Office, which oversees the largest jail in the United States, initiated with the Illinois Sentencing Policy Advisory Council an effort to use SD modeling to understand the dynamics of the jail population. Figure 8 depicts the initial criminal justice system conceptual model used to understand key dynamics of the population. Two stocks identified in the system play major roles: inmates and bonded individuals. Three additional support stocks determine how the dynamics of the system play out over time: courts capacity, bonding capacity, and post-bonding capacity. At the core of the dynamics in this system, the recidivism process—the process by which inmates leave the jail and then come back again—was identified to be crucial. After being booked, individuals become inmates while they wait to have their cases processed in court. Inmates stay out of jail if they qualify to be let out on bond. Bonded individuals who comply with all court requirements, and are not found guilty, are released. Those who do not comply with court requirements, are arrested again with new charges, or are found guilty, go back to jail. In order to expedite the flow of people out of jail via

bonding, additional “bonding capacity” is needed. This is achieved by authorizing additional bail bond agents and/or by relaxing the requirements to let people out on bail.

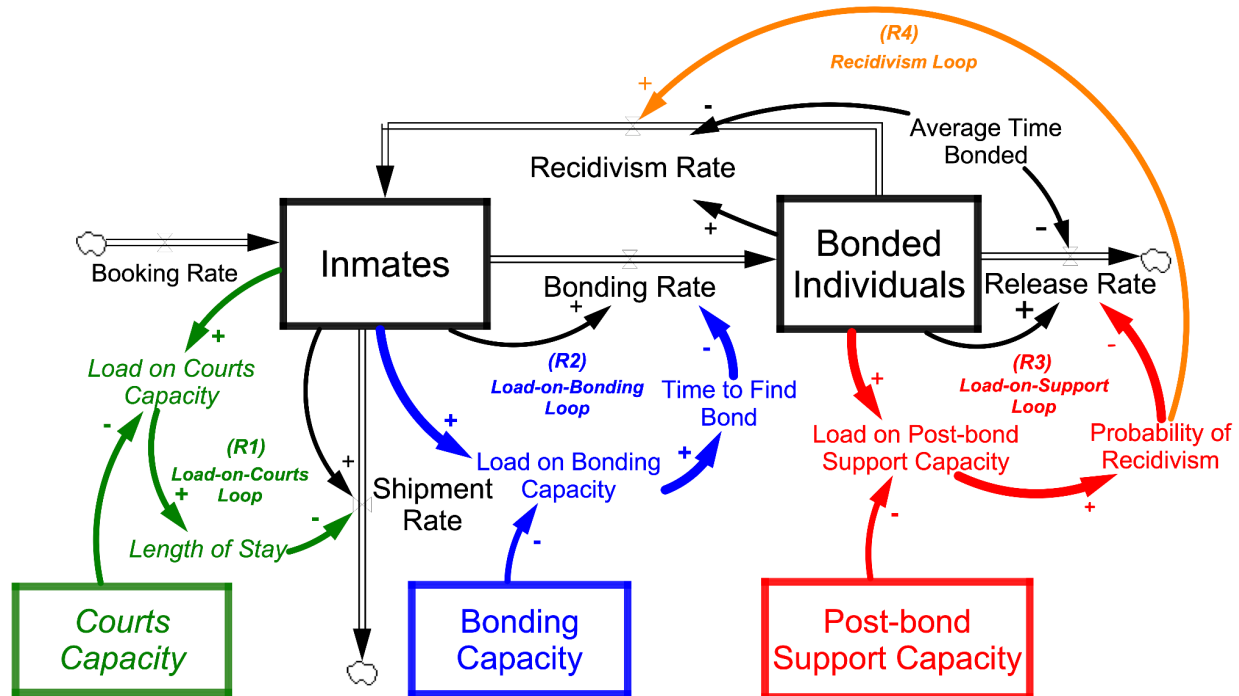


Figure 8: Criminal justice system conceptual model.

Some vulnerable bonded individuals, in order to successfully complete the process, are willing to comply but need support (e.g., childcare, transportation to court). Post-bond support capacity services provide such support.

One proposal to reduce overcrowding in jail was to increase the number of individuals who leave jail on bail. To explore the consequences of such a policy, a formal model, based on notional data, was used to simulate the effect of increasing bonding capacity by 20% (MoreBC run, Figure 9a). The simulation resulted in a net increase in the total number of people in the criminal justice system (from 3,200 to 3,783; +18%), whereas increasing the post-bond support capacity by 20% (MorePBC run, Figure 9b) had the opposite net effect (from 3,200 to 2,318 people; -28%). The model was initialized in equilibrium.

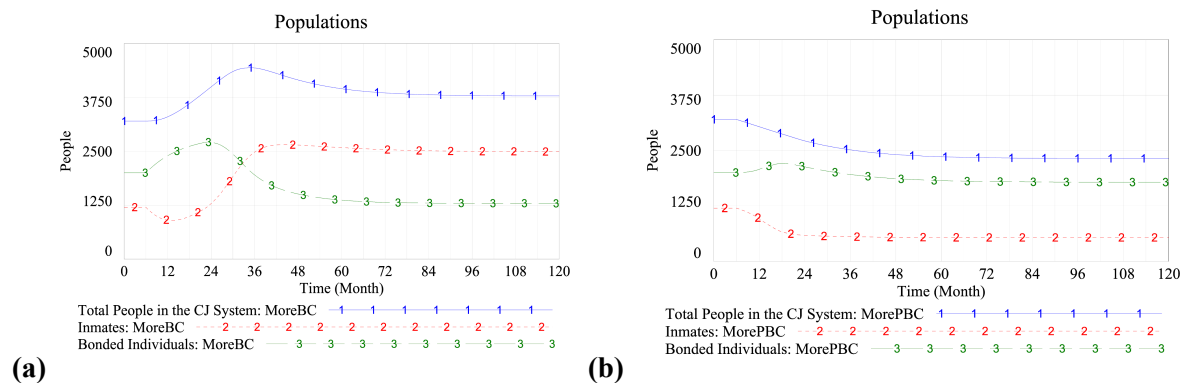


Figure 9: Criminal justice system conceptual model behavior.

The simulation results show that increasing the bonding capacity creates short-term desired effects on the inmate population (see Figure 9a, times 6–12) only to have important unanticipated consequences (sharp increases, times 12–42) due to the overwhelming long-term effect on post-bonding support capacity services. Overloading increases recidivism, resulting in an increased number of inmates. Investing in post-bond support capacity would have a sustained desired effect on inmate population (see Figure 9b, times 24–120), leading to an overall better systemwide state of affairs (compare the blue lines, labeled with the number “1”, in Figures 9a and 9b).

Closing

Feedback and endogeneity are core concepts of dynamic complexity in SD. Consequently, a closed-loop approach enables SD models to go beyond open-loop thinking and reveal, in a holistic way, *why* things change over time. In SD, the level of aggregation is always a function of model purpose; there is never a “correct level of aggregation independent of model purpose” ([24], p. 20). In order to produce a reliable answer, the SD approach develops empirically based, broad-boundary models that provide explanations, based on their feedback structures, of the behavioral evolution of the problems of interest.

Today, SD embraces methodological pluralism more than ever in order to evolve, adapt, and take advantage of advances in computer processing speed and storage, data availability and analysis, and social science research and analytical methods (for examples, see [25,26]). While expanding the toolkit for model development and analysis, testing, and evaluation is still part of the challenge [27,28], effort in clarifying the main elements of dynamic complexity is ongoing [29]. The development of clear data-integration processes, which enable the interaction and usability of large amounts of real-time data feeds with SD models, can transform the types of SD applications we will see in the future (for an example of a potential application, see [30]). Methodological considerations such as the ones expressed here may be a part of a possible upcoming fourth wave of systems thinking that is not only concerned with the application of the method but would also consider its evolution.

Acknowledgements

The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan.

<http://energy.gov/downloads/doe-public-access-plan>

References

1. Richardson GP. Feedback Thought in Social Science and Systems Theory. USA: University of Pennsylvania; 1991. Available: <https://dl.acm.org/doi/abs/10.5555/532214>
2. Forrester Jay W. Industrial dynamics. New York--London: Massachusetts Institute of Technology and John Wiley and Sons. 1961.
3. Forrester JW. Principles of Systems, Wright. Allen Press/MIT, Cambridge, MA. 1968;274: 12.
4. Midgley G. Systemic Intervention: Philosophy, Methodology, and Practice. Springer, Boston, MA; 2000. doi:10.1007/978-1-4615-4201-8
5. Richardson GP. System dynamics. Encyclopedia of Operations Research and Management Science Kluwer Academic Publishers Boston. 1996;656: 660. Available: https://www.researchgate.net/profile/Jan_Polowczyk/publication/340183370_Microstructure_and_Macrobehavior_of_Centralized_and_Decentralized_Systems/links/5e7cc0fe299bfla91b7eb82a/Microstructure-and-Macrobehavior-of-Centralized-and-Decentralized-Systems.pdf
6. Homer JB. Why we iterate: scientific modeling in theory and practice. System Dynamics Review. 1996;12: 1–19. doi:10.1002/(sici)1099-1727(199621)12:1<1::aid-sdr93>3.0.co;2-p
7. Randers J. Elements of the System Dynamics Method. J Oper Res Soc. 1997;48: 1144–1145. doi:10.1057/palgrave.jors.2600456
8. Richardson GP, Pugh AL. Introduction to System Dynamics Modeling with DYNAMO. J Oper Res Soc. 1997;48: 1146–1146. doi:10.1057/palgrave.jors.2600961
9. Sterman J. Business Dynamics: Systems Thinking and Modeling for a Complex World. McGraw-Hill Education; 2000. Available: <https://play.google.com/store/books/details?id=QcVZswEACAAJ>
10. Martinez-Moyano IJ, Richardson GP. Best Practices In System Dynamics Modeling. System Dynamics Review. 2013;29: 102–123.
11. Martinez-Moyano IJ. A Primer For System Dynamics Modeling And Simulation. 2018 Winter Simulation Conference (WSC). 2018. pp. 261–275. doi:10.1109/WSC.2018.8632550
12. Peterson S, Bush B, Inman D, Newes E, Schwab A, Stright D, et al. Lessons from a large-scale systems dynamics modeling project: the example of the biomass scenario model. Syst Dyn Rev. 2019;35: 55–69. doi:10.1002/sdr.1620
13. Olaya C. Cows, agency, and the significance of operational thinking: Cows, Agency, and Operational Thinking. System Dynamics Review. 2015;31: 183–219. doi:10.1002/sdr.1547
14. Popper K. The Logic of Scientific Discovery. Routledge; 1959. Available: <https://play.google.com/store/books/details?id=LWSBAGAAQBAJ>

15. Senge PM, Forrester JW. Tests for building confidence in system dynamics models. *System dynamics, TIMS studies in management sciences*. 1980;14: 209–228. Available: <http://static.clexchange.org/ftp/documents/roadmaps/RM10/D-2926-7.pdf>
16. Barlas Y. Formal aspects of model validity and validation in system dynamics. *System Dynamics Review*. 1996;12: 183–210. doi:10.1002/(sici)1099-1727(199623)12:3<183::aid-sdr103>3.0.co;2-4
17. Roberts EB. Making system dynamics useful: a personal memoir. *System Dynamics Review*. 2007;23: 119–136. doi:10.1002/sdr.374
18. Carlile PR. Transferring, Translating, and Transforming: An Integrative Framework for Managing Knowledge Across Boundaries. *Organization Science*. 2004;15: 555–568. doi:10.1287/orsc.1040.0094
19. Forrester JW. “The” model versus a modeling “process.” *System Dynamics Review*. 1985. pp. 133–134. doi:10.1002/sdr.4260010112
20. Richardson GP. Problems in causal loop diagrams revisited. *System Dynamics Review*. 1997;13: 247–252. doi:10.1002/(sici)1099-1727(199723)13:3<247::aid-sdr128>3.0.co;2-9
21. Coyle G. Qualitative and quantitative modelling in system dynamics: some research questions. *System Dynamics Review*. 2000;16: 225–244. doi:10.1002/1099-1727(200023)16:3<225::aid-sdr195>3.0.co;2-d
22. Martinez-Moyano IJ. Documentation for model transparency: I. J. Martinez-Moyano: Documentation for Model Transparency. *System Dynamics Review*. 2012;28: 199–208. doi:10.1002/sdr.1471
23. Oliva R. Structural dominance analysis of large and stochastic models: Structural Dominance Analysis of Large and Stochastic Models. *System Dynamics Review*. 2016;32: 26–51. doi:10.1002/sdr.1549
24. Sterman J. System dynamics at sixty: the path forward: J. Sterman: System Dynamics at Sixty. *System Dynamics Review* . 2018;34: 5–47. doi:10.1002/sdr.1601
25. Anderson EG Jr, Lewis K, Ozer GT. Combining stock-and-flow, agent-based, and social network methods to model team performance. *System Dynamics Review*. 2018;34: 527–574. doi:10.1002/sdr.1613
26. Anderson EG. Applying Sterman’s proposed principles of modeling rigor to hybrid models combining multiple simulation methods. *System Dynamics Review*. 2019;35: 8–14. doi:10.1002/sdr.1624
27. Richardson G. *Analytical Methods for Dynamic Modelers*. MIT Press; 2015. Available: <https://play.google.com/store/books/details?id=Gh3iCgAAQBAJ>
28. Ríos-Ocampo, J. and M. S. Gary. Using analytical equations to represent nonlinear relationships.

System Dynamics Review. 2022; 38: 354-370. doi.org/10.1002/sdr.1718

29. Yasarcan, H. The four main elements of dynamic complexity. System Dynamics Review. 2023; 39: 171-179. doi.org/10.1002/sdr.1731
30. Rahmandad, H. and J. Sterman. Quantifying the COVID-19 endgame: Is a new normal within reach? System Dynamics Review. 2022; 38: 329-353. doi.org/10.1002/sdr.1715