

How Do Complex System Behaviors Affect Program Theory, Logic Models, Methodology and Data Interpretation?

Jonny Morell

Presentation to the [2026 Meeting of the European Evaluation Society](#),
Lille France Oct 26th – 30th

Why Evaluate With Respect to The Behavior of Complex Systems?

The programs we evaluate are complex systems that are nested in contexts that are themselves complex. Those systems exhibit behaviors that are seldom recognized when programs are designed, implemented, managed, and evaluated. The result is suboptimized programs being evaluated with suboptimized logic models and inadequate methodologies. To illustrate, a typical logic model might explain change in terms of multiple outcomes and impacts, all connected via many unambiguous paths. But knowledge of complex system behavior would enrich that model by including equilibria and self-organization as intervening between the status quo and change. Invoking those behaviors would result in a program more likely to succeed, evaluated by a more correct model, and tested with more appropriate methodologies.

Early Stages In the Evaluation Lifecycle

At early stages in a program lifecycle, evaluators can work with commissioners to educate them about how the behavior of complex systems governs their programs. As a practical matter, the place for this education is during joint logic model development. Evaluators need to engineer these opportunities if those opportunities are not already present. The process of communicating with stakeholders is not difficult as long as the evaluators have the appropriate knowledge about what complex systems do.

Later Stages in the Evaluation Lifecycle

The second tactic is to evaluate a program with respect to complexity regardless of whether complexity was ever considered as part of program theory or the logic model that derived from that theory.

This can be done because it would not increase the burden of data collection. Nor would it have to invoke exotic and unfamiliar methodologies. The data needed would almost certainly be available through whatever mechanisms were put in place to do a non-complexity-based evaluation. To pick up on the previous example, the dynamics of self-organization are likely to be detected in qualitative data that were collected during staff interviews. Reporting these findings to commissioners constitutes a teaching moment.

Complex System Behavior, Logic Models, and Program Theory

The discussion above was cast in terms of how complex behavior can inform the models we use to construct an evaluation, and through the models, to methodology and data interpretation, i.e., on the empirical aspects of evaluation. But there is another dimension to consider. Models are the constructs we use to test program theory, and it is program theory that drives (or should drive) planning. Of course much evaluation is applied on its face, with no consideration of its implications for the (often implicit) theory that underlies program funding, design, implementation, and management. But when theory does guide programming choices, those choices should recognize “how the world works”. And the world works as a complex system.

Why Bother?

What will it take for all our talk about complexity and complex systems to have operational consequences for how we design logic models, devise methodologies, interpret data, and converse with stakeholders? What it would take is a shift in the conversation from the fact that a system is complex to an inspection of what complex systems do – how they behave.

To illustrate, compare two statements. 1) I will analyze my data with statistics. 2) I will do an analysis of variance and if I am satisfied with the P value I will follow up with conservative post-hoc tests. The first statement alerts us to an epistemology, but it tells us nothing about the data analysis. The second statement is operational. Good research design cannot stop with a statement about epistemology. It must cast its plans in operational terms.

So too with how knowledge of complexity can play a role in evaluation. I may find it interesting to know that the program I am evaluating is a complex system. But that knowledge will not tell me whether I should consider the joint consequences of system equilibrium, the depth of that equilibrium's attractor space, and proximate and distal tipping points when I build logic models to reflect program theory or decide on an evaluation design. As another example, it is one thing to observe that an outcome is emergent from a set of program outputs. It is something else to realize that if the outcome is emergent, it cannot be explained as a linear combination of its inputs, that it needs its own definition and metric, and that it has consequences for tracking causation backward from outcome to program.