A Systems Motivation for the Design of Human-AI Interdependence

Tyler Cody,
Stephen Adams,
Peter Beling

Systems and Information Engineering
University of Virginia

1 Abstract

Most applications of artificial intelligence (AI) involve interdependencies with humans. Left unconstrained, these interdependencies can become complex and entangled. Human- and AI-centric approaches to studying human-AI interdependence look outward from the human and AI respectively, and seek remedies to complications through general understandings. In contrast, systems-centric approaches use a top-down, holistic perspective from which context can be taken into consideration. From such a perspective, system goals and design emerge as mechanisms for an awareness and knowledge of a system's human-AI interdependencies. In this paper, we position systems theory as a formal framework for the joint consideration of human and AI aspects of interdependence, and we explore the role of goals and design in creating an operational understanding of human-AI interdependence, a step towards autonomous human-machine teams.

2 Introduction

Machine learning has moved beyond being a research field to being a workable approach for building autonomous functions into systems. However, a principled understanding of human-AI interdependence has yet to emerge. Traditionally, research into interdependence is human-centric, looking outward from the cognitive and affective nature of humans (Lubars and Tan, 2019), or AI-centric, looking outward from the algorithmic nature of AI (Carroll et al., 2019). Such approaches often lead to a scientific study of humans, AI, and their interrelationships. However, these studies are often ill-defined, and seek very general, broad-based understandings that, while satisfactory for scientific studies, can be insufficient from an engineering perspective due to a lack of context.

In contrast, herein we use systems theory to take a systems-centric approach to studying human-AI interdependence. Instead of looking outward from human or AI perspectives, we look top-downward on the two, contextualizing their relationship with respect to the system within which they are embedded, of which they themselves may only be parts of a greater whole. Systems theory is the theoretical study of systems, and general systems theories (Mesarovic and Takahara, 1975; Von Bertalanffy, 1969) serve as useful super-structures for

connecting seemingly disparate fields. A synthesis of systems theory and learning theory offers a mathematical foundation for relating learning algorithms to their systems, and, given systems theory's well developed ties to psychology and organizational theory (Kast and Rosenzweig, 1972; Von Bertalanffy, 1967), for understanding the interrelationships between humans and learning algorithms from a systems perspective.

Considering the system brings system goals, and, in turn, a scope and focus to the elicitation of human-AI interdependencies. In this paper, we motivate the focusing effect of contextualizing human-AI interdependence using system goals, and discuss how system design for such goals can create awareness and knowledge about human-AI interdependence in a system. To do this, we show how learning fits into a canonical general systems theory framework (Mesarovic and Takahara, 1975) and use the developed theory to both model the life cycle of a learning system and to posit a related system goal. We use the developed model as the basis for a case study where goals, design, and human-AI interdependence are explored in the context of a life-cycle engineering problem.

This paper is organized as follows. First we discuss relevant literature and provide some historical background on the development of general systems theory. Then we review and extend Mesarovician abstract systems theory. Subsequently we present a framework for relating learning systems to system life cycles, and explore an application in condition-based maintenance for machinery where generalization is dependent on an explicit acknowledgement of human-AI interdependence. We conclude with a synopsis and remarks.

3 Background

3.1 Human-AI Interdependence

A contrast can be drawn between human-, AI-, and systems-centric approaches to studying human-AI interdependence. Human- and AI-centric research are briefly described with the intent to provide a framing for this paper's position, but not to provide a complete literature review. Human-centric research into human-AI interdependence draws on philosophical, psychological, and organizational understandings of human behavior, cognition, and emotion. Lubars and Tan (2019) provide exemplary research in the human-centric study of human-AI task delegation (Horvitz, 1999; Lee and See, 2004; Parasuraman et al., 2000). AI-centric research, in contrast, focuses on mathematical and algorithmic aspects of human-AI interrelationships, wherein AI systems often take their natural, algorithmic representations and humans are modeled mathematically, sometimes in terms of relevant AI algorithms, other times as information- or decision-theoretic agents. In recent work, Carroll et al. (2019) do this approach justice when researching coordination between optimal or near-optimal AI and sub-optimal humans by modeling both AI and humans as machine learning agents. Both Lubars and Tan (2019) and Carroll et al. (2019) provide literature reviews with research following human- and AI-centric approaches, respectively.

3.2 General Systems Theory

General systems theory is the study of general systems, and its motivation is well-described by Ludwig von Bertalanffy, a founding father of the field, using an observation (Von Bertalanffy, 1969):

> "... there exist models, principles, and laws that apply to generalized systems or their subclasses, irrespective of their particular kind, the nature of their component elements, and the relationships or 'forces' between them. It seems legitimate to ask for a theory, not of systems of a more or less special kind, but of universal principles applying to systems in general."

Bertalanffy's efforts began in force in the 1950's and gave a banner to countless researchers in philosophy, engineering, mathematics, and science whose contributions helped shape and proliferate systems thinking and supported the establishment of systems engineering as a discipline.

Shortly thereafter, mathematical approaches to general systems theory emerged, replacing metaphors with axioms, and forming the basis of model-based systems engineering. Set-theoretic general systems theories, in particular, were mathematical superstructures that allowed for the generalization and integration of more specialized theory (Mesarovic and Takahara, 1989; Wymore, 1967). Set-theoretic systems theories provided formal, top-down theory leading to principled, top-down methodologies.

The development of AI is closely related to cybernetics (Wiener, 2019), and through cybernetics, to mathematical general systems theory. Learning theory and machine learning are sub-fields of AI concerned with learning to perform tasks from data (Bishop, 2006; Vapnik, 1995). At an intuitive level, systems and learning theory are closely related in their pursuit of general understandings and general methods. This relation persists at a technical level, where, in the large, systems theory is a theory of sets and learning theory is a theory of probability, or, in other words, a theory of measures on those sets.

4 An Abstract Systems Theory of Learning

Mesarovician systems theory, referred to by its originator first as *general systems theory* (Mesarovic and Takahara, 1975), and later by the more distinguishing title *abstract systems theory* (Mesarovic and Takahara, 1989), is a set-theoretic mathematical framework that seeks to realize von Bertalanffy's vision in a way that is "simple, elegant, general, and precise"(Mesarovic and Takahara, 1975). Concepts are introduced axiomatically, and mathematical structures needed to do so are introduced such that the formalisms are precise without losing their generality. In arguing for such a mathematical approach, Mesarovic states that (Mesarovic and Takahara, 1975):

> "...the investigation of the logical consequences of systems having given properties should be of central concern for any general systems

theory which cannot be limited solely to a descriptive classification of systems."

Mesarovic develops his theory using a process he refers to as *formalization*. The process involves giving a verbal description a precise mathematical definition using as few axioms as possible. Mathematical structure is added as needed to specify systems properties of interest. Thus, the formalization approach to general systems theory naturally identifies how fundamental particular systems properties are relative to others.

4.1 Mesarovician Systems Theory

We will now review the Mesarovician systems theoretic framework (Mesarovic and Takahara, 1975). First we define a system.

Definition 1. System.
A (general) system is a relation on non-empty (abstract) sets,

$$S \subset \times\{V_i : i \in I\}$$

where $\times$ denotes the Cartesian product and I is the index set. A component set V_i is referred to as a system object.

The formalization procedure continues by adding additional structure to the elements of system objects, or to the system objects themselves. Input-output systems are defined accordingly.

Definition 2. Input-Output Systems.
Consider a system S, where $S \subset \times\{V_i : i \in I\}$. Let $I_x \subset I$ and $I_y \subset I$ be a partition of I, i.e., $I_x \cap I_y = \emptyset$, $I_x \cup I_y = I$. The set $X = \times\{V_i : i \in I_x\}$ is termed the input object and $Y = \times\{V_i : i \in I_y\}$ is termed the output object. The system is then

$$S \subset X \times Y$$

and is referred to as an input-output system. If S is a function $S : X \to Y$, it is referred to as a function-type (or functional) system.

Mesarovician systems theorists use the formalization approach to specify properties and arrive at specific classes of systems. The relationships between classes of systems can then be formally studied using a category theory of systems (Mesarovic and Takahara, 1989). What results is a mathematically explicit understanding of how very general classes of systems relate to each other.

4.2 Learning as a Mesarovician Input-Output System

Learning can be thought of as learning a map f,

$$f : X \to Y$$

We call f the *learning task*. Algorithmic learning uses a learning algorithm $\mathcal{A}$ and sample D to select a hypothesis h for the learning task f from a set of hypotheses $\mathcal{H}$.

$$\mathcal{A} : D \to h, h : X \to Y, h \in \mathcal{H}$$

Using this high-level formulation of learning, we define a learning system in terms of component sets. Subsequently, we show our definition of a learning system to be a specialization of input-output systems.

Definition 3. Learning System.
A learning system S is a relation on the 5-tuple of component sets $\{\mathcal{A}, D, \mathcal{H}, X, Y\}$ such that,

$$S : \mathcal{A} \times D \times \mathcal{H} \times X \to Y$$

where

$$\mathcal{A} : D \to h, h : X \to Y, h \in \mathcal{H}$$

$\mathcal{A}$ is the learning algorithm, D is the sample set, h is a hypothesis from the set of hypotheses $\mathcal{H}$, X is the input object, and Y is the output object.

Proposition. Learning systems are a special case of input-output systems.
Proof: *Let system $S \subset \times\{V_i : i \in I\}$ with index set I and component sets V_i be such that $X' = \{V_i : i \in I_x\}$ and $Y = \{V_i : i \in I_y\}$ where $I_x \cap I_y = \emptyset$ and $I_x \cup I_y = I$, and $S : X' \to Y$. Let X' be such that $\{\mathcal{A}, D, \mathcal{H}, X\} = \{V_i : i \in I_x\}$. Then $S : \mathcal{A} \times D \times \mathcal{H} \times X \to Y \iff S : X' \to Y$ is a function-type input-output system.*

In other words, a learning system is an input-output system that takes as input an algorithm, a sample, hypotheses, and an input object, and outputs an output object. In training, the algorithm uses the sample to select a hypothesis, and in operation, the selected hypothesis produces an element of the output object when given an element of the input object. Thus, we have shown how a general formulation of learning connects to abstract systems theory, and, in turn, to the broader body of systems research. Such a formalization, when taken in concert with systems formulations of psychology (Von Bertalanffy, 1967) or organization theory (Kast and Rosenzweig, 1972), can help uncover foundational, systems theoretic aspects of human-AI interdependence.

5 Life Cycles of Abstract Learning Systems

We can add additional mathematical structure to more closely study the life cycle of a learning system. System life cycles describe the evolution of system structure and behavior over time, including non-physical changes like changes in goals, policies, and standards. Life cycle engineering is an important sub-field of systems engineering (Alting, 1995; Blanchard et al., 1990), and addressing life cycle engineering concerns related to the use of AI is an emerging research topic. By simplifying the broad topic to focus on system behavior, we can develop an abstract systems framework for studying the life cycles of learning systems.

Given an evaluation function v,

$$v : h \rightarrow \mathbb{R}$$

that maps from a learned algorithm to the reals, and a real threshold ϵ, we can define the neighborhood N,

$$N = \{P(X, Y) | v(h) \geq \epsilon\}$$

of probability distributions where the learned algorithm h performs satisfactorily according to v.

To the extent that $P(X, Y)$ represents system behavior, N represents the set of system behaviors to which a learned algorithm h can generalize. By the same reasoning, the evolution of system behavior can be captured by the random process $R(X, Y) = R$,

$$R(X, Y) = \{P_t(X, Y) | t = 1, ..., T\}$$

from time $t = 1$ to time $t = T$. Thus, generalization of h from $t = 1, ..., T$ requires that the random process R never leave the neighborhood N. This framework is depicted in Figure 1.

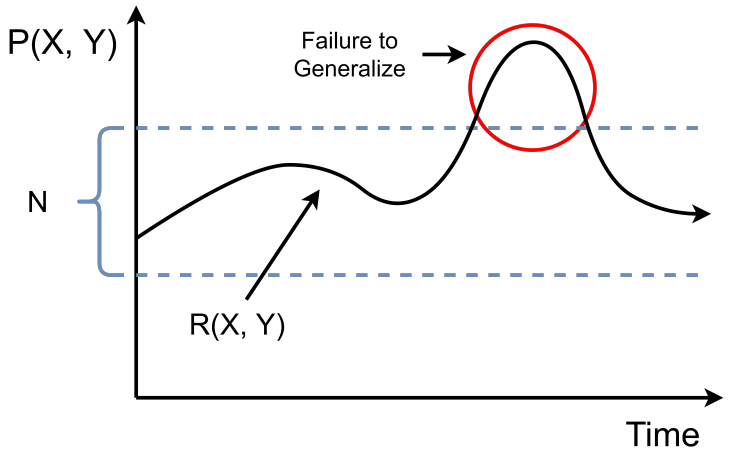

Figure 1: The random process $R(X, Y)$ models the system behavior over its life cycle, while the neighborhood N models the system behaviors where the learned algorithm can perform. When R leaves N, the learning system's performance is no longer satisfactory. To the extent that N is empirically modeled, when R leaves the empirical N, we no longer have model-based, statistical guarantees that the learning algorithm will perform satisfactorily.

This framework provides a general goal for life-cycle engineering of learning systems: the random process R should not leave the neighborhood N. With this goal in mind, the human-AI interdependencies of interest narrow to those

which influence R and N. Next, we explore a case study in condition-based maintenance where awareness and knowledge of those interdependencies are integral to achieving this goal.

6 Human-AI Interdependence in CBM

The sensorization of machinery has given rise to data-driven approaches to condition-based maintenance. Condition-based maintenance (CBM) uses estimates of current and future health states to inform maintenance decisions (Jardine et al., 2006; Peng et al., 2010). CBM has been applied in a wide range of fields including wind turbines (Hameed et al., 2009), rotary machines (Lee et al., 2014), electric motors (Nandi et al., 2005), and lithium ion batteries (Zhang and Lee, 2011). Machine learning approaches to CBM use data and learned algorithms to predict health states (Si et al., 2011).

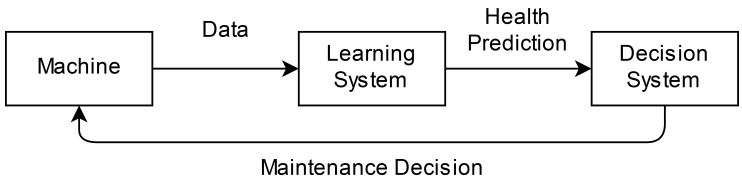

Figure 2: Block-diagram of condition-based maintenance. A learning system receives data from a machine and sends a health state prediction to a decision system, which results in a maintenance decision for the machine.

The structure of CBM processes, as depicted in Figure 2, presents a fundamental challenge to machine learning for condition monitoring. If the maintenance decision changes the machine, such as part repair or replacement, then it also likely changes the distribution of the data witnessed by the learning system, and, if so, can lead to a degradation in the learning system's predictive performance. In previous research, we have found such a phenomenon to exist when hydraulic actuators are rebuilt (Cody et al., 2019).

Maintenance actions can be modeled as events in a machine's life cycle that change the evolution of its system behavior R. The life-cycle engineering goal is that the process R does not leave the neighborhood N of the condition model. In this case, the human is the technician servicing the machine, and the AI is the condition model. If we only consider a single maintenance action, the process R is a piece-wise process,

$$R = \begin{cases} R_0(X, Y) & t < t^* \\ R_1(X, Y) & t \geq t^* \end{cases}$$

where t^* is the time of the maintenance action. The technician's actions can have a large influence over R, and, through it, achievement of the life-cycle goal. If the technician is unaware of this human-AI interdependency, he is likely to

service the machine to recover its physical performance, without considering the dissimilarity a particular maintenance action causes between R_0 and R_1. If he is aware that maintenance affects the condition model's performance, not just the physical performance of the machine, he may take more care to keep R_0 and R_1 aligned, adding it as an additional factor in his mental model for maintenance.

While the technician's trust in the condition model (Lee and See, 2004) and the condition model's robustness to distributional changes caused by the technician are important human- and AI-centric aspects of the interdependence in CBM, the context of a life-cycle goal gives a reason and context to analysis of human-AI interdependence that is tied directly to the CBM system's operational goals. Such factors may emerge in the top-down, goal-oriented analysis of interdependence, but their existence does not make them inherently relevant from a systems engineering perspective. We can go beyond using goals to create awareness of interdependence by using system design for those goals to create knowledge of interdependence.

Consider designing the maintenance protocol so that a condition model learned on R_0 generalizes to R_1. After characterizing the effect of maintenance actions on system behavior, the technician can know the tolerances and specifications necessary to maintain the AI's performance, e.g., criteria for the tensions of fasteners or locations of sensors. In this way, the human-AI interdependencies are integrated into the system by design. The technician not only has an awareness of his influence on the AI, but also an understanding of how his actions can influence it. We have focused on designing the technician's actions to determine the interdependence, but we can consider designing the AI as well. For example, perhaps using transfer learning between maintenance actions gives a flexible or wider neighborhood N, and more leniency to the tolerances and specifications of the maintenance protocol.

Thus, a systems approach serves to put humans and AI in the context of the same goal. A systems goal gives an awareness that both humans and AI jointly influence achievement of the goal, and it directs related analysis. Designing top-down for the goal has a clarifying and crystallizing effect on human-AI interdependence, by not only identifying a structure of interdependence that satisfies the goal, but also by integrating it into the system, thereby defining it.

7 Conclusion

We posited that a systems-centric approach to studying human-AI interdependence is better suited for engineering applications because it provides context. We showed how systems theory relates to learning, and suggested that systems theory can serve as the connective tissue between AI fields like learning theory and human-related fields like psychology and organizational theory. We then developed a framework for the life-cycle engineering of learning systems, identified a general goal, and then showed how that general goal and system design to achieve it serve to create awareness and knowledge about human-AI interdependence in condition-based maintenance systems.

AI and machine learning technologies are moving from laboratories to fielded systems; however, they are far from maturation. Basic and fundamental concerns about the functioning of these technologies in systems are under-addressed. Set-theoretic systems theory offers a formal framework for connecting the human-related and AI theories together. Such synthesized theories, synthesized in the language of foundational model-based systems engineering frameworks, would provide a base for the development of engineering methods that incorporate human-machine interdependence. Although much of the tradecraft for AI engineering will be based on heuristics, empiricism, and lessons learned, systems theory offers a research path for expanding the extent to which such tradecraft is mathematically grounded, and, thus, a path towards principled methodologies for AI engineering.

References

Alting, L. (1995). Life cycle engineering and design. *Cirp Annals*, 44(2):569–580.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

Blanchard, B. S., Fabrycky, W. J., and Fabrycky, W. J. (1990). *Systems engineering and analysis*, volume 4. Prentice Hall Englewood Cliffs, NJ.

Carroll, M., Shah, R., Ho, M. K., Griffiths, T., Seshia, S., Abbeel, P., and Dragan, A. (2019). On the utility of learning about humans for human-ai coordination. In *Advances in Neural Information Processing Systems*, pages 5175–5186.

Cody, T., Adams, S., Beling, P. A., Polter, S., Farinholt, K., Hipwell, N., Chaudhry, A., Castillo, K., and Meekins, R. (2019). Transferring random samples in actuator systems for binary damage detection. In *2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pages 1–7. IEEE.

Hameed, Z., Hong, Y., Cho, Y., Ahn, S., and Song, C. (2009). Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable and Sustainable energy reviews*, 13(1):1–39.

Horvitz, E. (1999). Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 159–166. ACM.

Jardine, A. K., Lin, D., and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20(7):1483–1510.

Kast, F. E. and Rosenzweig, J. E. (1972). General systems theory: Applications for organization and management. *Academy of management journal*, 15(4):447–465.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., and Siegel, D. (2014). Prognostics and health management design for rotary machinery systems—reviews, methodology and applications. *Mechanical systems and signal processing*, 42(1-2):314–334.

Lee, J. D. and See, K. A. (2004). Trust in automation: Designing for appropriate reliance. *Human factors*, 46(1):50–80.

Lubars, B. and Tan, C. (2019). Ask not what ai can do but what ai should do: Towards a framework of task delegability.

Mesarovic, M. D. and Takahara, Y. (1975). *General systems theory: mathematical foundations*, volume 113. Academic press.

Mesarovic, M. D. and Takahara, Y. (1989). Abstract systems theory.

Nandi, S., Toliyat, H. A., and Li, X. (2005). Condition monitoring and fault diagnosis of electrical motors—a review. *IEEE transactions on energy conversion*, 20(4):719–729.

Parasuraman, R., Sheridan, T. B., and Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans*, 30(3):286–297.

Peng, Y., Dong, M., and Zuo, M. J. (2010). Current status of machine prognostics in condition-based maintenance: a review. *The International Journal of Advanced Manufacturing Technology*, 50(1-4):297–313.

Si, X.-S., Wang, W., Hu, C.-H., and Zhou, D.-H. (2011). Remaining useful life estimation–a review on the statistical data driven approaches. *European journal of operational research*, 213(1):1–14.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer science & business media.

Von Bertalanffy, L. (1967). General theory of systems: Application to psychology. *Information (International Social Science Council)*, 6(6):125–136.

Von Bertalanffy, L. (1969). General system theory: Foundations, development, applications.

Wiener, N. (2019). *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press.

Wymore, W. (1967). A mathematical theory of systems engineering: the elements.

Zhang, J. and Lee, J. (2011). A review on prognostics and health monitoring of li-ion battery. *Journal of Power Sources*, 196(15):6007–6014.