2018

Cyber-Physical Embedded Systems with Transient Supervisory Command and Control: A Framework for Validating Safety Response in Automated Collision Avoidance Systems

Daniel K. Trembley

Binghamton University--SUNY, dtrembl1@binghamton.edu

CYBER-PHYSICAL EMBEDDED SYSTEMS WITH TRANSIENT SUPERVISORY COMMAND AND CONTROL

A FRAMEWORK FOR VALIDATING SAFETY RESPONSE IN AUTOMATED COLLISION AVOIDANCE SYSTEMS

BY

DANIEL K. TREMBLEY

BS, ROCHESTER INSTITUTE OF TECHNOLOGY, 2010

THESIS

Submitted in partial fulfillment of the requirements for
the degree of Master of Science in Systems Science
in the Graduate School of
Binghamton University
State University of New York
2018

Accepted in partial fulfillment of the requirements for
the degree of Master of Science in Systems Science
in the Graduate School of
Binghamton University
State University of New York
2018

April 17, 2018

Dr. Sarah Lam, Chair and Faculty Advisor
Department of Systems Science and Industrial Engineering, Binghamton University

Dr. Harold Lewis, Committee Member
Department of Systems Science and Industrial Engineering, Binghamton University

Dr. Yong Wang, Committee Member
Department of Systems Science and Industrial Engineering, Binghamton University

Abstract

The ability to design and engineer complex and dynamical Cyber-Physical Systems (CPS) requires a systematic view that requires a definition of level of automation intent for the system. Since CPS covers a diverse range of systemized implementations of smart and intelligent technologies networked within a system of systems (SoS), the terms “smart” and “intelligent” is frequently used in describing systems that perform complex operations with a reduced need of a human-agent. The difference between this research and most papers in publication on CPS is that most other research focuses on the performance of the CPS rather than on the correctness of its design. However, by using both human and machine agency at different levels of automation, or autonomy, the levels of automation have profound implications and affects to the reliability and safety of the CPS. The human-agent and the machine-agent are in a tidal lock of decision-making using both feedforward and feedback information flows in similar processes, where a transient shift within the level of automation when the CPS is operating can have undesired consequences. As CPS systems become more common, and higher levels of autonomy are embedded within them, the relationship between human-agent and machine-agent also becomes more complex, and the testing methodologies for verification and validation of performance and correctness also become more complex and less clear. A framework then is developed to help the practitioner to understand the difficulties and pitfalls of CPS designs and provides guidance to test engineering design of soft computational systems using combinations of modeling, simulation, and prototyping.

To Sharon A. Andrus

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AI	Artificial Intelligence
C2 (C^2)	Command and Control
C4 ($C^{4)}$	Command, Control, Computer, Communications
C5 (C^5)	Command, Control, Computer, Communications, Cyber-Intelligence
CAS	Collision Avoidance System
CCD	Charged Coupled Device
CE	Conformité Européenne
CPS	Cyber-Physical System
CSE	Cognitive System Engineering
DARPA	Defense Advanced Research Projects Agency
DSS	Decision Support System
EMI	Electro-Magnetic Interference
EPA	Environmental Protection Agency
EPRS	European Parliamentary Research Services
ESA	European Space Agency
FCC	Federal Communication Commission
FIS	Fuzzy Inference System
FSM	Finite State Machine
GD	Gradient Descent
HIL	Human-in-the-loop
HMI	Human Machine Interface
I/O	INPUT/OUTPUT
IoT	Internet of Things
LoA	Levels of Automation
LoRV	*Law of Requisite Variety*
LIDAR	Light Detection and Ranging
MIL	Machine-in-the-loop
MLLR	Multinominal Latent Logistic Regression
MPH	Miles Per Hour
NASA	National Aeronautics and Space Administration
NTSB	National Transportation Safety Board
PLC	Programmable Logic Controller
RADAR	Radio Detection and Ranging
SAE	Society of Automotive Engineers
SIL	Software-in-the-loop
SEPTA	Southeastern Pennsylvania Transportation Authority
TCP/IP	Transmission Control Protocol/Internet Protocol
TUV	Technischer Überwachungsverein
UL	Underwriter Laboratories
VW	Volkswagen

Chapter 1: Introduction

1.1 Thesis Purpose and Intention

Cyber-physical systems (CPS) covers a diverse range of systemized implementations of smart and intelligent technologies. The term “Smart” or “Intelligent” technology is used frequently in describing systems with the inherent ability of an engineered system to perform complex operations without the requirement of a human-agent (Sheridan, 2011). The terms “smart” and “intelligent” are relatively synonymous with only varying degrees of interpretation as to an actual agreed upon definition (Lee, 2015). This thesis researches smart and intelligent technologies a domain that refers to itself as the cyber-physical system. The difference between this research and most papers in publication on CPS, is that most of the research focuses on how levels of automation, or autonomy, are designed into the engineered system, and how the levels of autonomy affect reliability and safety of the system. This research focuses on the human-agent, namely an operator interacting or monitoring the system, as being directly affected by shifts and transitions within a system’s levels of autonomy, while the system is under operation (Chen et al, 2011). As the CPS systems become more and more common, the relationship between the human-agent and machine become ever more complex chorography of engineering, science, and cultural (Tweedale and Jain, 2011).

1.2 Cyber-Physical Systems

1.2.1 Definition of Cyber-Physical Systems

The terms and definitions used in the thesis are complicated by the newness and diversification of CPS technologies. The term cyber-physical system was first used in 2006 and is attributed to

Helen Gill at the National Science Foundation (NSF) for having coined the term (Lee and Seshia, 2012). The context in which the term cyber-physical system is used refers to a computational system that integrates itself into physical processes (Lee, 2015). This is an extremely generic definition but will suffice. The problem with defining a CPS is that it comes in all sizes, forms, and functions. The argument of when CPS came into existence is controversial. Some experts claim that CPS have existed since the early 1980's in industrial manufacturing processes using PLC controllers (Lee, 2015). However, other researchers claim that there is a much longer history and deeper history that surrounds the current model of CPS development and that technologies that use terms such as smart or intelligent are mere marketing attempts at making products "sexier." This research uses the latter rather than the former viewpoint because the evolution of technology depends on the infrastructure of where science, engineering, and society are currently rather than where or when the idea or invention took place. That is, using machines to offload tasks is essentially the whole point of the industrial revolution, and cyber-physical systems is the latest application that uses microprocessors and microcontrollers. The smarts and intelligence of the CPS is the ability to program by software into the hardware the system's ability to be reactive to external information (Tweedale and Jain, 2011).

1.2.2 Definition of Cyber-Physical Embedded Systems

Since there is little agreement on the definition of an embedded system, the embedded systems as presented in this research are the few to many smaller systems that interact with larger systems by a collection of smart technologies that interact with the physical system (Lee and Seshia, 2012). The term cyber-physical system, then, will be used throughout the remainder of the thesis to encapsulate the idea of a system in its entirety, and the definition of an embedded system is the component level of the CPS. Because the definitions are general and describe a wide range of system types, the generalization will work adequately to explain the complex nature of the CPS without the need to understand the lower level details of the embedded system's structure.

The research in the field of CPS and embedded systems is limited; however, it is growing as both the conceptual ideas of CPS and embedded systems usage expands and is rapidly advancing into every area of daily life (Yarmoluk, 2017). For example, the Internet of Things (IoT) is causing much excitement in the way intelligent technologies are used (Lee, 2015). The IoT is composed of embedded intelligent technologies, and thus is considered the cyber-physical system that integrates smart sensors, microcontrollers, and networking technologies in existing products (Tweedale and Jain, 2011). The IoT forms communication networks that interact as information conduits across multiple types of technology platforms. The IoT architecture is being used and adapted in all facets of industrial and commercial based applications (Yarmoluk, 2017).

1.3 Cyber-Physical Embedded Systems Forms and Usage

The definition of the cyber-physical embedded system is therefore general and is mostly used to describe a "thing" with intelligence that can interact in both closed and open loop systems (Lee, 2015). The specific nature of such a system is its relation to its application. It becomes an appliance to a larger framework of a System of Systems (SoS). For example, the cruise control in a car is an embedded system that appeared in automobiles around the mid-1950's. These control systems were analog and used to maintain a car at a constant set speed. Cruise control systems in automobiles today use computers (Vahidi and Eskandarian, 2003). However, the implementation is not what is important but that the same system exists in many different implementations. What needs clarification then is the system or systemness of what is being described. The definition of system is a wide range and class of "things" forming "relations" with its environment (Klir, 2001).

The description of the CPS demonstrates the necessity of viewing the system as a SoS and using the "black box" perspective to form its model, so an attempt to analyze its design can be made (Ashby, 1956). This assumes that the inherent decision-making processes of the CPS, i.e., intelligent system, is in fact operating within design parameters in an environmentally rich input data landscape that uses much of the dominant information to from hypotheses on which to act and react (Chen et al., 2011). In most cases, the external responses of those decisions are only noticed in the observable state. The internal data structures remain internal to the hidden layers that are embedded, and therefore, transparent to the operation of the system. This produces a natural emergent layer of security to the system (Backhaus et al., 2013). The idea of security will be addressed only briefly, but as an abstraction to the ability of the system to defend itself against threat, it becomes a very important topic, but beyond the scope of this thesis.

1.4 Level of Automation in Adaptive Systems

The level of automation is a topic that has relatively disappeared from the literature, which is troublesome because as this thesis maintains, the validation of the system is the demonstration of controlling transient levels of automation (Moradi-Pari et al., 2014). These are adaptive systems. In human factors engineering, the interaction between human and machine is best described as adaptive automation which is thus defined:

> *…refers to human interaction with electro-mechanical control devices that interact with humans such that the allocation of control function (to either human or computer) changes with time to accommodate changes in the conditions of either the physical environment or the human.* (Sheridan, 2011)

This definition somewhat describes the CPS and the involvement of the human-agent with the operation of the system. The problem is with the evolution of the technology that uses "electro-mechanical control devices" to interact with the physical world. This is reinterpreted as the CPS with human-agent in a mix of responsibilities over the system at certain times based on certain conditions (Sheridan, 2011).

The idea that both the CPS and human-agent are decision-making is loosely based on the fact that sensory observation is being used to create or enhance experience. This provides the command and supervisory authority to create actionable events (Chen et al., 2011). In this sense, the case that the system is intelligent is partially made if it were not for the fact that the human-agent is involved. Since the CPS is reactive to the sensory inputs, and the levels of automation are used as a design specification, the reliability and safety factors are determined although not necessarily known. There is a chance of emergent behavior from the system that the designer did not account for. For the system engineer, the system needs to meet or exceed the primary design requirements for the success of the system. The decision-making supervisory controls define the level of automation and become the definition of the system's architecture and robustness of its design (Parnell et al., 2011).

1.5 Propositions of the Thesis

There are three fundamental propositions of this thesis that form the basis of a framework to validate any cyber-physical system in its ability to remain reliable and safe.

1.5.1 Proposition One

The first proposition uses the definition of an embedded system that possesses properties that define cyber-physicality. Decision-making is a general process that leverages the ability of the human-agent and intelligent machine through a physical world of hardware, software, sensors, actuators, controllers, and human physical capital. The first proposition of this thesis is as follows:

> **Proposition 1:** ***The human-agent and machine use similar principles, rules, and guidelines to generate decisions that form actions. The combination is a reactive system to external input stimuli and external output response.***

The human-agent is the person(s) interacting with the CPS. By interacting, the human-agent is either monitoring or physically providing stimulus to the machine, i.e., information (Ruff et al., 2002). The principal operating guidelines to the decision-making process, or structured source system, are generalized. The application of the system determines the operational state in dynamic conditions that are either discrete, continuous, or a combination of both (Novak et al., 2017).

1.5.2 Proposition Two

The second proposition pertains to the composition of the machine state and is considered the algorithm that the system operates under. It is the use of input states to the internal structure of the machine that form the basis of decision-making. The source structure is the external stimulus of information that is forming knowledge and experience and being transformed into actions. This is how an intelligent system would arrive at the actionable state (Tweedale and Jain, 2011). *Proposition 1* assumes an isomorphic relation between human-agent and intelligent machine as each being equally capable of making the same decision by using similar decision-making processes. The second proposition of the research is as follows:

> **Proposition 2:** ***Decisions and actions are only optimal if the system continues to operate at equal or greater performance to the previous machine-state cycle.***

There are many types of decision-making systems that are useful in governmental, business, military, emergency service, air traffic control, etc. decision-making processes (Alippi et al., 2017). The fundamental algorithmic process of decision-making is then deployed on the CPS. Since the CPS bases its performance on collaboration with the human-agent, this is essentially the idea of teamwork because there are at least two decision-making systems involved in a single CPS. This complicates the design paradigm of the system because first, the decisions of the system are not allowed to cause degradation of the overall performance from state-machine cycle to state-machine cycle; and second, the machine's performance is confined to equal or greater to the last time it performed an action. System performance is the requirement to have a perpetually functioning system that is equal or greater to its start state, which goes against the first and second laws of thermodynamics, so the machine is constantly using information to maintain the appearance of perpetual motion (Kang et al., 2018).

In this case, it is the adaptation to uncertain input states that must be sufficiently dealt with by the system for continuation to the next machine-state cycle. A decision that is optimal sets a maximum to the decision-to-action function. Therefore, this requires the system to have knowledge and experience of "good" and "bad" attributes, which are more qualitative than quantitative. The machine-states then become a probabilistic continuous system and not a directly observable deterministic system. This is the definition of an "Intelligent Machine."

1.5.3 Proposition Three

The third proposition uses, and is derived directly from, Ashby's Law of Requisite Variety. The law is stated as follows:

> *The larger the variety of actions available to a control system, the larger the variety of perturbations it is able to compensate* (Ashby, 1956).

The Law of Requisite Variety is also known as the first law of cybernetics (Boisot and McKelvey, 2011). The third proposition is as follows:

> **Proposition 3:** ***The Law of Requisite Variety remains valid and is conceptually necessary to the design of test and validation methodologies where the states of hidden variables within a generative system is not known.***

This proposition sets forth the test and validation methodology that defines the informational entropy in the system. The knowledge of the variables and their states is not as important as quantifying the informational energy and entropy within the system. That is, the information, known or unknown, hidden or observable, forms the context of the system whose external observation will be used to gain deep insight into the internal machine-state decision-making. The validation only needs to identify the guard bands and constraints of the system's behaviors that are either consistent or inconsistent with the requirements of the design. To prove the integrity of the system becomes a multifaceted problem of linear and non-linear optimization (Zhang et al., 2017).

1.6 Significance of the Research

In the current era, where the rise and almost intrusive nature of automated systems is becoming extremely advanced, the need to verify and validate such systems as driverless cars, pilotless planes, deep space probes, and other unmanned systems becomes paramount (Wan et al., 2017). Although a driverless car may have an intelligent machine at the steering wheel, it will also have flesh and blood passengers. Turning complex high reliability tasks over to such intelligent automated systems is in the beta-test phase as of the writing of this thesis (Noh and An, 2018). However, the debate on how to prove the reliability and safety of such systems is controversial and loosely defined in many engineering circles. Here the groundwork of a framework is laid down on how to accomplish such an important and difficult task.

The thesis carries forward several paradigms from game, control, and information theory (Backhaus, 2013; Wiener, 1948; Shannon, 1948). One such paradigm is that all information contained within systems will not be made available, and the composite of information that is available will allow certain mathematical properties of the system to be used. The information will need mapping and interpretation that it conforms to the design and model of the system. By defining an autonomous system using an informational based approach, through a good modeling, simulation, and prototyping activity, it will provide the bounds and operational test limits of the system (Sheng et al. 2017). The model only needs to substantiate the keys elements of the design; whereas, the simulation and prototyping need only factor in the possibilities of conditions that have strong causal relations, such as to the external stimuli (Sheng, et al. 2017; Alippi et al., 2017).

1.7 Challenges of Cyber-Physical Systems

The development of CPS requires a variety of detailed expert knowledge concerning the system's software and hardware architecture, and how it automates decision-making processes, maintains levels of automations, and the internal and external command and control structure of the system (Chen et al., 2011). It is not enough to understand individual characteristics of the above-mentioned categories, but to understand the complex set of dynamical processes at play in harmonious balance when the system is working well, or the imbalances when it is not. Placing such systems in charge, especially in safety critical applications, has engineering, governmental regulatory, and cultural challenges (Sheridan, 2002).

This thesis exemplifies the use decision-making process at hierarchical levels of automation within the cyber-physical system architectures (Robertson, 2010). By leveraging the human-agent ability to help the CPS react, the design builds in a safety factor; however, this will prove difficult to validate. A framework of verification and validation test methodologies of cyber-physical systems, as well as other autonomous systems that use software and hardware based neural computing techniques will be discussed, developed, and designed (Dundar et al., 2017). This framework will help ensure integrity, reliability, and safety of the system for optimal trust.

1.8 Summary and Overview

The thesis is separated into five sections to form a framework of working knowledge to aspects of solving autonomous design related issues. The literature review discusses the historic, current, and future applications of CPS and autonomous systems. The following four sections give a review of decision-modeling, embedded automation design, and levels of automation architectures. The combination of these sections lays the ground work for the basis of a framework to both real and generalized design. Further, two case studies examine automation in CPS are described, and the inherent problem of how automation is both “friend and foe” to the end user. The case study uses a train accident and a plane accident. The final section proposes a framework for use in modeling and simulating the intelligently designed CPS systems. By applying common tools, the framework will be demonstrated as a standardization of tests for verification and validation of intelligent systems using modeling, simulation, and prototyping.

Chapter 2: Literature Review

2.1 A Brief History on Cyber-Physical Systems

In the time period from the late 1940's through the 1950's, control theory, information theory, cybernetics, artificial intelligence, and complexity were being discussed and formulated into mathematical equations, scientific theory, and applied engineering applications (Lee, 2015). The first digital computers were bulky, hard to maintain, and slow. However, it was the gateway to the age of the cyber-physical system. Performance of the machine would increase as larger and larger problems would be thrown at it. Psychology, mathematics, biology, and engineering were brought together to research the human-machine relationship by improved means of automation as technologies in materials allowing a continued growth in the power of digital systems from the analogue systems (James, 1953; Wiener, 1948; Sanders and McCormick, 1993; Von Bertalanffy, 1974).

Automation in the early stages of the 21st century has many promising possibilities. Most machines are now built with some form of automation that adds a layer of ease of use to the machine. From cars to refrigerators to smartphones to GPS devices, the lives of 21st century denizens have become unencumbered from the drudgery of working with a small amount of information, usually hard to obtain, to having mountains of information, easily accessible but difficult to decipher (Gubbi et al. 2013; Sheridan, 2017). For example, refrigerators can upload grocery lists to a Smartphone, and the Smartphone can notify the car, which makes the recommendation of the store to shop based on traffic flow and produce pricing. Cars are now coming equipped with Lane Change Detection, GPS, satellite radio, cruise control, and Collision Avoidance Systems (Safety, 2015; Katzourakis et al., 2014) The idea of carrying a roadmap in the glove compartment is obsolete. Daily life becomes highly optimized and control by systems embedded within the technology.

The telephone system is a good example of a Cyber-Physical System that has maintained longevity and increased the use of automated technology over a period of a century. The common practice of making a telephone call is a great achievement in the telecommunication technology of the past decade. The telephone has been around since 1887, and the Smartphones of 2017 perform the same basic function as phones from 1900. Here, the technological history demonstrates a basic mapping of automation within a framework of solutions to improve on the existing technology. In Figure 2-1, a telephone switchboard operating room is shown from circa 1914. The women are sitting at the switchboards connecting callers to one another by manually inserting a plug from the caller into the receptacle of the callee.

Figure 2-1: Telephone Operators Salt Lake City, Utah circa 1914 (Reyner Media, 2009)

Essentially, the technology of 1914 was manually and mechanically driven by the telephone operators. When someone called into the switchboard, the caller would ask the operator to simply connect them to the phone number of the callee. The problem is that the expansion of the phone system required more and more space for the switchboards, operators, and power systems. The telephone system in this configuration was not very scalable. In Figure 2-2, the Nortel digital switch replaces all the operators by performing the same essential functions, i.e., connecting callers, but with less transparency to how this is being accomplished with end results that are much faster and accurate because of the automation. The switch operates 24 hours a day, 7 days a week for the entire year. Even a rare break for maintenance will not take the whole system offline. It allows scalability from the few hundreds of phones in 1914 to over one-million in 2007.

Figure 2-2: Nortel DMS-100 Digital Multiplexing System (Mudares, 2007)

But the history of Cyber-Physical Systems does not begin there. The first transatlantic telegraph cable was first laid in 1858. A telegraph operator on one side of the Atlantic would send communiqués by Morse code to another operator on the other side of the Atlantic. It would be 100-years further on until the first transatlantic telephone cable was put into service. However, from the isomorphic nature of the technology, the telegraph was in essence a digital communication system. The telegraph operated by sending bits of information in the form of short and long pulses via Morse code. Today, a person making a call from New York to Moabi is converted into a bit stream that is packetized through communication protocols, such as

Transmission Control Protocol/Internet Protocol (TCP/IP), a more sophisticated and robust version of Morse code.

The human and the machine became an important area of study in the engineering profession especially during the World War II era from 1939 to 1945. The term *"cybernetics"* was coined by Nobert Wiener at the Massachusetts Institute of Technology (MIT) (Wiener, 1948). With new technologies and rapid advancement in fields such as aerospace, the importance of the human-machine interface (HMI) was even more closely scrutinized by research using scientific methodologies of descriptive studies, experimental, and evaluation research (Gobbo and Benini, 2013; Sheridan, 2017; Palmer et al., 2003). Job titles such as human factors engineer, ergonomic engineer, and engineering psychologist began to appear in the help wanted ads and became part of the curriculum of engineering colleges and universities. Machines being designed from 1940 onward increased rapidly in complexity, operation, and performance (Zadeh, 2008). The operators of these machines needed to keep up with the learning and training. However, the operators would reach a limit on involvement with the cyber-physical processes. Automating the processes would become the goal to achieve system efficiency and affectedness.

2.2 Classifying System Automation and Autonomy

The term Cyber-Physical System was used circa 2006 and attributed to Helen Gill at the National Science Foundation (NSF) for coining the term (Lee and Seshia, 2012). The CPS computationally integrates itself into a physical process, and although the definition is relatively simple, the difficulty is trying to understand the computational mechanism that underlies that physical system (Lee, 2015). Because of this, a clarification between the definition of autonomy and automation needs to be made:

> **Automation** – *Any mechanical or electronic replacement of human labor (Sheridan, 2002).*

> **Autonomy** - *The attribute of a system to meet mission performance requirements without external support for a specified period of time (Turner, 1985).*

In the definition of *automation*, the term *labor* is used to mean physical or mental labor. However, the definition of "mechanical or electronic" device is the "embedded system." The definition could read as, "Any embedded system mechanical and/or electronic…" which provides further clarity of the system's autonomous designs. An example of an embedded system with mechanical and electronic features would be the television remote. The buttons on the remote would be the mechanical aspect of the system, and the ability to control the channels on a television is viewed as the electronic feature. The user of the remote gains no insight into how the system works, but they can sit on the couch and remotely control the channels to watch without much physical effort. Although using a remote with televisions was a rarity in the first couple of decades of the television, the remote was adopted early on by higher-end models of television and later improved as the television became more digital. Almost no televisions are manufactured that do not require a remote. This is an example of the technology becoming extremely advanced digitally and embedded over time. The human-agent has only to perform channel selection and watching, and with the exception of changing the batteries in the remote occasionally, there is not much else required.

From the above example, it is interesting to note that engineering design in cyber-physical systems has a relation to an economic and social class structure. The early adopters of technologies tend to be high-end products afforded by a wealthy class of consumer, such as luxury car models. The high-end car models have always been accessorized by the latest trends and fashions in technology. For example, although it is hard to imagine a car without a radio and music system, it was common through even the late 1990's where radios, tape players, CD-ROM drives existed as options and expensive add-ons. Today, the dash board of even economy class cars makes these features standard or part of a service package, such as Bluetooth, General Motors OnStar™ or SiriusXM™. With global positioning navigation, backup cameras, satellite radio, hands free communication, lane departure detection, etc., the functionality of the automobile has radically departed from the idea of a 1910 Ford Model-T. That is, an automobile is no longer a purely functional means of getting from point A to B. It is a fully accessorized extension of existing technologies embedding themselves into the structure known as "car". And the push to make the car self-sustaining and driverless is the next phase in this evolution (Borenstein et al., 2017).

2.3 Levels of Technology Development

Currently, there is much literature and research in the driverless car technologies and the companies investing in these technologies. The large and well-known technology companies, such as Tesla, Google, Microsoft, Amazon, and Tom-Tom are willing to compete against long time established companies like Ford, BMW, and Toyota for the driverless car market share. The threshold of available technology appears at the precipice of making the driverless car a reality. There are many technical challenges and regulatory hurdles to overcome, but many companies are achieving success in the beta-testing phase (Borenstein et al., 2017; Lazanyi and Maraczi, 2017).

The driverless car technology comes at the high price that it may takes years to migrate the newer autonomous technology from prototypes into all cars. Current driver assistance systems commonly found in high-end vehicles (circa, 2017) are the Collision Avoidance System (CAS), automatic parallel parking, and lane drift warning (Liu et al., 2017; Schnelle et al., 2017; Katzourakis, et al., 2014; NTSB, 2015). Though these three mentioned features will become standard in economy class cars, it will be by governmental regulatory means that the safety standards of all automobiles sold within the United States will be made. This creates the parity between a high-end and economy car. The National Transportation Safety Board (NTSB) has directed automakers to install forward and rear collision avoidance systems and the backup camera in all cars in the next few years. The NTSB states that 80% of all deaths and injuries would have been prevented if such systems were installed (Safety, 2015). So, the question is, “Why not do it sooner than later?”

The simple answer is economics. A decision is being made purposely by automobile manufacturers to leave low end model cars without the safety features to contain cost. The irony is that everyone shares the same road, and the cars without collision avoidance will still end up colliding with cars with the collision avoidance (Bajpayee and Mathur, 2015). But the technology, although produced and sold circa 2017, is not as mature as the sales and marketing brochures tell. There are different types of collision avoidance systems that have advantages and disadvantages depending on the driving environment. Also, not all CAS has the same performance record and may be more detrimental to the driver who assumes that the system is safe, and therefore, relaxes vigilance (Safety, 2015).

There are three common types of technology used in the CAS. These technologies are essentially the eyes of the car. First, CAS technology can use a LIDAR-based sensor data system. These are fairly accurate systems with fewer false alarms than the other types of technology, but LIDAR systems lack information such as target speed and direction. Speed and direction are critical attributes of a 2-dimensional system in motion (Safety, 2015). Whereas the RADAR-based CAS provides excellent data of target motion and speed, the drawback is that RADAR is easily interfered with by outside sources of electro-magnetic inference (EMI) (Safety, 2015). The third type of CAS technology uses a camera-based data collection system. By using machine vision algorithms, the car sees what a human would see; however, the ability of the algorithm to interpret the captured data and turn the data into speed and distance determines the performance quality of a camera-based CAS. The camera-based CAS is limited to the same factors of human vision such as conditions of poor visibility (i.e., rain, fog, night time, etc.). So ideally, it would seem the design solution should include all three CAS; however, the complexity of such a system would be formidable, and the reliability uncertain. Based on the above discussion, automation challenges the available technology to be able to deliver results that are optimal in terms of good reliability and safety metrics (Young et al., 2017). By defining the autonomous systems at a particular level of automation, the descriptive language that classifies the roles and responsibilities of human-agent and machine-agent will lead to the expectation of performance and operation of any cyber-physical system. The test methodology frames the validation measures necessary to prove the such system requirements (Young et al., 2017; Parasuraman et al., 2000; Sheridan, 2011).

2.4 Usefulness of Autonomy

In a different mode of autonomy, deep space exploration also leverages the use of automated systems. The reason for this is simple. The communication time between Earth and the deep space probe gets longer as distances grow between the two. Radio signals, even at the speed of light, become so delayed that working in anything close to real-time system is impossible. The communication turn-around time could be in the neighborhood of hours and days as in the cases of deep space probes such as New Horizon, Cassini, and Voyager (Reinholtz and Patel, 2008; NASA, 2015; Popken, 2007). Additionally, if data is coming from a deep space probe, the bit-rate (baud) determines the amount of data that will be received per unit of time. This could potentially constrain the system as much more data could be collected than downloaded, especially when high resolution pictures are being sent. So, it is critical to get the memory storage system optimized because radio carrier frequencies used to transmit data over the vast distances of space are low frequency, meaning, less bits are transferred per unit time than if the signal were at a higher frequency.

The propagated communications delay between Earth and a space probe creates the environment where autonomy built into the probe is essential. The probe may need to detect, orient, make decisions, and take actions that are necessary to keep the mission safe. For instance, the New Horizon mission is a deep space probe that reached Pluto in 2016. The distance the probe had to travel was a staggering 3 billion-mile journey that took over 9 years. The probe travelled at a speed of approximately 50,000 mph and approached Pluto and its moons for a one-time encounter before hurtling into the deeper regions of the Kuyper Belt. The near approach to Plutonium System was an event that only took 4 hours to complete before it was all over. At a distance of 3 billion miles, communications take a little more than 5 hours one way. The quickest message turn-around time was approximately 11 hours. Because so little was known about the Plutonium System (uncertainty), the probe was essentially communicating back to Earth the pictures of what lay ahead in its course. The telemetry of the positions of Pluto, its moons, and anything else that might be in the path of New Horizon was a guess. Since no Earth based telescope had the resolution to provide navigation information to New Horizon, the navigation system was control by the probe. The probe sailed through the Plutonium system without incident, all the while collecting data by being able to point the scientific equipment on board with pinpoint accuracy (NASA, 2015).

This is considered a Level 5 type of automated system (SAE INTERNATIONAL, 2014), except that the maneuvering plan for the New Horizon flyby was automated based on the predefined mission parameters and was adjusted in the last months before the Plutonium encounter. However, New

Horizons would autonomously fire its maneuvering thrusters to gain the best angles for the scientific equipment it was carrying. The flyby needed to happen without real-time coordination from ground control because of the distance to Earth. Once New Horizon cleared the Pluto systems, it would take nearly a year and a half to send all the data to Earth that it collected from the 4-hour encounter (NASA, 2015).

Aviation systems safety and accident investigation has been a vast field of research for well over 100 years. The accident investigations become a forensic tool to understand design problems in a CPS. Since the early age of heavier than air aviation, autonomy has been used to assist and aid in the control of the aircraft. The first autopilot was used in 1914, just ten years after the Wright Brothers' first flight. Automation remains a top priority to embed into any cyber-physical system. For example, a B-2 Stealth Bomber is a plane that will not fly without computers and other autonomous control systems. These systems control and supervise the aircraft in flight and are outside the range of the pilot's need to adjust. On February 23, 2008, the United States Air Force lost a $1.4 billion-dollar B-2 as a result of improperly calibrated sensors that the aircraft computer interpreted as correct. The input sensors caused a mismatch between what the automated system was interpreting and what the aircraft was doing. The pilot had no chance to react during takeoff when the plane suddenly flew itself into the ground (USAF, 2008).

The term human factor is used to define the human-agent as an entity that interacts and provides feedback to keep the system operational. In aviation, the operational state is the management of the operation being performed based on the level of criticality (i.e., landing, taxiing, embarking, takeoff, etc.). An example of human-agent and machine interacting poorly happened on June 1st, 2009 when Air France Flight 447 from Rio de Janeiro to Paris, France crashed with the loss of 216 passengers and 12 crew. The events that led up to the crash were barely notable. However, in the matter of minutes, the plane was flying through heavy weather, which is normal for the latitudes at the equator, but suddenly the plane's autopilot disengaged. The plane essentially gave control back to the pilots. (This is a transient down-shift in the level of the automation). However, there was no indication of why the autopilot disengaged, because all instruments appeared normal, except air speed. The air speed indicated that the plane was flying too slow at a very high cruising altitude. The aircrew had control of the plane but were unaware that the airspeed pitot tubes had frozen over. The air speed indication from the flight computer was mismatched to the actual speed of the aircraft. The sensor data was being reported in error. The crash investigation determined that this problem should have been easily recognized by the aircrew. The plane at the time before the crash was operational and safe, but for unknown reasons, the aircrew accelerated the plane, and essentially put the plane into a fatal non-recoverable stall at speeds at an altitude outside the performance limits of the aircraft. The investigation that followed concluded that training of the aircrew was insufficient, and that the plane did not have redundant air speed sensors (BEA, 2012).

2.5 Developing a Cyber-Physical Autonomous System

Science and engineering from the late 1940s onward has brought about many changes in fields that use automation and autonomy in a variety of disciplines such as astronomy, mathematics, computers, aerospace, etc. There is a distinct definition of what the term autonomous means for science and engineering. When speaking about autonomy, the definition of choice is the one used by the aerospace industry:

> *The attribute of a system to meeting mission performance requirements without external support for a specified period of time (Reinholtz, 2008).*

The research concerning the design of engineering systems that embed autonomous features into the design has initiated debate for more than 50 years (Sheridan, 2011). Using Sheridan's classification of autonomy (Table 3-1), design engineers are required to describe and discuss the motivations for choices being made for the design. These choices will inevitably impact the design functionality as related to cost, schedule and quality. If planning for a design with some or all automation, three types of distinctions need to be made (Sheridan, 2002):

1. Command and Supervisory Controls
2. Modes in which the automation is to be used
3. The Failsafe

The first distinction is about who or what is in control during normal operation, and who or what is in control during an actionable event. For CAS, automation is fully implemented from the system, that is, the CAS has full command and control when it detects a possible collision event. In normal operation, the system is monitoring and during the actionable event the car is braking. The human-agent is at all times driving the car, but the CAS can go into operational mode within milliseconds of the possible collision (NTSB, 2015; Vahidi and Eskandarian, 2003; Schnelle et al. 2017).

A cyber-physical autonomous system by definition has decision-making capabilities. There is either a linear or cluster of details that define what actions the system is to take based on sensor data input (Noh and An, 2018). But herein lies the cleverness of cyber-physical system design. The human-agent is part of the system as well as the machine. As levels of automation become greater and more complex, as defined by the requirements of the system, the algorithms and types of Artificial Intelligence (AI) used in design become more difficult to test, especially when an automation level is transient and can shift upward or downward unexpectedly (Sheridan, 2002;

Chen et al., 2011). This is the software of the system, and in the strictest sense, software source code coverage and cyclical complexity measures become very unclear whether the use of a probability-based machine model can be reliably tested and guaranteed safe. If the system optimizes for the choice of best decision, what will the system truly do once the threshold of an activation point is reached or exceeded? This is where the case of the human-agent as a variable within the equation of the CPS is slightly different from the machine's role. The decision whether to interact either partially or fully with the machine is an uncertainty. It is based on human-agent state conditions like training and alertness (Parasuraman et al., 2000; Sheridan, 2011).

The second distinction implies whether the level of automation is optimal for the system. Implementing automation into a CPS may only achieve a level of novelty, but again, it may have adverse effects. This is, of course, why testing is crucial. For example, if the design is to contain an internal monitoring system, and the system does not achieve the performance metrics necessary to be implemented, much time and money could go into fixing a bad design, making one that is barely adequate. The case of Volkswagen (VW) emissions testing is a case in point. When VW was testing the emission system standards for diesel cars, there was a precipice drop of engine horsepower with the emissions system "on". However, VW used the data with the emission systems "on" to report to the Environmental Protection Agency (EPA), but turned "off" the emissions system when the cars went to dealerships. The clients of VW's enjoyed the exceptional horsepower of environmentally friendly diesel cars, when in fact, the cars were polluting the air far beyond the regulatory limits set by the EPA (Mansouri, 2016; Blackwelder et al., 2016).

The third distinction of a failsafe is simple. In the event that things go terribly wrong, can the system recover? Again, this is where the knowledge of environmental conditions coincides with the internal operation of the CPS to provide controlled responses that keep the CPS operating safe at the equal or better performance prior to the event (that is, Proposition 2). In the case of the CAS, the driver of the car has the ultimate say in the matter, as the CAS can be turned off. The driver is, by the design, the failsafe (Petit, 2009).

2.6 The Taxonomies of Levels of Automation

The taxonomies for defining levels of automation are numerous and growing (Sheridan, 2002; ASME, 2012; EU, 2015). There are many similarities in the various definitions of levels of autonomy continuously revisited by designers of autonomous systems. The refinement is based on the science and an engineering acumen of autonomous systems need for both a specific and general purpose. Automated systems are improving and becoming more numerous. The embedded processors and microcontrollers that these systems use have more computational power, higher digital-analog resolutions, faster communications channels, and low power consumption, than just a decade ago. In the embedded autonomous system world, the realm of sensors and data streams is getting more powerful, cost effective, and reliable. The Internet of Things (IoT) revolution that is currently being acknowledged by most technologists as the next growth industry will revolutionize by automating processes through the linking of systems together over a network fabric. IoT is embedding itself into an already existing infrastructure that forms a tapestry of digital information (Yarmoluk, 2017; Gubbi et al., 2013).

The notion that small computer-like systems being used as part of the layer of monitoring and servicing a larger system in scalable designs is not new. The idea of embedded systems has been around since the 1970’s. However, there has been impracticality to using embedded systems. Previous generations of embedded systems have been computationally under-powered, expensive, and unreliable to operate. Also, because of the low level of automation used in systems in the past, most autonomous systems were for the most part unnecessary and added very little to the design. This notion has changed in the last few years that parallels the evolution of cell phone technology. It can be shown that the benchmarks of cost, computational power, and reliability has been achieved to allow this technology to become a part of everyday systems. This essentially is the formation of the "*Internet of Things*" (Chen et al., 2018; Dhanalaxmi and Naidu, 2017; Gubbi et al., 2013).

The agent-based frameworks within these designs are very complex. There are many different types of agents that could be discussed: mobile, intelligent, software, information, etc. (Tweedale, 2012). The word "agent" means anything that interacts with the system and/or its environment. In this research, the consideration to the human-agent with the embedded system is implied. The human interacts with the system in a monitoring task or performs an action task that is required to keep the system operational.

2.7 Testing Architectures for Command and Control

The testing architectures for automation, especially where the human-agent takes command and control within the higher levels of autonomy, complicates the design, and therefore, its testability. Rarely is there an agreement for the type of testing architecture that implements and engages a technology framework that covers all facets and forms. There will be gaps or loss of test coverage based on unknowns and assumptions made. The information within the system exists, but it is hidden and does not advance the knowledge of the system (Shannon, 1948).

This is due to the informational loss within the system. The system is a black box (Ashby, 1956). However, by understanding the requirements of the design, and being able to provide test plans that have knowledge of the system performance in the environment, the cyclical complexities of the software (software-in-the-loop) and firmware (hardware-in-the-loop) to the properties of the system, (meaning, the execution of software in conjunction with the firmware) become inextricably intertwined and abstract. The external stimuli input will invariably cause different outputs as the complexity of the system gets larger. The software and hardware states are controlled through the pre-emptive and cooperative multitasking of the systems (Reinholtz and Patel, 2008; Ryan and Cummings, 2016). Additionally, defining the CPS model as discrete or dynamic, state-transition or time-transition, is analogous to trying to understand Maxwell's Demon or Schrödinger's Cat for these systems.

2.8 Cyber-Physical Systems in the Real World

The concern is the command and control structure of the CPS. The case studies will demonstrate the importance of a command and control (C^2), and the consequences that erupt when the system is not used properly. Command and control is a large area of research across many different systems. The US Department of Defense uses a term called C^4, C^5, or C^{4+1} ISR (Command, Control, Computers, Communications, Cyber, Intelligence, Surveillance, and Reconnaissance) to formulate decision making in strategic and tactical military environments (Arne, 2000). The C^4 model is also used in business and emergency services decision-making systems. Essentially, in any system reacting to its environment, the more complex and diverse the system is, the more sophisticated the decision-making capabilities are required to perform predefined algorithms offloads to the decision-making authority. This becomes the high-level architecture and design of a Decision Support System (DSS) (DHS, 2002; National Research Council, 2006; Arciszewski, 2009).

For example, the computer center would be collecting data for fighting forest fires (i.e., wind, humidity, available area responders, equipment on hand, ground reports, etc.). The computer uses the variety and complexity of the information to make recommendations, such as allocating firefighting resources to an area, or predicting the spread of a wild fire based on wind conditions (DHS, 2002). Systems such as these improve safety and reliability for both responders and bystanders. Responders are allocated to where they are needed most, and the bystanders can be guided away from the potential threats wild fires bring.

The command and supervisory control structure within the design of the CPS has to be carefully mapped out, fully documented, and rigorously tested. For example, the use of “watch-dogs” in CPS are essential to its effectiveness. The watchdog is essentially a timed interrupt to the system where an event has not occurred. If servicing the watchdog fails to occur before time expires, the watchdog will fall into a different set of routines that are not part of the normal operation. However, the problem is twofold and does matter whether the CPS is time or state controlled. The watchdog expires because something happens on the system that prevents the watchdog from being serviced, and the system has to find a recovery path. The command handler, based on the state or time transitions, will perform certain tasks to recover the system. That is, the system should be designed to recover. The control structure is what interacts with the system’s environment. So, based on the needs of the command handler, the control system will interact with sensor and effector to carry out the instructions.

Communication is the single most complex standard to any system (Klir, 1995). Signaling is how a network of systems communicate with one another. How systems communicate with each other is part of the initial design of the system, and without radical redesigning, remains a permanent infrastructure to the lifetime of the design. The CPS consists of the software-in-the-loop, hardware-in-the-loop, and a human-in-the-loop. Using the adopting principles of cognitive psychology and human factors engineering, there is an enormous opportunity to identify and solve problems where learning, memory, attention, and perception are concerned. The development of machine intelligence in artificial systems is isomorphic to that of human intelligence. In the design of machine intelligence, terms like neural networks, genetic algorithms, etc. are used to describe the machine intelligence from an animal intelligence perspective, essentially mapping natural systems onto artificial ones (Holland, 1975).

2.9 Multi-disciplinary Approach of CPS

At the nexus of cognitive psychology, systems engineering, and design engineering is an engineering discipline called joint cognitive systems engineering (CSE). By using multidisciplinary approaches to problems of human factors engineering, the design of systems (i.e., software, hardware, interface, space, etc.) uses certain principles and guidelines to improve and leverage through the arrangement and augmentation of internal and external features – switches, displays, lights, software, hardware, etc. Training is a learning process by which both the machine-agent and the human-agent interact with the CPS to reach the full potential of the design. Keeping the performance of the CPS within safe limits during operations for which it was designed reduces the risk of unwanted occurrences happening, i.e., faults (Kang et al., 2018; Martin et al., 2014; Novak et al., 2017).

For example, an elevator that has voice control will detect passengers stepping into and out of the elevator. The elevator would ask which floor the rider would like to visit and would inform the rider once the elevator reached the desired floor. But is this a good design choice of automation? If the rider does not understand the language of the voice system, cannot speak to the voice system, or cannot hear the voice system, the voice system becomes irrelevant. The designers of the system assumed that all riders would understand the language, would speak back to the request of the automated system, and could hear the voice from the automated system. However, if the rider did not understand the language or how to control an elevator, the elevator and rider would remain in a virtual deadlock. The rider is then assumed by the designer to understand how to use the elevator's manual system. Currently, there are very few human elevator operators, but in the not too distant past, the job description of elevator operator was found in the help wanted ads. Today, elevators are almost all exclusively automated. The economics of having a person as an elevator operator is no longer feasible. The elevator designers have leveraged the assumption that human agency is competent to use an elevator as designed. The human-agent boards the elevator and pushes the button to the desired floor. The machine-agent of the elevator's automated control systems close the doors, and proceeds to the desired floor while stopping along the way to pick up other passengers. The passenger is only singly tasked whereas the elevator is multitasking.

Cognitive psychology is a branch of psychology that concerns itself with mental processes such as learning, memory, language, perception, creativity, problem solving, and thinking. The cognitive psychology that is of interest here is the branch consisting of computer analogies (Figure 2-3). The middle branch of Figure 2-3 is concerned with approaches to intelligence that are most apt to be used in the internal development of the intelligent machines. This is especially the case when computer algorithms are used to create artificial intelligence that mimics human behavior, like in robotics (Chen et al., 2011).

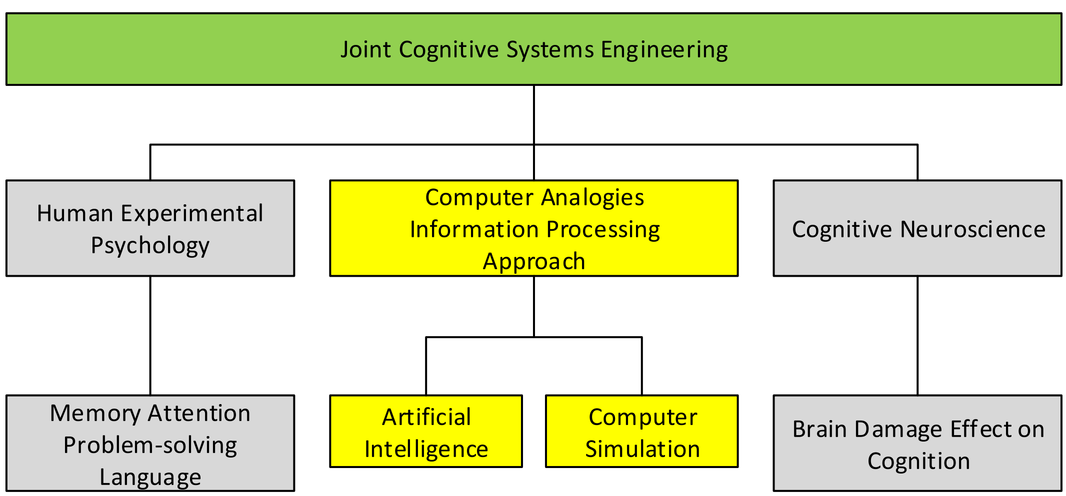


Figure 2-3: Joint Cognitive Systems Engineering Chart (Hollnagel et al, 2005)

Cognitive psychology is the integration of all these aspects into a framework attempting to describe what it is to be human from the perspective of experience. In the mid-20th century, with new technologies concerning warfare being developed, cognitive psychology was at the forefront of this research to understand human performance with newly designed machines. The designers needed to know a person's aptitude to operate the system. One of the first key aspects discovered was that training and attention were extremely important. Attention to the indicators and the actions necessary when situations arose needed to be dealt with quickly. This was especially the case in aerospace design (Casanova et al., 2014; Tweedale and Jain, 2011).

Recognizing and reacting to the system when the system is operating normally is much different than reacting to the system under duress (Smets et al., 2010). It is bridging this understanding with the design, especially in the initial phases of the design, that becomes extremely important (this is why modeling and simulation are very important at the preliminary design phase). For example, if a person is told to monitor a light on the panel, the person has to know what the light means, and what to do about it when it either comes on or goes off. This is the learning and memorization activity that the person acquires in order to have knowledge of and control the system. There lies the simplest of problems with a design: the requirement. “What is the requirement of the light, and what is the person to do if the light blinks?” This example is level 3 in the Sheridan's Level of Automation. If the light blinks, the operator needs know what actions to perform. The scenario goes like this, “What if after performing the action the light is still blinking?” Is the operator going to have enough knowledge of the system to properly assess the situation? Or is it beyond the normal operating strategies and standard operating procedures? Also, what is the safety factor involved? Should the person take flight or fight the response to the system? In any design that leverages the human-agents, it is essential to have an operator who is trained and experienced. This prevents the operator from being subjected to situations beyond their cognitive skills and abilities to command and control the CPS (Tweedale and Jain, 2011; Sheridan, 2011).

Design and systems engineering are multidisciplinary fields within the context of engineering disciplines. The attempt to identify and apply good design and processes is a result of experience and knowledge from several other disciplines, such as construction, mechanics, electricity, chemistry, etc. The system engineer derives requirements of the system where the human-machine interface is concerned based on the specification set by the end user. In essence, the system engineer would be the cognitive systems engineer (CSE) in embedded CPS designs. The strategies employed would be to recognize and define all confluences to the design through an adaptive activity of the modeling, simulation, and prototype testing (Chen et al., 2011; Klien et al., 2004).

The CSE identifies the interface that is necessary to allow for performance, safety, and knowledge to improve adaptive activities between the human-agents and machine-agents. Through good system development of interfaces of control panels, software, ergonomic design, teams, etc., the establishment and certifications can be obtained for the CPS. The amount of operator training and education that allows for the safe and optimal performance of systems is paramount to success. In applying these preliminary adaptive design activities, the development of the system goes through three phase concurrent process of design, analysis, and evaluation (Wang, 2016; Rovere et al., 2016).

2.10 Future Growth and Consequences

Understanding how the human-agent relates to the environment is a growing concern in the world of automation. The desk jobs of accountants, computer programmers, and administrative assistants have a negative consequence of being too easy on the human body. In the early part of the last century, machines in factories were notorious for putting out eyes and chopping off limbs. However, in the 21^{st} century, the human-agent gets carpal tunnel syndrome from typing too much, or too fat from being sedentary at the computer terminal. The economy benefits from doing tasks quicker and more efficiently but has a negative side effect. This is especially true when the human interfaces with technology (i.e., the intelligent machine) that improves performance but sometimes leaves little else to the human-agent except to watch data scroll on a monitor. The health and welfare of a human-agent depends on understanding the human interacting with the machine (Robertson, 2010). Many companies now employ ergonomic standards as a matter of worker compensation, government mandate, and litigation.

The human-agent should always be considered at the center of the design (Robertson, 2010; Parasuraman et al., 2000). Whether the person is the end user of the system or a member of the development team, when designing a new system or upgrading an existing system, the cost associated with implementing automation must be clearly understood. Properly managing -- not only the different teams, but the expectation of the end user, or customer -- will have great economic benefit through cost savings and improved capacities and throughputs. Using modeling and simulation greatly improves the design process of both large and small projects by identifying areas where improvements need to be made before the physical system even exists (Parasuraman et al., 2000).

The human-agent experience of interfacing with and becoming machines may seem farfetched today, but as technology continues developing with societal and cultural norms readily adapting to new technologies, the understanding and implication of how these technologies are used, and what repercussions the technologies will have, is a rich and deep subject open for much debate (Young et al., 2017; Quintas et al., 2017). Technologies that become engrained into society will both have a negative and positive effect, and since it is the desire to have technology work in the most economically efficient manner, it is important to understand the effects once a significantly complex and well-developed autonomous system is deployed. This applies to newer technologies, such as autonomous cars, unmanned aerial delivery vehicles, embryonic DNA editing, quantum computing, etc. It is sufficient to state that a technological system must be useful and must meet all regulatory and safety standards. This is easier said than done.

Humans desire to enhance performance, i.e., using prosthetics, surgical alterations, ingesting substances, etc., and the list of the technologies available on the market, i.e., hearing aids, liposuction, Viagra, anabolic steroids, etc. are many. The near future technologies that are being developed such as autonomous cars, unmanned aerial delivery vehicles, embryonic DNA editing, are still future technologies, but these technologies could be available very soon. Other future technologies that are seriously being considered: cancer fighting nanobots, genetic transplants, ion propulsion space travel, and 3-D printing of organ tissue. The futurist Ray Kurzweil states in his 2005 best seller *The Singularity Is Near,* that by the year 2049, distinguishing between man and machine will become so blurred that the whole paradigm of the human as a stimulus-response organism to technology will no longer be applicable as the human will be part of the new matrix of part organic and part machine. The research of this thesis is scalable to this paradigm shift should it occur.

Science fiction movies consider many possible scenarios and combinations of robotics and artificial intelligence. Often what is first dreamt of in science fiction, later becomes a technological reality (e.g., the Motorola Flip Phone was inspired by the 1960’s television series Star Trek). Consider technologies that offload tasks that seem dreary and full of drudgery. Most households in the United States have dishwashers, washing machines, vacuum cleaners, etc. These are all first world machines automating daily chores that would take much longer in their absence. These machines at one time were deemed luxury, and now they are common-place as they are found in most households in the United States. The machines are also getting smarter by the use of embedded systems. Dryers can now detect when the clothes are dry and stop the drying process. This aids in the efficiency by saving the consumer on power bills. Refrigerators can send emails listing grocery item needed, identify necessary servicing and maintenance cycles, sound an alarm when the door is open too long, report food storage dates and spoilage, etc.

This type of futurist thinking can lead to discovering negative possibilities of implementing such technologies. Going back to psychology, if a machine can think, can the machine have a severe psychotic breakdown? Consider the HAL 9000 computer in Stanley Kubrick's 1968 movie *2001: A Space Odyssey*. The HAL 9000 killed all but one astronaut because the computer was worried that the astronauts would discover the secret of their mission. So just imagine a Roomba™ vacuum cleaner attempting to kill the dog for shedding too much hair. The idea that the machines can be hacked and turned into weapons is also a real concern. This brings up concerns as to the security of such systems from either internal or external influences (Dundar et al., 2017; Khorrami et al., 2016).

The trends in CPS autonomous systems will go unabated in the foreseeable future (Yarmoluk, 2017; Gubbi et al., 2013). As mentioned, the concepts and designs of unmanned aerial vehicles is currently being worked on by Amazon™ (Hernandez et al., 2018). In news reports, there is mention of a UAV called the Taco-copter that delivers tacos to the front doorstep. The pizza delivery person may soon be a thing of the past like the elevator operator. Concerning the development and testing of autonomous cars, the research is ongoing, and there are many players with different ideas on how to solve the technical issues. A best guess is that the first production release of a fully autonomous car is still five years away. As for avionic systems, both the commercial and military systems will continue developing better landing systems both in the aircraft and the air traffic control system. Commercial airliners, such as the Boeing 777, already have the capability to take off and land autonomously (that is, if the airport supports the technology to do this). However, the FAA, the regulating authority for flight in the United States, still requires planes taking-off and landing to have an actual human pilot do the work. Still, it is possible and legal elsewhere.

Chapter 3: Methodological Framework

3.1 Modeling Paradigm

Cyber-Physical Embedded Systems are becoming ever more involved in day-to-day activities without much fanfare or notice, in spite of the increase in the interconnection, autonomy, and complexity with decision making capabilities (Lee, 2015). The system will still have the normal constraints of resources (Wehrmeister et al., 2013). Pure embedded systems, by their very nature, are under-resourced. Such things as computational power and memory resources of embedded systems will be limited as compared with non-embedded systems (Drumea and Dobre, 2014) Using designs of parallelism helps with the resource allocation, but it also makes the system more complex in the amount of hardware and software required to operate such a system. The ability to test is hampered by complexities of hidden machine states (Shi et al., 2017; Gobbo and Benini, 2013).

The definition of a decision is that it is an irrevocable allocation of resources (Parnell and West, 2011). Any decision-making system deployed in the cyber-physical context will need to use hardware efficiently, software effectively, and the human-agent both efficiently and effectively to prevent overload demand of the finite resources. Decision making usually is a process of selecting a best choice from a number of different alternatives and options, and it is usually assumed that the decisions are based on rational and reasonable thought for a course of action (Quintas et al., 2017). Even if a choice were irrational and extremely biased, the basis for making the best choices out of subjective or perceived outcome will optimize the decision process as long as the decision holds to the Proposition Two. Decision making reduces the complexity and the uncertainty as a matter of determining a reaction or solution, that in the end, may have multiple objectives. In this thesis, the focus is on a supervisory command and control structure in the decision-making process of the cyber-physical embedded system using human-agents (Xin et al., 2015).

Supervisory command and control structures are usually well organized and complex (National Research Council, 2006). These structures assume a bottom up, feed forward information flow (see Figure 3-1). In supervisory command and control structures, the decision-making usually works extremely well as a means to an end, but at other times, the structure can fall apart and completely collapse on mismatching or contradictory information flowing into the system. For example, typical systems that uses supervisory command and control structures are emergency management services and military command operations. For responders to an emergency, the management of the emergency is essential to coordinate responses across large and different agencies, infrastructures, and network topologies, while optimizing the allocation of resources to their most effective end (Alippi et al., 2017; DHS, 2002).

Decision making applied to cyber-physical embedded system design has a pivotal role in defining and shaping the human-machine interface within the societal norms of different cultures. The machine-agent and human-agent are no longer separable (Backhaus et al., 2013). The application of the decision-making system within the construct of the cyber-physical system impacts the attributes of system such as performance, safety, economics, etc. There is much debate as to whether design principals and guidelines can be well understood or expressed beyond just performance (Alippi et al., 2017). There is also the correctness of the design to consider. When the design is complex, it should require that the engineering and science be well understood, or at least, shown to be both correct and economically beneficial for using the system to increase performance. This increase in performance is both a physical and economical advantage.

3.2 Cyber Decision Modeling

The framework for the decision model comes from John Boyd (b.1927-d.1998) who was a USAF fighter pilot. At the time around the mid-1950's, pilots were mainly taught to navigate the airplane and drop bombs. Boyd took a very different strategy to making air to air combat decisions. He developed a decision-making model for the combat situation called the OODA Loop (Figure 3-1). (OODA - Observer, Orient, Decide, and Act.) The generalized structure of the model uses basic feed-forward and feed-backward loop within the decision-making process (Novak et al., 2017). This type of activity is essentially the cybernetic system described by Nobert Wiener. The principles are being applied here to gain a deeper understanding of the internals of the model. The decision-making principles and guidelines are not that different between machine-agent and human-agent (Wan et al., 2017). The underlying principles and theory of the OODA Loop applies to any reactive system, i.e., CPS. By leveraging the OODA Loop as the definitive decision-making architecture of the cyber-physical embedded system, and because of the generalized form used to describe the systems, the architecture is manageable to the design of intelligent systems.

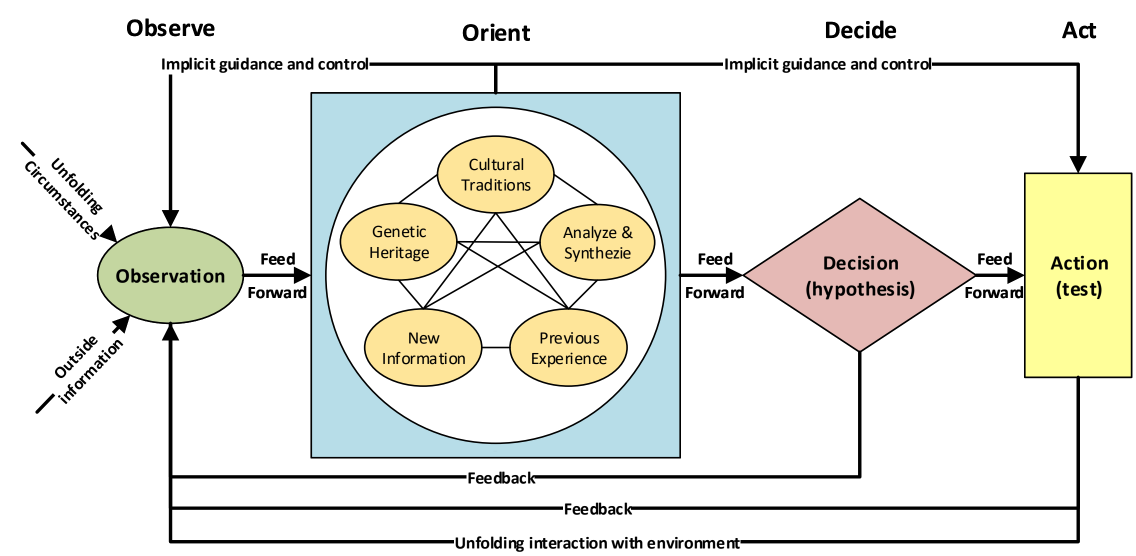


Figure 3-1: OODA Loop: Human-agent (Boyd, 1976)

The OODA Loop describes in simple but meaningful terms the functionality of linear and non-linear systems to external stimulae and internal guidance that changes the behavior of the system. This is an open system model where events are not always either continuous or discrete. The use of the OODA Loop is to form an agent-based model that accounts for both machine and human (Novak et al., 2017). Most of the application of the cyber-physical system would be considered a mixture of continuous and discrete input; and therefore, a hybrid system. The research and application of using the OODA Loop model to assist in the decision-making process (i.e., DSS) is used in many different areas, such as military command operations, emergency response management, business ventures, and now cyber-physical embedded systems (National Research Council, 2006; DHS, 2002). However, at current, the OODA does not appear in the literature for cyber-physical or embedded systems.

In the algorithmic design of the software and firmware for the cyber-physical embedded system, the OODA Loop divides the architecture into four separate entities of phases of the decision-making process: observe, orient, decide, and act. At each of the stages, the time series of both internal and external events must hold the system in balance as it continually operates.

The OODA Loop works as follows:

- Observe: the external world is being detected. Both human-agent and machine are sensing events in the environment.
- Orient: the internalization of the events unfolding in the environment are being computationally digested by both human-agent and machine-agent. New information is being added to the external sense data.
- Decide: this is when a decision is made. The decision can be either to do something or nothing until more information is acquired.
- Act: if in the decision stage, the system should perform the action that is external to it.

The interesting thing to note is the feed-forward and feed-backward at the different stages with guidance as a control measure of the insurance that the system is functioning properly. This idea is further investigated in Chapter 4: The type of AI used in the decision-making process is similar to the backpropagation neural network (BPN). In Chapter 4, the Blackjack Player will demonstrate the power of using such an artificial neural network (ANN) in the CPS design.

The machine-agent uses the OODA loop with the only difference being in the Orient phase of the OODA process. Instead of the using the decision-making biases of "cultural traditions" and "genetic heritage", the machine-agent terms those biases as "processing power" and 'base-type algorithms". The definition of the processing power bias is at the heart of the machine's hardware architecture. It will ultimately determine the throughput of the decision process. It is essentially a bottleneck. The base-type algorithm is the efficiency and effectiveness in the choice of algorithmic design used. This is the software and is also considered a bottleneck.

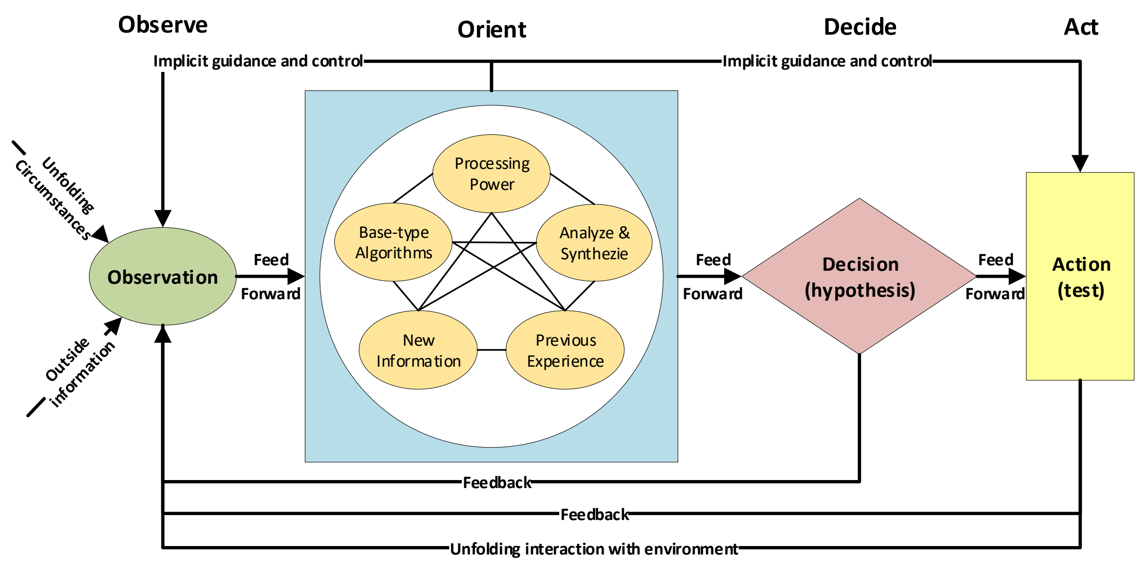


Figure 3-2: OODA Loop: Machine-agent (Trembley et al, 2017)

Since the OODA Loop has a time component, and it can be described as a stochastic system with random events happening in the external environment, it must quickly interpret the external sources of unfolding events and outside information to resolve to an action. However, the bottlenecks that are naturally in the design must be dealt with. Because the bottlenecks constrain the throughput performance of the decision-making, the bottlenecks then become the points of interest in design optimization (Figure 3-3). For example, in cyber-physical embedded systems, bottleneck B1 is the sensor data input. The considerations are for the types of sensors used and for whether the sensors should be of higher or lower level informational value. The bottleneck B2 and B3 are internal to the systems and are driven by the hardware and software algorithms. The bottleneck B4 is the reactive event from the system based on the decision that was made in connection with the sensory inputs from external stimuli.

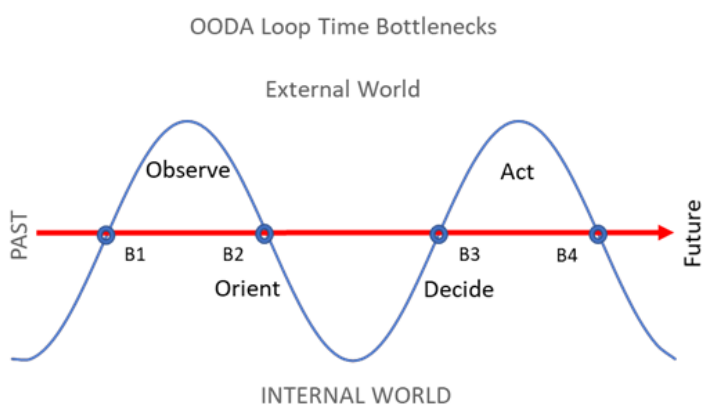


Figure 3-3: OODA Loop Time Bottlenecks (Boyd, 1976)

Using the CAS example from the literature review, in order for the collision avoidance to take place, the sensors on the car must detect (Observe) that an object is within the path of the moving vehicle (Orient). The vehicle's computer will make the determination (Decide) and trigger event that will cause the brakes to be applied or not (Act) (Schnelle et al., 2017; NTSB, 2015).

The case for machine-agents and human-agents have biases, and those biases are based on a set of learned experiences. Whether the human-agent is working, individually or in a group, the predictability of human-agent is less certain than the machine-agent. It is for this reason that the cyber-physical systems designers need to be aware of systems, especially those with high levels of automation, and the responsibility of the role in the system that human and machine advocate (Wang, 2016; Chen et al., 2011). This is undeniably the engineer's responsibility when embedding autonomous features into a system (Shneiderman, 2007).

3.3 Cyber-Physical Embedded System Automation Design

This thesis considers combining several concepts of Machine Intelligence: Deep Learning, Bayesian Reasoning, and Fuzzy Logic (Figure 3-4). By combining these methods, a number of possibilities emerge to design sophisticated cyber-physical intelligent machines and systems. By using softcomputing computational techniques on cyber-physical embedded systems, considering the computation processing power of small low-end embedded processor devices, there is current fascination in the industrial and commercial development and implementation of off-the-shelf embedded systems, i.e., IoT, that integrates into a wider fabric of available technology, such as WIFI, Internet, Ethernet, etc. The cyber-physical systems go from structured finite state machines (FSM) to hybrid combinations of different technologies (Shi et al., 2017; Zhang et al., 2016).

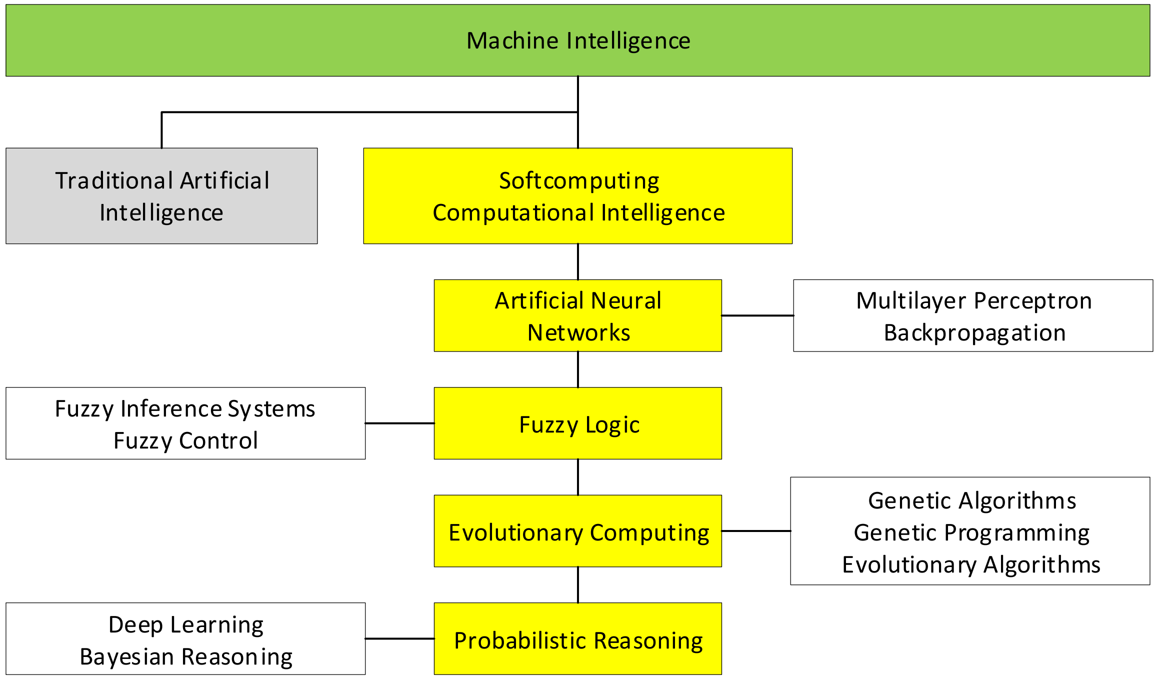


Figure 3-4: Machine Intelligence Tree Diagram (Lewis, 2015)

When considering implementing embedded system design into a cyber-physical configuration, the design becomes intelligent by adding autonomous computational control to the machine. The example in the embedded processes within the CPS is a manual coffee maker that is interfaced with a microcontroller to set temperatures, times, and brew preferences. The microcontroller can also connect to the Internet and be worked remotely from a distance. The coffee maker can essentially be controlled from anywhere there is an Internet connection. The coffer maker is now IoT capable and forms the cyber-physical design of a CPS. Features like automatic timers and self-cleaning modes could be implemented based on the needs of the system, and those features are embedded. The IoT coffee maker, by being accessible remotely, does not impede with its prime purpose (i.e., to make coffee).

3.4 Cyber-Physical System and AI

Artificial Intelligence and the ideas of neural networks can be traced back to the period just after World War II. The idea of Artificial Intelligence is credited to have been invented by the Dartmouth conferences of 1957. The beginning of the digital computer age was generating much excitement amongst mathematicians who followed the school of thought of the Logic Theorist (Goldstein, 2002). Herb Simon and Alan Newell from the Carnegie Mellon Institute had spearheaded the idea that if computers could solve complex logic problems in novel ways, the “thinking machine” could be designed. Other notables who were around at the time were John von Neumann and Norbert Weiner. Von Neuman was very instrumental in the early hardware architecture of computer systems, and Weiner spearheaded the idea of feedback systems in manmade and natural systems (Weiner, 1964). The Boyd OODA Loop is essentially a feedback and feedforward system that has many parallels to Weiner’s Control Theory and was developed in the same time period.

Viewing the configuration of the AI Venn Diagram (Figure 3-5), this thesis considers the cyber-physical embedded system as an architecture that follows the principles from the deepest attribute, i.e., Deep Learning, to that which culminates in a system that is artificially intelligent. The thesis demonstrates the choice of correct or best methods when designing such a system, a system where correctness and performance are verified and validated.

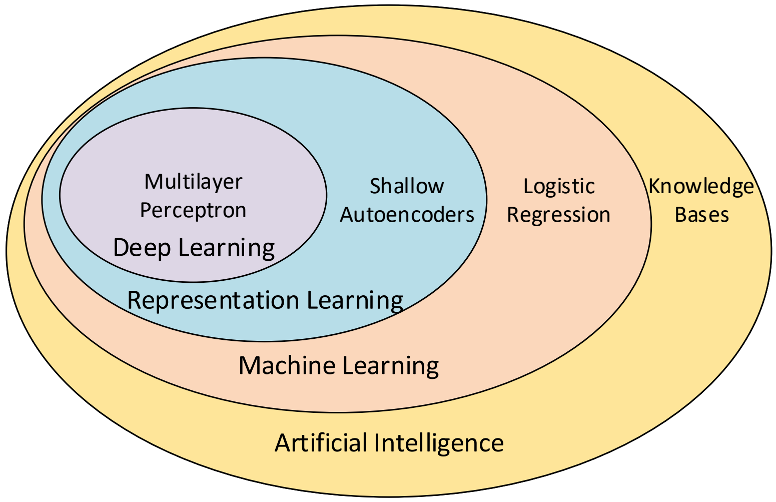


Figure 3-5: AI Venn Diagram (Goodfellow et al, 2015)

3.5 Backpropagation Neural Network

In 1985, the Backpropagation Neural Network (BPN) performed the iris classification problem with great success, which led to a more optimistic view to neural networks. The philosophical arguments that thinking machines were impossible, and that the science of neural networks would not amount to much, rapidly vanished. Artificial Neural Networks (ANN) are now commonly found in in everything from cars to smartphones. Thus, the BPN, using a probabilistic non-discrete method for feature detecting, works. The science of why it works is described in the Blackjack Player, Section 4.3.2.

Essentially, in a BPN, there are a number of hidden layers that combine and assist in the decision-making of pattern classification. These hidden layers are the black box of the system. Since the BPN is very generalized and uses a supervisory level of training, i.e., learning, to a wide range of classification and optimization problems, by performing designed experiments using rigorous verification and validation methods, the BPN can be tuned to find the optimal or best “fit” of the “things” classified (Thalassinakis et al., 2006).

3.6 Fuzzy Inference and Fuzzy Logic

The historical record shows evidence that mathematicians and physicists were developing partial models and discussing 'vague' and 'fuzzy' set theory in the early days of the 20^{th} century (Garrido, 2012). Among these people were Bertrand Russell (b. 1872 - d. 1970). Bertrand Russell who attempted to solve the 23 Hilbert problems during the entire first half of the 20^{th} century, as well as constructing an elegant layout of mathematics in his three volume set Principia Mathematica, had great impact on twentieth century thinking dealing with unknown states and conditions by finding good answers to very hard problems. However, the credit for the invention of fuzzy systems goes to Lotfi A. Zadeh (b. 1921). Zadeh's intention was to create a formalized system to handle more efficiently the imprecision of discrete reasoning. The use of fuzzy logic within an embedded system to solve for probabilistic non-discrete data in a stepwise method for decision-making (Zhang et al., 2017). Why a method like this would work is described in the Blackjack Player experiment (Section 4.3.2).

3.7 Logistic Regression

Logistic regression is also an exceptionally powerful tool when the independent variables are a mixture of categorical and continuous variables. Logistic regression extends the ability to collect data of metric and non-metric types that are usable and mathematically justifiable. Since logistic regression makes no multivariate normality assumptions of distribution, such as a discriminant analysis, logistic regression breaks with methods that impinge on the requirement of normality or a standardization of the data set being used (Xu, 2016).

The method of logistic regression as an *a priori* and *a posterior* predictive technique is a powerful tool. By considering and using simple Bayesian statistics of calculating odds, the generalized logistic regression technique can be used in many problems of classification. A defined system using logistic regression can, in the face of imperfect information, recognize through classification, and make a determination to the best or optimal path forward, until the next decision is required. The logistic regression multivariate analysis brings with it the assumption of accuracy to classification objects and real-time associations to the classifications, i.e., immediate decision-making data for action (Frankot, 2012).

3.8 Levels of Automation

The term "Levels of Automation" is a description of the quantity of interaction between human-agent and machine. Figure 3-6 shows the four stages of information processing (Sheridan, 2011; Parasuraman et al., 2000; Wilkins, 2003). It is essentially a scaled down version of the OODA Loop. The beginning stage is the sensory input and processing. The second stage is where the sensory input data from the first stage is processed. The third stage is the decision-making process, and the fourth stage is the response. This is only looking at the process on a higher level, in that the Sheridan and Boyd model of information processing and decision making are similar and symmetrical. Sheridan uses a simple description of receiving input from the environment in a one-way direction using the biases associated with memory and experience to arrive at a decision to take an action or not. The Boyd model coincides with the control theory paradigms of feedforward and feedback loops; whereas, Sheridan uses a feedforward only model. It is by using the Sheridan model that the level of automation is considered in the cyber-physical system.

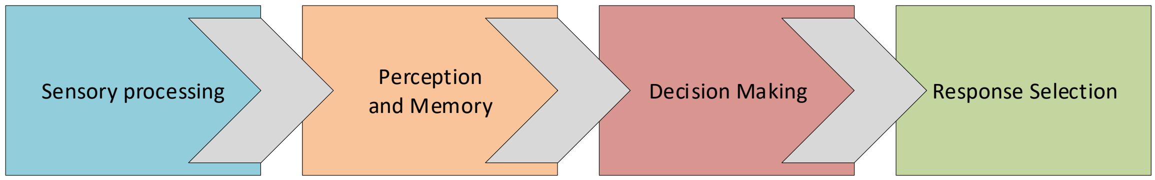


Figure 3-6: Four-stage Model of Information Processing (Sheridan, 2011)

Using Sheridan's Levels of Automation, (see Table 3-1), the refinement of dividing automation into ten levels is best when working with a system that is being designed or implemented with enhanced autonomous features that are more generalized (Wilkins, 2003). The idea is to match the expectation of the human-agent with that of the machine-agent. The machine-agent is what provides the automation. The levels provide a discrete continuum of ratios of human-agent/machine-agent systematic automation, from Level 1, where the human-agent is fully in control and the machine-agent offers no assistance, to Level 10, where the machine-agent is fully in control and ignores the any input from the human-agent. In the intermediary levels from Level 2 to Level 9, the human-agent and machine-agent basically compromise as to how the system will work under operational conditions. The roles and responsibilities need defining based on the level of automation in the design (Vahidi and Eskandarian, 2003; Noh and An, 2018; European Parliamentary Research Service, 2016). For example, the CAS is a Level 10 system. Although the CAS is not in control of the car per se, it could apply the brakes at the time when a braking action is detected. The human-agent would not be warned and may even be surprised by the action.

Table 3-1: Sheridan Level of Automation (Sheridan, 2002)

LEVELS OF AUTOMATION OF DECISION AND ACTION SELECTION		
HIGH	10	The computer decides everything, acts autonomously, ignoring the human,
	9	informs the human only if it, the computer, decides to.
	8	Informs the human only if asked, or
	7	executes automatically, then necessarily informs the human, and
	6	allows the human a restricted time to veto before automatic execution, or
	5	executes the suggestion if the human approves, or
	4	suggests one alternative.
	3	Narrows the selection down to a few, or
	2	the computer offers a complete set of decision/action alternatives, or
LOW	1	the computer offers no assistance: the human must make all decisions and actions

Driverless or autonomous vehicles (e.g., cars, airplanes, ships, etc.) are at the forefront of the cyber-physical embedded systems technologies, and the latest trend in engineering autonomous systems (Wehrmeister et al., 2013). Both the Society of Automobile Engineers (SAE) and the European Parliamentary Research Services (EPRS) use six levels of automation to describe levels of automation for cars (SAE INTERNATIONAL, 2014; European Parliamentary Research Service, 2016). The SAE provides comprehensive and complicated definitions at each level, providing designers of autonomous cars the specification to the requirement that needs to be met in order to be certified. The ultimate certification of autonomous vehicles in United States will come from the National Transportation Safety Board (NTSB) (NTSB, 2015).

In Table 3-2, the SAE Level 0 is like Sheridan's definition for Level 1. The human is fully in control of the vehicle and is offered no assistance by the onboard computer systems; however, the SAE definition allows the human-agent to be warned. In accordance with the SAE definition of Level 1 and Level 2, the driver is still fully in control of the vehicle, but now there is an embedded automation system, i.e., machine-agent, starting to take control of some features of the car in motion. The Automated Driving Systems (ADS) that is helping the driver maintain coordination with the Dynamic Driving Task (DDT) works in cooperation with the human-agent. An example of this would be the lane departure warning system. At Level 5, is the ADS is fully performing the DDT.

Table 3-2: SAE Levels of Automation (SAE, 2014)

Level of Driving Automation	Role of User	Role of Driving Automation System
DRIVER PERFORMS THE DYNAMIC DRIVING TASK (DDT)		
Level 0 - No Driving Automation	Driver (at all times): • Performs the entire DDT	Driving Automation System (if any): • Does not perform any part of the DDT on a sustained basis (although other vehicle systems may provide warnings or support, such as momentary emergency intervention)
Level 1 - Driver Assistance	Driver (at all times): • Performs the remainder of the DDT not performed by the driving automation system • Supervises the driving automation system and intervenes as necessary to maintain safe operation of the vehicle • Determines whether/when engagement or disengagement of the driving automation system is appropriate • Immediately performs the entire DDT whenever required or desired	Driving Automation System (while engaged): • Performs part of the DDT by executing either the longitudinal or the lateral vehicle motion control subtask • Disengages immediately upon driver request
Level 2 - Partial Driving Automation	Driver (at all times): • Performs the remainder of the DDT not performed by the driving automation system • Supervises the driving automation system and intervenes as necessary to maintain safe operation of the vehicle • Determines whether/when engagement and disengagement of the driving automation system is appropriate • Immediately performs the entire DDT whenever required or desired	Driving Automation System (while engaged): • Performs part of the DDT by executing both the lateral and the longitudinal vehicle motion control subtasks • Disengages immediately upon driver request
AUTOMATED DRIVING SYSTEM (ADS) PERFORMS THE ENTIRE DYNAMIC DRIVING TASK (DDT)		

Level of Driving Automation	Role of User	Role of Driving Automation System
Level 3 – Conditional Driving Automation	Driver (while the ADS is not engaged): • Verifies operational readiness of the ADS-equipped vehicle • Determines when engagement of ADS is appropriate • Becomes the DDT fallback-ready user when the ADS is engaged DDT fallback-ready user (while the ADS is engaged): • Is receptive to a request to intervene and responds by performing DDT fallback in a timely manner • Is receptive to DDT performance-relevant system failures in vehicle systems and, upon occurrence, performs DDT fallback in a timely manner • Determines whether and how to achieve a minimal risk condition • Becomes the driver upon requesting disengagement of the ADS	ADS (while not engaged): • Permits engagement only within its ODD ADS (while engaged): • Performs the entire DDT • Determines whether ODD limits are about to be exceeded and, if so, issues a timely request to intervene to the DDT fallback-ready user • Determines whether there is a DDT performance-relevant system failure of the ADS and, if so, issues a timely request to intervene to the DDT fallback-ready user • Disengages an appropriate time after issuing a request to intervene • Disengages immediately upon driver request
Level 4 - High Driving Automation	Driver/dispatcher (while the ADS is not engaged): • Verifies operational readiness of the ADS-equipped vehicle • Determines whether to engage the ADS • Becomes a passenger when the ADS is engaged only if physically present in the vehicle Passenger/dispatcher (while the ADS is engaged): • Need not perform the DDT or DDT fallback • Need not determine whether and how to achieve a minimal risk condition ADS (while not engaged): • Permits engagement only within its ODD ADS (while engaged): • Performs the entire DDT • May issue a timely request to intervene • Performs DDT fallback and transitions automatically to a minimal risk condition when: • May perform the DDT fallback following a request to intervene • May request that the ADS disengage and may achieve a minimal risk condition after it is disengaged • May become the driver after a requested disengagement	ADS (while not engaged): • Permits engagement only within its ODD ADS (while engaged): • Performs the entire DDT • May issue a timely request to intervene • Performs DDT fallback and transitions automatically to a minimal risk condition when: • A DDT performance-relevant system failure occurs or • A user does not respond to a request to intervene or • A user requests that it achieve a minimal risk condition • Disengages, if appropriate, only after: • It achieves a minimal risk condition or • A driver is performing the DDT • May delay user-requested disengagement

Level of Driving Automation	Role of User	Role of Driving Automation System
Level 5 - Full Driving Automation	Driver/dispatcher (while the ADS is not engaged): • Verifies operational readiness of the ADS-equipped vehicle • Determines whether to engage the ADS • Becomes a passenger when the ADS is engaged only if physically present in the vehicle Passenger/dispatcher (while the ADS is engaged): • Need not perform the DDT or DDT fallback • Need not determine whether and how to achieve a minimal risk condition • May perform the DDT fallback following a request to intervene • May request that the ADS disengage and may achieve a minimal risk condition after it is disengaged • May become the driver after a requested disengagement	ADS (while not engaged): • Permits engagement of the ADS under all driver-manageable on-road conditions ADS (while engaged): • Performs the entire DDT • Performs DDT fallback and transitions automatically to a minimal risk condition when: • A DDT performance-relevant system failure occurs or • A user does not respond to a request to intervene or • A user requests that it achieve a minimal risk condition • Disengages, if appropriate, only after: • It achieves a minimal risk condition or • A driver is performing the DDT • May delay a user-requested disengagement

Table 3-3 is the European Parliamentary Research Services (EPRS) definitions of Levels of Automation, which are directly borrowed from the SAE definitions, but provide an easier explanation as to driving mode, human-agent, and car (machine-agent) (European Parliamentary Research Service, 2016). In both tables, it is interesting that there is a hard separation between human-agent and machine in performance of the driving task, the human-agent becomes unknowingly disengaged after Level 2 and is almost knowingly disengaged at Level 4.

Table 3-3: UE Levels of Automation (EPRS, 2016)

	SAE	Name	Steering, acceleration, deceleration	Monitor driving environment	Fallback performance of dynamic driving task	System capability (driving modes)
Human Monitors Environment		**No Automation**				
	0	the full-time performance by the human driver of all aspects of the dynamic driving task, even when enhanced by warning or intervention systems.	Human	Human	Human	N/A
		Driver Assitance				
	1	the driving mode-specific execution be a driver assistance system of either steering or acceleration/deceleration using information about the drivng environment and with the expection that the human driver perfom all remaining task.	Car	Human	Human	Some driving modes
		Partial Automation				
	2	the driving mode-specifc execution by one or more driver assistance systems of both steering and acceleration/deceleration using information about the driving environment and with the expection that the human driver perfom all remaining aspects of the dynamic driving task.	Car	Car	Human	Some driving modes
Car Monitors Environment		**Conditional Automation**				
	3	the driving mode-specific performance by an automated driving system of all aspects of the dynamic driving task with the expectation that the human driver will respond appropriately to a request to intervene.	Car	Car	Human	Some driving modes
		High Automation				
	4	the driving mode-specific performance by an automated driving system of all aspects of the dynamic driving task, even if a human driver does not respond approporiately to a request to intervene.	Car	Car	Car	Some driving modes
		Full Automation				
	5	the full-time performance by an automated driving system of all aspects of the dynamic driving task under all roadway and environmental conditions that can be managed by a human driver.	Car	Car	Car	All driving modes

Using levels of automation to design a model of an embedded system should be relatively straightforward. If working with a generalized design, Sheridan's definition should be used as the guideline. At Sheridan Level 1, the human-agent does not need or require a machine-agent for assistance in any task. The human-agent acts as the autonomous control to the machine that is operating without autonomy. However, the design of the cyber-physical embedded system should discretely define the level of automation as the human-agent would need this knowledge to be able to work with machine-agent. The roles and responsibilities of each should be well established. The importance is that the design is implemented correctly and that it performs as intended (Backhaus et al., 2013; Vahidi and Eskandarian, 2003; European Parliamentary Research Service, 2016; SAE INTERNATIONAL, 2014).

Billings detailed and argued about of a human-centered approach to autonomy (Billings, 1997). There are six important concepts that engineers should remember and internalize that provide guidance in automated designs:

1. Automation systems should be comprehensible.
2. Automation should ensure operators are not removed from the command role.
3. Automation should support situation awareness.
4. Automation should never perform or fail silently.
5. Management automation should improve system management.
6. Designers must assume that operators will become reliant on reliable automation.

This is the second point of the thesis: the transient supervisory command and control features of many autonomous systems (Tweedale and Jain, 2011; James, 1953). For example, if the machine-agent has detected a problem, i.e., fault, and downgrades the level of automation, the human-agent has to be the one to react (Marquez and Ramirez, 2014). This reaction in many cases can have fatal consequences, and this needs to be well understood. It will be shown that modeling, simulating, and prototyping are the best tools in the engineer's kit (Novak et al., 2017; Arne, 2000). The case of Asiana Airline Flight 214 demonstrates the inherent problems with autonomy. Even if the autonomy is well thought out, proved using the most rigorous of testing measures, with the human-agent involve, it does not always work as expected (NTSB, 2014).

3.9 Combining Design Attributes for Autonomy

In theory, the ability to design "good" cyber-physical embedded systems by using machine intelligence that combines with human agency is directly associated with the level of automation the system is required to achieve. The cyber-physical embedded system's level of automation requirement allows, in the conceptual and preliminary design phase, the identification of risks to cost, schedule, and quality. The greater the level of automation of a system implies greater cost, whereas with less automation, the associated costs are spread over the operational lifetime phase of the system. To strike a balance with a system design leveraging automation, the definitions of the external stimulus on input effects to the internal structures of the machine-agent must be of prime consideration (Bosetti et al., 2015). The external output actions produced by the decision processes need only be correct. That is, the internal states of the machine are essentially treated as a black box and are tested based on a stimulus-response model (Novak et al., 2017; Xin, et al., 2015). The choice of hardware (processors, memory, sensors, etc.) and software (programming language, compiler, algorithms, etc.) will have the ultimate ramifications to the limits that the system can achieve and how it can be verified and validated through testing (Xin et al., 2015; Moradi-Pari et al., 2014).

3.10 Summary

The successful integration of the human-agent with a machine agent requires a well-understood balance between the roles and responsibilities in the supervisory command and control decision making structure. The decision modeling of the cyber-physical system, using the OODA loop decision making model that architects the isomorphic relation between human and machine as similar but separate agencies, must be merged to assume the broad, human-agent, and narrow (machine-agent) intelligence attributes of the system. In this way, the modeling of cyber-physical systems becomes less complicated and more tractable because of combinations of linear and non-linear transitions modes within the hidden variables of the CPS. Whereas the artificial intelligence of the CPS is narrowly focused on the designed task, the human-agent is broadly focused on responsibilities that include any number of unknowns from the environment sensor systems in which the machine-agent is not proficient in determining. The inherent problem is that while operating, the system as a function of its level of automation can transition to a lower level of automation. The state change of the system releases back more control to the human-agent. This is potentially a dangerous time where quick thinking and reaction is needed.

The reverse is also possible when the AI of the CPS detects an imminent condition, for which it is trained, and reacts by transitioning to a higher level of automation; thereby wresting control away from the human-agent. It is in the creation of attempting to develop structured models for the CPS that the meta-model paradigm emerges. This helps create a more simplified model of the system that allows for the testing of the sensory inputs and actuator outputs of the system. Again, the meta-model is a “black box” idea of the system but at a hierarchical level of abstraction whereby system performance and correctness are determined. The verification and validation of the system can be certified for critical system usage (Karsai, 2003).

Chapter 4: Experimental Frameworks Using Modeling, Simulation, and Prototypes

4.1 Modeling an Experimental Framework

By using various commercial off the shelf tools, such as LabView, MATLAB, SolidWorks, CATIA, Simulink, etc., modeling, simulating, and prototyping a system becomes a very practical engineering and development experience. The ability to rapidly develop and deploy models, as well as automatically generate software and firmware source code and test branches or perform thermal and factual structural analysis, becomes a spring board to getting a design finished quicker and with less problems. However, software and firmware bugs still get inserted at all levels of the design based on the initial assumptions, or not well understood requirements of implementation. So, how does the design need to be tested in order to prove that it has met the requirements for performance and correctness?

The framework for simulating a particular model relies on the granularity to variables that dominates the optimizations of the real system (Kang et al., 2018; Corno et al., 2016). The key to effective simulation is not to bog down the model in minutiae using variables that have little or no relevance or influence to the control of the system's behavior. Experience can find many of the contributing variables, but if the model is built to the generalized specifications of the system, then those variables should become identified through running simulation scenarios (Corno et al., 2016).

Information is useful when it becomes knowledge and the knowledge allows for decisions and actions to be based on the ideal workings of the system (see Figure 4-1). This is the decision-making properties to OODA Loop in a feedforward design. The information is gathered and sorted from external stimulae (i.e., sensor recordings) to refine the data into meaningful sets that should identify the need to action and to what action that should be (Quintas et al., Menezes and Dias, 2017; Parasuraman et al., 2000; Sheridan, 2011).

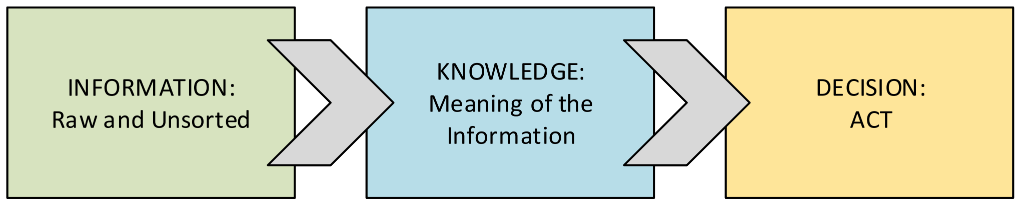


Figure 4-1: Basic Data Flow (Parasuraman et al., 2000)

Since it is recommended that many different simulation scenarios be run to test and experiment the model, the model will need to clarify the nature and define the number of variables for each scenario. Managing the model is critical for accuracy and validity in the experimental scenarios. The correct level of understanding of the system's functionality will allow for an ascertainable level of predictability using the models (Novak et al., 2017). Since there are hidden variables within the internal structure of the cyber-physical embedded system, the designer should choose designs where information lost in critical areas of the internal structures are at a minimum. Figure 4-1 depicts the three high level structures of the system informational data flow. The areas of most concern, since each has internal structures that remain hidden, is the software-in-the-loop and the hardware-in-the-loop. The human-in-the-loop is assumed to have the cognitive ability to coexist within the cyber-physical system, i.e., natural intelligence (Ruff et al., 2002; Chen et al., 2011). The software-in-the-loop has the multitasking control of the hardware-in-the-loop by using intelligent machine algorithms to decide and act based on the structure of its inference engine (see Figure 4-2). The software-in-the-loop and hardware-in-the-loop are inextricably tied together to perform the machine information processing and are the embedded portion of the CPS (Sheridan, 2011).

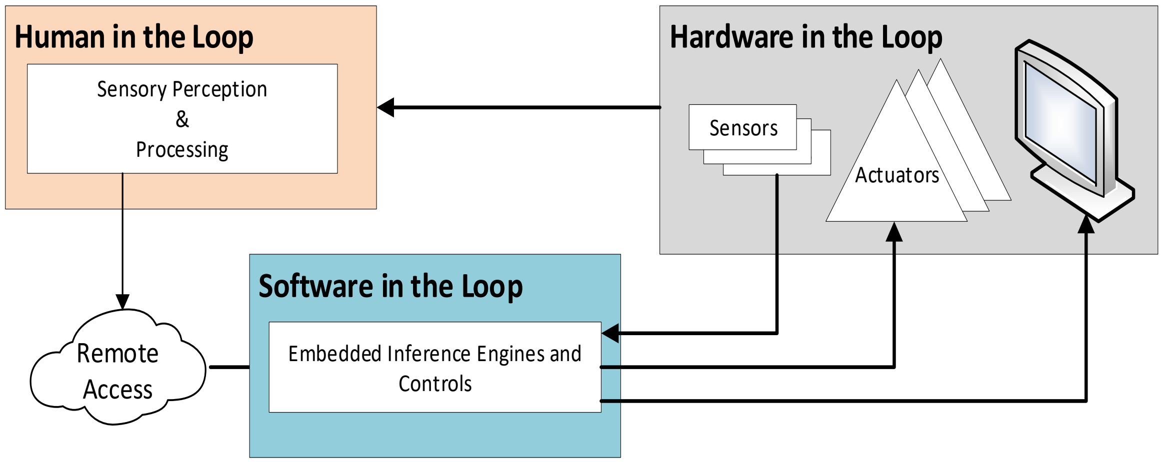


Figure 4-2: Human, Software, Hardware Loop Architecture (Trembley, 2015)

Many test methodologies use the Black Box theory of operation. By controlling the inputs to the system, the outputs can be observed in temporal relation to the inputs. Because some systems can be peered into, the white box methodology of testing could also be used in conjunction with these designed models (Novak et al., 2017). The measurement of performance of the system is the amount of information that can be processed into useful knowledge, allowing the cyber-physical embedded system to decide and produces actions that are beneficial. The word beneficial is used in conjunction with Proposition 2 because a system not reacting in time because of information overload or starvation, i.e., lack of performance, is a measure of the correctness of the system.

The prediction of problems becomes the next hurdle in the model and simulation. Once demonstrated that the model is valid, the simulation should be able to demonstrate system behavior. Simulation can begin (Frankot, 2012). The problem in defining the simulation scenarios is whether the system's behavior, especially in large complex systems, needs to allow coherency to a predictable level of possible and sometimes even plausible outcomes. A scenario would be to refine the model to provide an idea of the stimulus for the decision-making process to either the human-agent or the machine-agent (Jin et al., 2018). This is the criticality of the model and simulation. A scenario of having the automation mode shift to a different level need to be clearly stated or defined as a recovery action, i.e., the machine takes more control or the human takes more control. The model then assumes a method of the intention of the design to leverage the level of automation in any transient event that correlates to upshifting or downshifting in automation. Since the software-in-the-loop uses softcomputing nondeterministic states, the awareness to the adaptive and emergent behaviors of radical, or events leading too quickly shifting, must be adjusted to fit the criticality of the system (Parasuraman et al., 2000).

4.2 Cyber-Physical System Test Architecture

The experimental design would use a recursive test architecture (see Figure 4-3). The test architecture uses the different scenarios to generate and drive the system where the output or behavior of the system can be monitored and reported (Ohta et al., 2017). The test system architecture has the capabilities to allow access to the software and hardware running in real-time. This is accomplished by means of a debugger, JTAG, or Boundary Scan device.

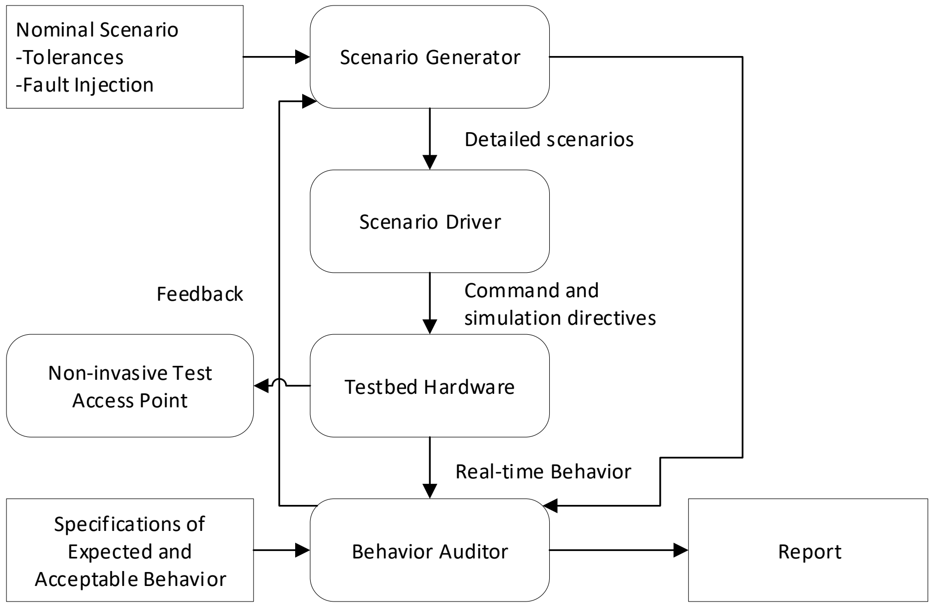


Figure 4-3: Test System Architecture (Trembley, 2015)

4.3 Applied Experiment of a Cyber-Physical System

4.3.1 Simulated Moon landing

The moon landing has been a classic programming exercise for students in the computer sciences since the late 1960's (Martin et al., 2014). The design and requirement of the program is simple, but the exercise is a NP-Hard type problem where optimization is the difference between life and death. The scenario is this:

The astronaut is in a moon lander and starts a descent at 50,000 feet above the surface of the moon. The mass and weight of the moon lander is considered all the way to the surface of the moon. That is, as the astronaut burns fuel to slow the descent, the moon lander gets lighter; The basic physics of the surface approach is the same as a feather being dropped from the same height since there is no atmospheric resistance. The bonus is that less energy is needed to slow the moon lander as it becomes lighter. From the start of the program, the moon lander descends to the surface of the moon commensurate with the gravitational tug of the moon, which is 1.62 m/s^2.

The goal is for the astronaut to land on the moon surface at speeds not exceeding 5 miles per hour. In order to achieve a safe landing, the astronaut fires the moon landers thrusters to slow the craft. If the fuel is used prior to reaching the surface, the craft plummets from that height, and accelerates at 1.62 m/s^2 to its total destruction on the surface of the moon, where unfortunately, the astronaut is killed. The optimal speeds of the descent rate at different altitudes to achieve a good landing are not linear. The moon lander cockpit controls are written with National Instruments LabView (see Figure 4-4). The virtual display has different interfaces to indicate fuel levels, altitude, and speed. The experiment also brings in the idea from Marquez and Ramirez experiments where the surface of the moon comes into play (Marquez and Ramirez, 2014). Usually, a type of RADAR system is used to detect the surface directly beneath the moon lander (see Figure 4-5). The RADAR data digitally texturizes the surface so the heights and widths of the objects beneath the moon lander are known with precision. However, live video feeds to the cockpit are also available. For example, the video information of the moon's surface starts with very low-resolution video representing a distance to the surface (see Figure 4-6), and when the moon lander is 100 feet from the surface, the detail become high resolution, and complicated, fraught with danger (see Figure 4-7).

Figure 4-4: Moon Lander Cockpit LabView (Trembley, 2018)

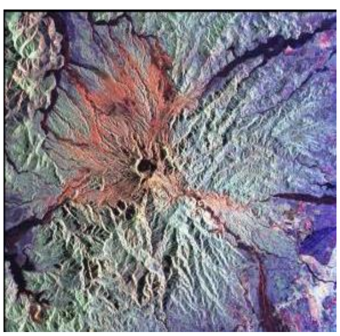


Figure 4-5: RADAR Image (NASA, 2009)

Figure 4-6: Video Moon Surface Distant (NASA, 1972)

Figure 4-7: Video Moon Surface Near (NASA, 1972)

The controls that the astronaut, or human-agent uses consist of a joystick (see Figure 4-8) and throttle control (Figure 4-9). The joystick is used to laterally move the moon lander. The throttle is used to control the descent of the moon lander.

Figure 4-8: Saitek X52 Joystick (Trembley, 2018)

Figure 4-9: Saitek X52 Throttle Control (Trembley, 2018)

There are three different experiments that should be performed using the SAE Levels of Automation (Table 3-2) (SAE INTERNATIONAL, 2014):

1. The astronaut is fully in supervisory control of the craft and has only the basic control indicators. There are no warnings, alarms, or indication as to whether the moon lander will make it to the surface or not. This is Level 0 autonomy.
2. The astronaut has warning systems indicating that the descent speed is too slow or too fast. This is either Level 2 or Level 3 autonomy.
3. The astronaut has a warning system and a collision avoidance system using an embedded process that will take control of the moon lander and attempt to successfully land. This is a Level 4 or 5 autonomy.

The results of the experiment are the success rates for landing safely on the moon. This also tests the softcomputing algorithms and does a baseline comparison between how well the autonomous system did against full human agency. This benchmarks the software design for correctness to the performance of the experiment (Wilkins, 2003).

4.3.2 Blackjack with the BeagleBone™

The Blackjack Player uses a Model-Based Design (MDB) method to document the design process. This allows a concurrent engineering process that can help resolve issues much quicker, especially in the early design phase where assumptions are necessarily revised. The Blackjack Player is modeled and simulated prior to prototyping. The requirements of the design are the definitions of the inputs and outputs, and what qualifies the Blackjack Player as being any good. This is an experiment to test the verification and validation methods from design through development of cyber-physical systems with hidden variables (Shi et al., 2017).

First, the initial model must show a simplified top-level block diagram of the individual components that make the system (see Figure 4-10). These three components are further refined to the individual processes that will be developed and tested individually: visual input system, neural network, and fuzzy inference system (Xu, 2016; Xin et al., 2015). Once the individual subset components are tested and verified, the integration of the subsets are combined to supersets and more verification testing is required. These components form the full software design that will make the Blackjack Player autonomously function.

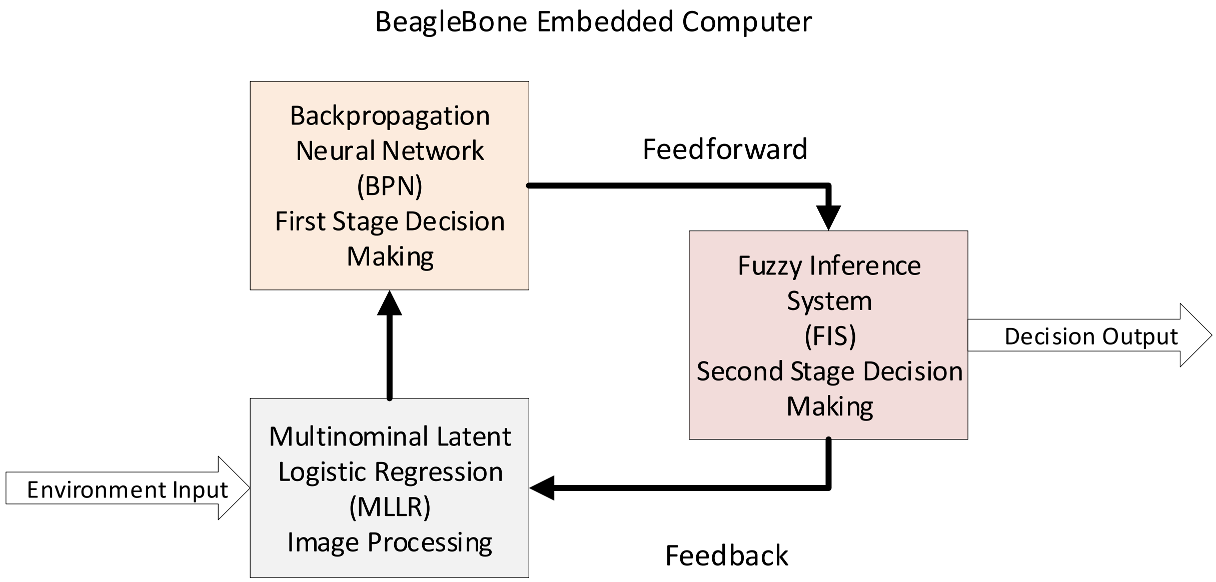


Figure 4-10: Model Based Design Blackjack Player (Trembley, 2016)

The following individual activities surround each component:

- Model-in-the-loop (MIL)
- Software-in-the-loop simulations (SIL)
- Hardware-in-the-loop (HIL)
- Real-time simulations, targeting, verification and validation, design of experiments,
- Further model refinement

All the above listed items are required to ensure the correctness of the designed autonomous system, i.e., the CPS, and performance that should allow the Blackjack Player to make money (Dundar et al., 2017; Backhaus et al., 2013).

The BeagleBone Black microcontroller board was used in this research. The Beagle Bone consists of a fully capable general control input/output (I/O) using an ARM[1] Cortex-A8 processor clocked at 1GHz with 512 Mb of DDR memory.[2] Figure 4-11 shows the BeagleBone with the CCD camera attached. The fast processing speeds and considerable memory are key attributes to the system's

[1] The ARM Cortex-A8 is a processor that supports mobile and embedded designs. ARM processors are mainly used in today's smartphones.
[2] DDR memory stands for "double data rate synchronous dynamic random-access memory.

overall performance, especially for the image process system (see Figure 4-10). Other notable characteristics of the BeagleBone is that it runs on Opensource software using a scaled down embedded Linux operating system. This provides a platform that is friendly to the hobbyist as well as the serious engineer. Because of its low cost, and its technical performance measures, it was chosen over its competitor the Raspberry Pi.

Figure 4-11: BeagleBone with Camera (Trembley, 2017)

The visual recognition system incorporates the most difficult requirements to achieve. Object recognition is a very difficult problem to solve. The understanding of an object is complex subject. Visual recognition systems often employ both cognitive psychology and neuro-science techniques to model and build the technology that mimics human visualization. If such a system is to be developed, it must know the concept of "object" and the 3-D environment in which it resides. All the other objects and conditions from the environment must be distinguished and dealt with in order to concentrate on the object of interest (Gao et al., 2016).

An example of this complexity is a visual recognition system attempting to detect chairs in a room. The simple object known as "chair" becomes vastly complicated when the many different kinds of chairs are considered: armchairs, Adirondack chairs, bikini chairs, chaise longues, etc. And what about stools and benches? Are they considered in the category of chairs? Usually, objects can be broken down into a taxonomy of objects that have strong relational characteristics. The

Multinominal Latent Logistic Regression (MLLR) is used in this way with the visual recognition component (Xu, 2016).

In order for a machine learning system to image process, it must be able to model the variability within the environment and the objects within it. The images contain data in a grid, and variability in the environment would include such things as illumination, shadow, contrast, object angular position, motion, and shape parameters. These parameters need to be accounted for by the Blackjack Player. To simplify the system, the Blackjack Player only has to recognize a standard 52-card playing decks (see Figure 4-12) from an orthogonal angle to the playing card. Illumination, shadow, contrast, object angular position, motion, and shape parameters will all be optimized for the benefit of the Blackjack Player's performance. However, later models that get closer to real world scenarios must account for these environmental factors.

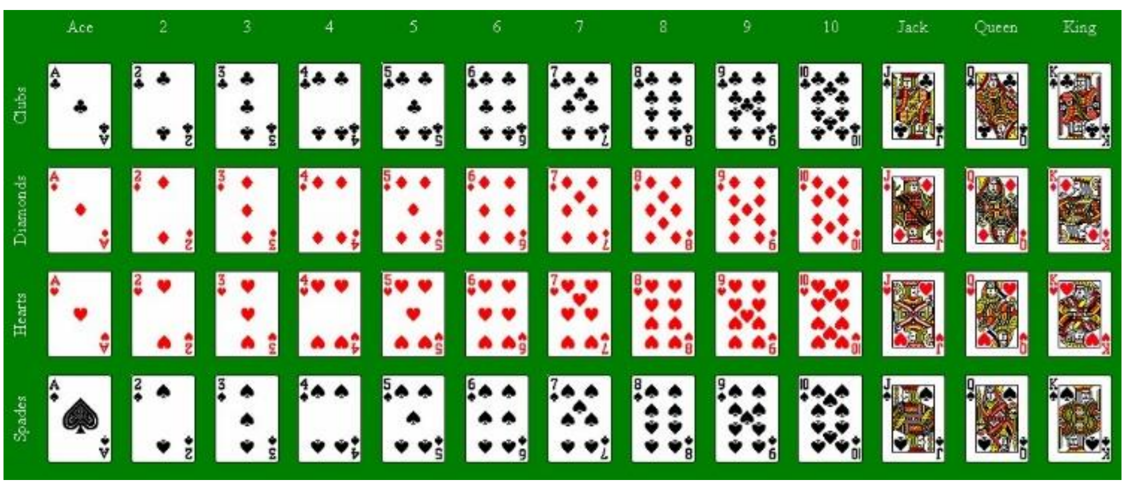


Figure 4-12: Standard 52-card Deck (Shutterstock, 2018)

The camera is a 5-megapixels digital camera with a frame rate of 60Hz (see Figure 4-13). This should provide adequate resolution of the cards and real-time video feed as the game of Blackjack is played. Using a minimum resolution while still being able to recognize the cards in play is key to system performance. This is the first optimization area that needs to be considered. The camera will be capturing one frame of 640 x 480 picture data at a rate of 60Hz.

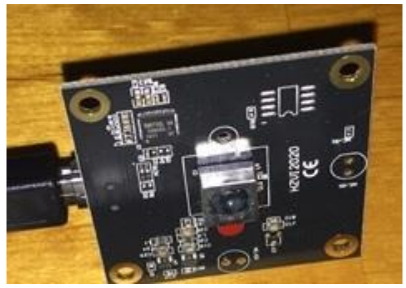


Figure 4-13: CCD Camera (Trembley, 2017)

The Multinominal Latent Logistic Regression (MLLR) is a softcomputing technique that uses a supervisory training for image processing (Xu 2016). It is used in this application to improve the performance measures against standard image recognition software, such as the Linux OpenCV. The MLLR is a refined Latent Structural Support Vector Machine (LS-SVM), which is known to be a good classifier, and extends the ability of the LS-SVM hybridization using Regularized Multinominal Logistic Regression (RMLR) (Zu, 2016).

The Algorithm 2 is used in the training of the MLLR. This algorithm uses a Gradient Descent (GD) method to improve the overall learning performance of the image processor:

Step 0. Initialize training data and latent variable for positive examples.

Gradient Descent Loop:

Step 1. While true, do Steps 2-4.

Step 2. Relabel the latent variables.

$$Optimize\ h_i\left(w_k^{(t)}\right) = \arg max_h w_k^{(t)} \cdot \phi(x_i, k, h)$$

Step 3. Update model parameters.

$$Update\ w_k^{(t+1)} = w_k^{(t)} - \alpha_t \cdot \nabla l(w_k^{(t)})$$

Step 4. Output w.

To describe the decision-making process that needs to happen in order for the Blackjack game to be played, Figure 4-14 shows the input and output of the BPN. The player's count, which is known in full, and the dealer's card count, which is partially known, will determine whether the Blackjack Player wants another card (Hit) or does not (Stand). The BPN configuration for this research is a simple input and output model.

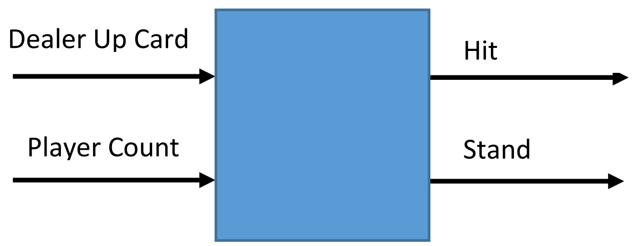


Figure 4-14: BPN/FIS 2-Inputs 2-Outputs (Trembley, 2017)

The schematic for the Blackjack Player BPN is shown in Figure 4-15. The similarity to the Deep Learning Multilayer Perceptron architecture is the BPN learning (see Figure 3-4). It is therefore considered a good choice of algorithms for this design.

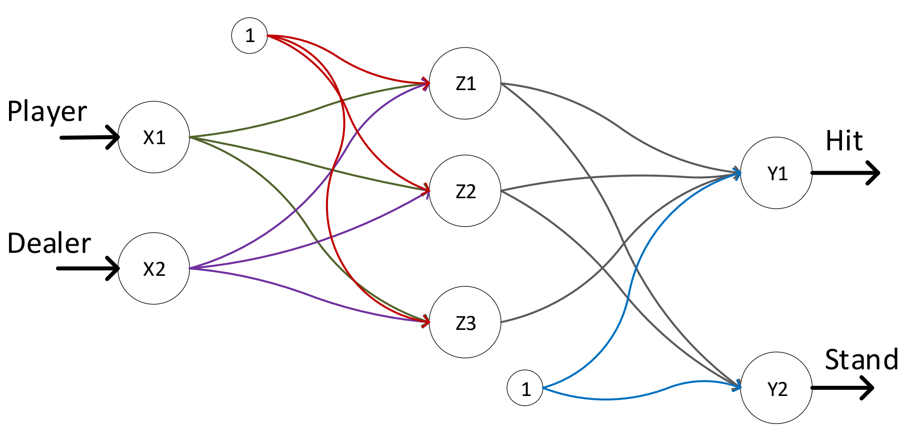


Figure 4-15: Blackjack BPN Schematic (Trembley, 2017)

There are three items to address for the BPN configuration:

- Training of the network,
- Testing of the network,
- Setting up of the network.

Training the BPN is accomplished by using a training data set that has a built-in collection of known playing card selections. Testing the network's training against another verified data set, or even known real-world outcomes, allows the verification of the training. An example of the data set for the Blackjack BPN training is shown in Table 4-1. The entire data set contains only 40 elements, even though there are hundreds of combinations of card deals that could be used. This points to the power and robustness of a neural network that is given limited information, and based on similarity of common circumstance, the BPN makes possible good decisions based on its machine learning.

Table 4-1: Blackjack BPN Training Data

ID	PlayerCount	DealerCount	Action
1	10	8	Hit
2	17	5	Stand
3	15	10	Hit
4	12	7	Hit
5	13	2	Hit
6	20	11	Stand

For the setup of the BPN, only three variables are considered: learning rate (α), activation function, and momentum. The learning rate (α) is a constant parameter used to control the speed of the Gradient Descent (GD). Momentum is an adjustment of a learning rate that improves the neural network's response due to errors in the training set. Additionally, the activation function transforms the net input to a neuron into its activation thresholds. The BPN will have to be treated as a black box where only the inputs and outputs are observed.

An Epoch is a complete pattern training iteration to the data set. The number of Epochs that the BPN is trained to is set at 2,500. This should be sufficient to adjust the weights to values that do not change much with additional training. That is, the neural network has finished training and will not learn much more. In this case, the network is now “learned.”

The form of the BPN training algorithm is as follows:

Step 0. Initialize weights to small random values.

Step 1. While true, do Steps 2-9.

Step 2. For each training pair, do Steps 3-8.

Feedforward:

Step 3. Each input unit receives the input signal and broadcasts it to the hidden layer.

Step 4. Each hidden layer unit sums its weighted input signals and sends signals to the output units.

Step 5. Each output unit sums its weighted input signals and applies its activation function to compute its output signal.

Backpropagation of error:

Step 6. Each output unit receives a target pattern corresponding to the input training pattern and computes its error.

Step 7. Each hidden unit sums its delta inputs, multiplied by the derivative of its activation function, to calculate its error information term. It calculates its weight correction and calculates its bias correction term.

Update weights and biases:

Step 8. Each output unit updates its bias and weights. Each hidden unit updates its bias and weights.

Step 9. Test stopping condition, set to false if complete.

Once we arrive at a good set of weights for the Blackjack Player BPN, a test for the verification and validity of the data can be used.

For testing, the test data of the BPN will be different from the training data. The weights generated from the training will be used, but it is imperative to use a second set of data to understand the quality and reliability of the network. The testing of the BPN is relatively simple compared to the training algorithm. The neural network accepts its two inputs, and the outputs determine whether to take another card or stand. There is the data to be used that will determine whether the Blackjack Player Hits or Stands.

The following is the general form of the testing and application algorithm for the BPN:

- Step 0. Initialize weights from the training algorithm.
- Step 1. For each input vector, do Step 2-4.
  - Step 2. Set the activation of the input units
  - Step 3. Set the hidden layer.
- Step 4. Capture the outputs from the BPNN.
- Step 5. Use the application to compare results.

The Blackjack BPN is designed with 2 input nodes, 3 hidden nodes, and 2 output nodes (see Figure 4-15). The learning rate is set at α = 0.5 and momentum = 0.5. The values of the learning rate and momentum are arbitrary. In this case, however, the performance of the network will either improve or degrade based on the selection of these variables. The lower the learning rate, the longer the network could take to train.

The type of activation function chosen is called the bipolar sigmoid function:

$$f(x) = \frac{2}{1 + \exp(-x)} - 1; range(-1, 1)$$

Each node in the Blackjack BPN will use this equation to figure out its activation. Figure 4-16 shows a partial screen shot of the activation function calculations on the network node during training.

Activations

Trial	Dealer Input	Output	Player Input	Output
0	-0.282596	-0.277245	0.959239	0.930328
1	0.987397	0.926438	0.158263	0.124410
2	-0.742784	-0.768803	0.669531	0.710500
3	0.963356	0.950881	-0.268225	-0.122236
4	-0.910491	-0.888234	0.413529	0.381704
5	-0.446427	-0.453335	0.894820	0.903116
6	-0.781594	-0.800954	0.623788	0.657729
7	0.290957	0.273779	0.956736	0.925527

Figure 4-16: Trial Input / Output Activations (Trembley, 2017)

The Fuzzy Inference System (FIS) references the Fuzzy Inference Hash Table probabilities given a count of the dealt cards. This will be very critical to making the final determination of Hit or Stand. The FIS strategy will be to use the probabilities generated by the BPN and make the final decision of Hit or Stand. For example, if the BPN recognizes that it has an 80% chance of beating the dealer and suggests a Standing, the FIS is used to tweak the solution based on probabilities of how many cards have been played and how many players are also in the deal.

4.4 Blackjack Player Experimental Results

The results of the experiment show that by using only a Blackjack BPN strategy, the win-loss ratio does reach an equilibrium as predicted by game theory. For example, from Blackjack game 20 on, the win/loss percentage remained a steady at 41% (see Table 4-2). With the Blackjack BPN trained to only one strategy, the Blackjack BPN will only win about 42% of the games played. The dealer is still in favored in winning. In review of the training data set for the backpropagation neural network (see Table 4-1), the training data set contains the opportunity for the Blackjack player winning at least 54% of the games played. However, in the face of real-world odds and playing against a human dealer, it is much less.

Table 4-2: BPN Only Win/Loss Ratio

Total Games Played 55			
Trial #1	Wins	Losses	%
BPN	23	32	41.8182%
Dealer	32	23	58.1818%

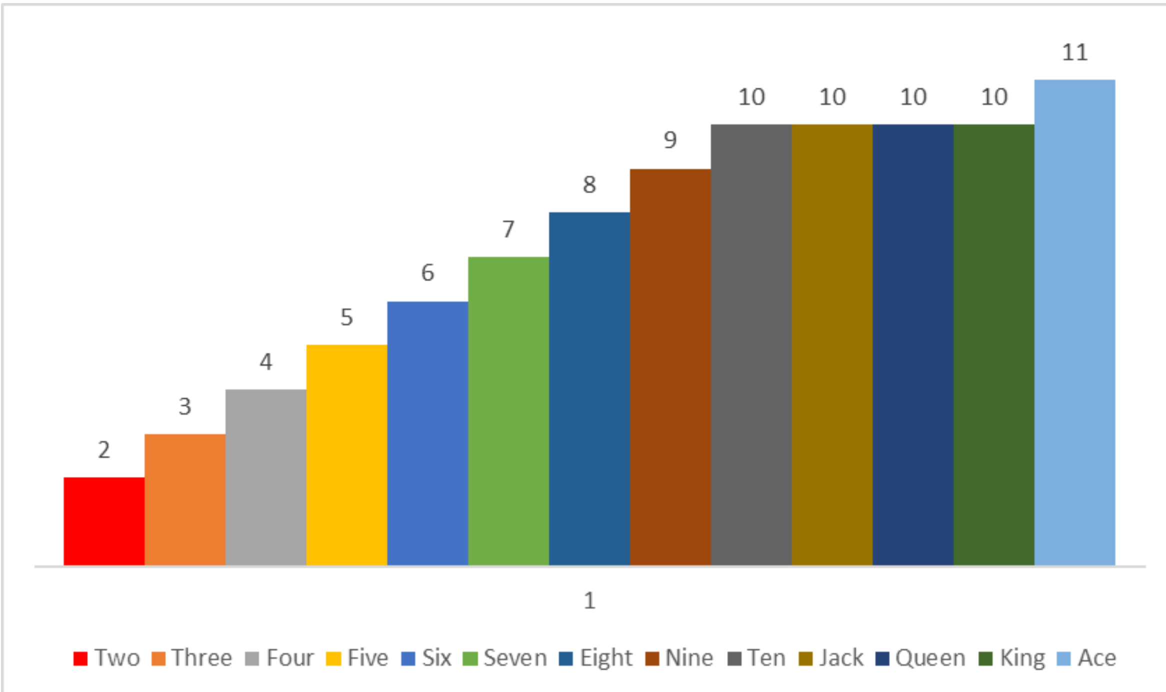
Using a combined Blackjack BPN/Fuzzy Inference strategy, which makes the Blackjack Player a more hybridized system, the win/loss ratio again achieves an equilibrium, but becomes slight worse in its decision making. From Blackjack game 20 on, the win/loss percentage remained steady at approximately 35% (see Table 4-3). However, when using the Fuzzy Inference to determine whether the player should Hit or Stand, the win/loss ratio actually worsened. This is a result of not adjusting to the marginal calls of hands involved in certain plays. That is, the FIS caused the system to act more conservatively by not taking risks associated with the BPN alone.

Table 4-3: BPN with FIS

Total Games Played 55			
Trial #2	Wins	Losses	%
BPN	19	36	34.5455%
Dealer	33	22	60.0000%

In the game of Blackjack, the binary choice of Hit or Stand is based on applied rules of whether in the face of odds for or against the dealer, is not a discrete decision. For example, if the BPN is trained to stand at the card count of 17, while the dealer possesses an up card count of 10, the player is in the marginal range of winning the hand. That is, the dealers down card only needs to be an Ace, 10, 9, 8, or 7 to win. The game is statistically in the dealer's favor. The distribution within a randomly shuffled deck indicates that it will take luck to win (see Table 4-4) as more than 67% of the cards remaining that the dealer could have is most likely the case.

Table 4-4: Card Deck Distribution



4.5 Summary

A modeling framework that contains the resulting data necessary to verify and validate both the performance of the system (meaning its ability to sustain and react to the operational environment), and the correctness as a measure of the performance (meaning the end result) will be conducive to a system that continually operates as designed. This is Proposition 2 from Chapter 1. The linear and non-linear attributes of the CPS are a complex set of feedforward and feedback loops that are best describe using the OODA loop model. The OODA model can be seen as information flow through the system as a matter of input sensors and output actuators; however, the information internal to the system remains hidden and unknown. With better models of the CPS, the test bench setups of the system's natural architecture emerge as a meta-model. The engineering tools allow for the data extraction of some of the hidden information, whereby the model predicts the action and reaction of the system. In a sense, the system can be tuned. This is because the hidden information is the memory of the system, and it is the learned memory that is of most importance as it will need adjusting through training. The two proposed bench systems, the Moon Lander and the Blackjack Player, provide a framework of how this memory extraction is accomplished and how to adjust. In each case, the AI systems within the CPS, interacting with a human-agent, is investigated using the principles of model-based development, real-world prototypes, and non-invasive test access points.

Chapter 5: Application of the Thesis

5.1 Introduction to Applications

The frontier of modern softcomputing employs very fast computers and their associated algorithms operating on state of the art hardware. Algorithms that execute the probability models in the decision-making aspects of the machine-agent for decision-making will compress the timing latency of such systems for the foreseeable future. Softcomputing algorithms -- Boltzmann and State Vector Machines (SVM), Radial Based and Backpropagation Neural Networks, Self-Organizing Maps, etc. -- require significant execution overhead that would prevent decisions and actions to be made in a timely manner (Geyer and Carle, 2016; Kang et al., 2018). As the CPS systems become more complex as a result of the increase in operational complexity with broad range functionality, and with increased levels of automation that the system needs to achieve in order to carry out its end designed objective, the CPS becomes increasingly difficult to test, and harder to verify and validate as to performance and correctness (performance being related to the function, and correctness related to the behavioral). However, through an understanding of the system's model, simulation can and should be used, not only as a proof of concept, but as an official part of the record in the certification of the system's use. Applications of the current theory are well underway but are still lacking the infrastructure of sound and bullet proof testing (Hernandez et al., 2018).

5.2 Autonomous Vehicles

The autonomous vehicle is part of an ongoing development of CPS intelligent technologies. In the commercial, industrial, military, medical, governmental, and transportation sectors, has a unique role and requirement that can benefit from leveraging the machine-agent to higher levels of autonomy in vehicular motion (Korssen et al., 2018). Each sector provides its own oversight, not just internally to its own guidelines and principals, but to rules and regulations that legally, through licensing and certification, allow the CPS autonomous vehicle technologies to be used in existing infrastructures, such as the highway system (Gao et al., 2016; NTSB, 2015). The SAE defines levels of automation for autonomous vehicles, but the SAE is only the "specifying" authority for the automotive industry (SAE INTERNATIONAL, 2014); a company like Tesla will use the SAE specifications in the design of their autonomous car; however, it is the Highway Transportation Board (HTB) that will approve any real testing on the highway or certify the autonomous vehicle as safe for driving on the highway (Ohta et al., 2017) .

The autonomous car, when fully certified and licensed for use, will become part of the highway infrastructure as overseen by the US Highway Transportation Board. Additionally, the National Transportation and Safety Board (NTSB) has the overall say in whether to license an autonomous vehicle or not. And this point is made because there is an ongoing debate regarding whether the autonomous car gets licensed independently or whether a licensed driver must be with the autonomous vehicle. The liability using autonomous vehicles has also to be defined (Lazanyi and Maraczi, 2017). In the matter of the collision avoidance system (CAS), this level of automation is not considered part of the licensing process of the car, since the car needs a licensed driver, but instead, part of the certification process that the automobile needs to establish in order to be considered safe for driving on the highways and interstates (Schnelle et al., 2017; Vahidi and Eskandarian, 2003).

5.3 Unmanned Aerial Vehicles

Since CPS autonomous vehicle systems are what can best be described as "cutting edge," the distinctions of a regulating authority are fuzzy. Technology is sometimes so new and novel that it is uncertain how to categorize it. Other times, the technology is a hybrid of different uses, and could fall within the realm of several regulating authorities (Sun et al., 2017). This is the case of Unmanned Aerial Vehicles (UAV) where the FAA and local and state governmental authorities interact. Because federal and state laws differ, and state to state laws and regulations differ, understanding the laws often will determine where a technology is developed (FAA, 2018).

Unmanned aerial vehicles, a.k.a. "drones", are controlled remotely or fly autonomously. As with every new high-tech "thing" on the market, there is also a learning curve. For example, there are many reported incidences of airline pilots in landing patterns at heavily congested airports that spot drones at or near the altitude they are flying. Near misses and collisions between UAV and aircraft have been reported (Carey, 2017). Because these incidents are very worrisome to the airlines and the FAA, congress established the *FAA Modernization and Reform Act of 2012* to regulate the commercial drone industry. The FAA needed laws to regulate the unexpected popularity of commercial off-the-shelf (COTS) drones, and their subsequent misuse by the consumer (Carey, 2017).

The commercial drone industry has not slowed in popularity due to increased regulations. The commercial sector has the widest set of uses for UAV technologies, everything from crop dusting to aerial inspection of power lines. The sales of drones grew from $44 million in 2013 and is estimated to reach $1.3 billion in 2017 (statista, 2017). Plans for Amazon to use drone technologies to deliver packages from warehouse to door is being engineered. The days of the pizza delivery person are numbered.

5.4 Internet of Things

Continuing the coffee maker example from Section 3.3, consumers have a wide variety of choices of coffee makers with various levels of automation. The automatic coffee maker is an ideal example of levels of automation in design. Because coffee makers of simple design only have an on/off button to press to start the brewing process, this is the simplest example of a CPS at a very low level of automation. The person still has to fill the coffee maker with water, feed in the ground beans, and figure out when the brewing process is complete; however, there are now coffee makers that grind the coffee beans, set the water levels, and schedule when and what temperature the coffee is made. The person has very little to do with this coffee maker from the initial setup except to make certain that the supply of beans and water in the hoppers are adequate. The person can also remotely control the coffee through a smartphone app that can enable or disable features or just check in on the status. This example considers the coffee maker as an object within an Internet of Things (IoT), a CPS system derived from existing technologies (Dhanalaxmi and Naidu, 2017).

Garage doors can be checked over the smartphone to verify whether the door is open or not. A person can remotely open and close the garage door using the smartphone. However, the concept of taking a “thing” and hooking up to the Internet gets frightening because often times the technology is not as reliable as the marketing brochures claim. For instance, getting the garage doors on WIFI can be somewhat difficult, and because the company who manufactured the garage door opener probably did not secure the communication channel between opener and the WIFI router, a person using this technology is now more vulnerable to a threat vector from outside attack.

5.5 Security

In terms of the state of cyber-physical embedded system technology, and where it will be in the foreseeable future, there are a few general rules and guidelines that can be applied. The CPS systems will be wireless, more capable, and safer (Bajpayee and Mathur, 2015; Backhaus et al., 2013). Currently, it is known that there is a security problem with this technology. Much of it is not safe from intrusion and infiltration. Many home Internet of Things (IoT) devices transmit wirelessly, but the wireless transmissions are not secure, so the transmission can be intercepted by an outside source monitoring the signals. This outside source, for good or bad, while monitoring the signals is invading privacy but is essentially legal as long as the signals intercepted are outside the property. The intercepted signals have the potential for being "hacked," in which possible malicious behavior to the CPS could occur. The scenario would go like this: the homeowner leaves for the day to go to work. A thief uses a device that captured the signals from the garage door opener and decodes the captured signal to open the garage door to gain access inside the home in the owner's absence. In a worse case example, the hacker could knowingly turn on devices without proper setup, such as turning on an empty coffee maker, or turning on the kitchen stove while flammable materials are on top. This has the potential for damage to the device, persons, and property.

For example, in 2012, the Stuxnet computer worm was introduced to an Iranian nuclear power plant; the Stuxnet worm took control of the CPS system that controlled the uranium centrifuges by over-speeding the centrifuges, causing massive destruction to the machinery. If security of CPS systems is required, this adds another layer to the design that is not part of the functionality, but the correctness of the system designed to be hardened against attacks (Kushner, 2013).

Unfortunately, there are no clear rules or regulations to force manufacturers to build in safeguards against attack vectors. However, there are manufacturers that do take security seriously. A case in point is the General Motors (GM) OnStar system. Since cars have a lot of automatic features (automatic door locks, remote engine start, etc.), GM OnStar provides a secured communication link between the car, the owner, and a remote manager. Providing remote assistance to the authorized user of the GM vehicles, command and control over the automated features become seamless. If a person with a GM vehicle gets locked out of the car, the person can call OnStar with an ID, and the remote GM OnStar manager will unlock the car. This service seems beneficial, and other car manufacturers offer similar type services but without the necessary security in place. This could raise questions of safety if access to systems can be remotely control while the CPS is in operation (Zhang et al., 2016).

5.6 Safety

Using the test architecture described in Section 4.2, security can be tested concurrently with the functionality by using attack vector scenarios that attempt to intercept or take control. The US Consumer Product Safety Commission (CPSC) has oversight on products within the United States; however, the CPSC is a reactionary organization to safety concerns of products being sold on the market. The CPSC is not a regulatory agency but a watchdog group that collects data being reported by and about consumers, typically through hospital emergency room visits. If there is sufficient evidence that a product is unsafe, the CPSC will issue a notification and potential recall. Most manufacturers of products sold within the US perform a certain amount of testing to certify their product as safe. For example, electronic products, cell phones, laptops, televisions, etc., usually go to the Underwriter Laboratories (UL) for testing the safety of the product, and also have Federal Communication Commission (FCC) testing as well for Electro-Magnetic Interference (EMI).

If the product is to be sold in Europe, the product must get the Conformité Européenne (CE) for safety and the Technischer Überwachungsverein (TUV) for safety and EMI certification. This reduces the overall liability if a product does not behave as expected.

These are examples where even rigorous testing demonstrated compliance to the rules and regulations and allowed the sales of a product to consumers, but the products still have turned out to be very dangerous. This is the case of the Samsung Note 7. The Samsung Note 7 has caught on fire on planes, in people's pockets, etc. even though the Note 7 had all the certifications from testing. The problem was a defect in the lithium ion battery that caused the phone to suddenly explode and catch fire. The eventual loss in revenue for Samsung for this one product is estimated between $5 to $17-billion (Mullen and Thompson, 2016).

5.7 Hardware and Software Maintenance

Additionally, with any technology, the technology will require upgrades usually in software, but also hardware. This is the life-time cycle management paradigm that occurs with most technologies. However, the question is, "Does the product with an upgrade need to be retested and recertified?" This is a problematic question, and one where modeling and simulation could help quickly resolve questions of impact to the system updates. For example, if the product gets a software upgrade, the manufacturer could run simulation scenarios using the design changes to previous simulations that the product was certified with and provide proof that the changes make the product better or fix the bugs from the previous release. This data could then be used as evidence to the certification authority, FCC, CE, UL, etc. The regulating authority, manufacturer, and consumer would see this as beneficial in saving time and money as compared with a full retest as done in the initial certification of product. In the past, regulating authorities were reluctant to use modeling and simulation as a method of certification. However, this idea is changing as modeling and simulation methods are vastly improved from where they were a decade ago (Quintas et al., 2017).

Chapter 6: Conclusion

This thesis frames automated or autonomous systems at the cyber-physical embedded system level. Embedded systems that interface with larger systems that form networks for monitoring and communications are part of the CPS considered the Internet of Things (Xin et al., 2015; Chen et al., 2018). These embedded systems must apply known principles of human factors to the design. The ability of the human-agent to adapt to how the machine reacts is paramount to the usefulness and eventual success of the system. This essentially becomes the performance measure of the system, and for optimal human agency to interact and benefit from the use of an automated system, the system must be designed within the limits of skill, knowledge, and experience of the user. This is especially the case in safety critical applications (Korssen et al., 2018; Quintas et al., 2017).

Past experience with system design has considered the human-agent as the stimulus-response function within the system; however, today, many systems have autonomous features that use softcomputing techniques such as artificial neural network, genetic algorithms, fuzzy logic, etc., which obviate the necessity of human-agent action. However, these systems are difficult to test for functionality, performance, and correctness based on dynamic changes in the operating environment (Novak et al., 2017).

The response to external events that are shared between human-agent and machine-agent are shown in roles of changing responsibilities to those events that require that the level of automation be well known (Zhang et al., 2016). Both the experiential element of human-agent and the hard-wired experience of the machine-agent must navigate and ultimately reach decisions that are at least optimal or better than what could be achieved by either alone. However, with reliance on ever higher levels of automation, the human-agent can become physically and cognitively disassociated from the system (Parasuraman et al., 2000). This could lead to disruptive and disastrous consequences that jeopardize the value and success of the system. Understanding measurements of complex human-agent and machine-agent response states, especially in uncertain environments, requires that critical activities be fully tested and proven prior to the system being safely deployed (Sheridan, 2002). The use of modeling, simulation, and prototyping become the ultimate tool in the verification and validation process. Having a good model provides a method to test the design prior to service, or if in service, allows methods to demonstrate how system upgrades can be safely adapted and operated. Simulation gives insights to the changes of input stimulae by monitoring the output response. Prototype gets the system close to the final system that allows for deep insight to the hidden variables in the system and emergent behavior (Korssen et al., 2018).

The sensors, software and hardware systems that comprise the CPS embedded system derive from technologies where the design, test, and deployment of intelligent technologies is difficult and techniques often poorly understood (Novak et al., 2017). The choice of using greater autonomy in systems requires designing the system to the appropriate level of automation and knowing how the CPS embedded system leverages the ability of the human-agent and the machine-agent to form a hybridization of statistical methodologies and techniques for decision-making (Xin et al., 2015). It is not enough to use standard design and engineering principles and practices in design and test. It is necessary to understand the “systemness” of such design to achieve a better understanding of the system’s core competencies. In order to achieve technological milestones in the development and deployment of such systems, the available technology must take into account the knowledge of how these systems will be used. This thesis opens the door to pushing the frontiers of how these systems should be designed, tested, and ultimately fielded (Alippi et al., 2017; Layadi et al., 2015).

The framework of human-agent and machine-agent design, in order to be effective, needs to coincide with technologies that employ the correct level of automation. These machine-agent intelligent systems involving the human-agent are difficult to test because of transient upshifting or downshifting in the machine-agents level of autonomy. Once a shift in the level of automation occurs, the roles and responsibilities of the machine and human change (Sheridan, 2011). The standard black box approach limits the level of understanding of such systems -- that is, "are the systems safe to use if a transient downshift occurs?" As the case studies in passenger airline and train disasters are a reminder (NTSB, 2014; NTSB, 2015), the answer is "No." This is due to the unknown complexity of hidden variables that the automated system is attempting to solve with. Automation in systems such as driverless cars, drones, collision avoidance systems, etc., use probabilistic models, and the decision systems of these systems are used mainly to cognitively offload tasks from the human-agent. So, before any CPS embedded system is deployed, the roles of human-agent and the machine-agent to the level of autonomy and expected autonomy must be well understood.

The future of CPS embedded systems using softcomputing, especially CPS systems embedded within larger CPS systems, raises the possibility that such systems can and will help and prevent accidents and become optimally efficient (Thalassinakis et al., 2006; Corno et al., 2016). However, misuse of such a system is likely to occur without sufficient understanding of the entirety of the environment in which these systems operate. The need for safeguards is paramount to prevent the misuse and abuse of such systems. The machine-agent and human-agent are prone to Type I and Type II errors. However, the supervisory command and control structure of either the machine or human should allow for the actionable decisions to remain viable, with the ability to recover even under high levels of uncertainty (Xin et al., 2015). This is Proposition 2 from Section 1.

In adapting the principles of many disciplines such as biology, economics, computer, cognitive system, human factors, electrical, mechanical, systems engineering, etc., the thesis takes a multidisciplinary approach to CPS systems in science and engineering in its design and test, but also in its effects on the world at large. Using CPS embedded systems, such as IoT or M2M, machine and human intelligence using similar guidelines and principles will have potentially some very beneficial outcomes (Lee, 2015). The features of the CPS embedded system will allow for the human-agent to operate with different levels of knowledge, learning, and language. This safeguards the control of the system so that it operates in its intended effective operational manner. In the design of these systems, taking the human out of the loop is not the best of ideas, but by coupling the knowledge of the design to the cognitive ability and expectation of the operator, the operator trains, learns, and reacts to the normality of the conditions as they arise (Parasuraman et al., 2000).

The idea that the future will be handled by very smart "intelligent" machines is essentially here. In the recent past, designing intelligent systems with any level of sophistication and robustness was impossible because the computational requirements of such a system were not available. Most current CPS embedded systems perform simple tasks that remove the human-agent from the design equation without any impact to safety. The automatic coffee-maker example shows that although the cup of coffee end result remains unchanged, the methods employed are vastly different within the current time and epoch when such advanced technological possibilities are realized. But replacing human agency with a machine tends to dehumanize. The human-agent becomes a stimulus-response to the design of the system's operation, which is gradually being replaced as technology advances. In many cases, the human is treated as a bio-mechanical computer. As Werner von Braun was once quoted as saying, "The best computer is a man, and it's the only one that can be mass-produced by unskilled labor."

Although computer systems offload much responsibility from the human-agent, it is the human agency requirement, the “human-in-the-loop”, that determines whether the system succeeds or fails, and the human agent should be the final arbiter of the system.

Chapter 7: Future Research

7.1 Introduction to Research and Development

The need for research and development in the field of CPS embedded systems will continue unabated for the foreseeable future. Government agencies like DARPA, NASA, and ESA continue to push the frontiers of cyber-physical and other intelligent technologies. Corporations like Lockheed Martin, Boeing, Tesla, Sony, Honda, Samsung, Apple, Google, etc. also are in the business of advancing to continually push the limits of cyber-physical systems technology. The future outlook is promising for research funding and development dollars (Lee, 2015).

7.2 Mathematical Models and Meta-Modeling

A formal mathematical model would be a nicety for the development and validation of cyber-physical systems; however, the feasibility of producing a good and well-understood mathematical model of a CPS may not be possible. The complexity of the system does not allow for a descriptive model to be easily formulated. Although control system theory describes linear and non-linear dynamics in both discrete and continuous systems or a hybridization of both, this thesis recognizes the drawback of using such structured methods to describe CPS's in terms of performance and correctness where complex hardware and software systems interact with multidimensional time domain. The majority of cyber-physical systems are hybrid versions using combinations of the fore mentioned. In the context of using non-trivial intelligent systems with the greater construct of the CPS, system behaviors based on unknown or unquantified inputs are likely to attribute to abrupt disorganization of the system, where the system must adapt or fail. Since catastrophic failure within the context of an operational system is not a desirable outcome, in safety critical systems, the liability of not investigating far enough the range of these narrow system behaviors is problematic in the resource intensive activities of design verification and validation. The question is then asked, "Is there a method that can provide the model with sufficient oversight that allows the details of the systems to always operate within a safety window?"

In researching this question, the DARPA META Program was discovered (DARPA, 2017). The program was founded on the idea that improvements in the integration and testing of complex cyber-physical systems must rely on a model-based design method that takes into account hierarchical abstractions in the system's architecture (Korssen, et al. 2018). The true mathematical models are now dispensed with in favor of further abstracting the systems as a set of objects. DARPA has essentially developed a new set of tools that allows for the verification and validation of the complex CPS design. The tool set is call the META Tool Suite and was developed to improve defense contracting manufacturing and development processes by providing a formal meta-modeling language. System engineering of large scale projects would proceed along the lines of the following:

> *…to optimize system design with respect to an observable, quantitative measure of complexity for entire cyber-physical systems; and to apply probabilistic formal methods to the system verification problem, thereby dramatically reducing the need for expensive real-world testing and design iteration (DARPA, 2017).*

This is the problem the thesis is attempting to resolve. With the increasing complexity levels of cyber-physical systems, the meta-modeling through hierarchical abstraction is an alternative to that of traditional structural modeling methods. With the development of meta-languages and tools, the challenge is to investigate the application of hardware and software within the context of the cyber-physical system designs. This begins a research position to better understand the implications of the CPS in operation, and how trust is built into the systems.

7.3 Cyber Security in Cyber-Physical Systems

Cyber security is a major concern, and a large part of cyber-physical system architecture. In Chapter 2, the discussion of “hackers” taking control of cyber-physical systems, like an IoT coffee maker or a Roomba™, is a real possibility and a big problem for designers of such systems (Khorrami, et al., 2016). The cyber security aspect of CPS’s is especially important when protecting critical infrastructure, such as power systems, emergency communication systems, transportation, etc. The Stuxnet worm (Section 5.5) is an example of how cyber-physical systems can destroy themselves from the inside out. The proposed research in the area of cyber security with cyber-physical systems is then to form a model, using the META tools mentioned in Section 7.1, to demonstrate the inherent gaps or flaws that could lead to dangerous problems when two or more cyber-physical systems are combined. Since many cyber-physical systems embed themselves within larger systems, the hierarchy of their Systems of Systems (SoS) architecture needs to be verified for its usability, maintainability, and security (meaning that the “good” guys get to use the system, and the “bad” guys are denied access to it).

In the Figure 7-1, two CPS’s are shown and share a connection called the “CPS Access Bridge.” The CPS Access Bridge is how information is passed between the two systems. Both systems are in the Cloud, and each system has a different level and type of security protocol. For example, CPS 1 is connected to the Internet and uses a password authentication protocol, while CPS 2 has to be physically accessed at the site. However, because CPS 1 is able to be accessed remotely, and it is connected to CPS 2 through the bridge, the question becomes, “Can it be shown that the access gateway from CPS 1 to CPS 2 is secure?” In cyber security terms, the combining of the two systems decreases security though an increase in system complexity. Both systems provide potentially more unsecured channels through compromised passwords and other unknown access points. It is in this manner that the possibilities of exploiting existing security vulnerabilities becomes even more of a concern.

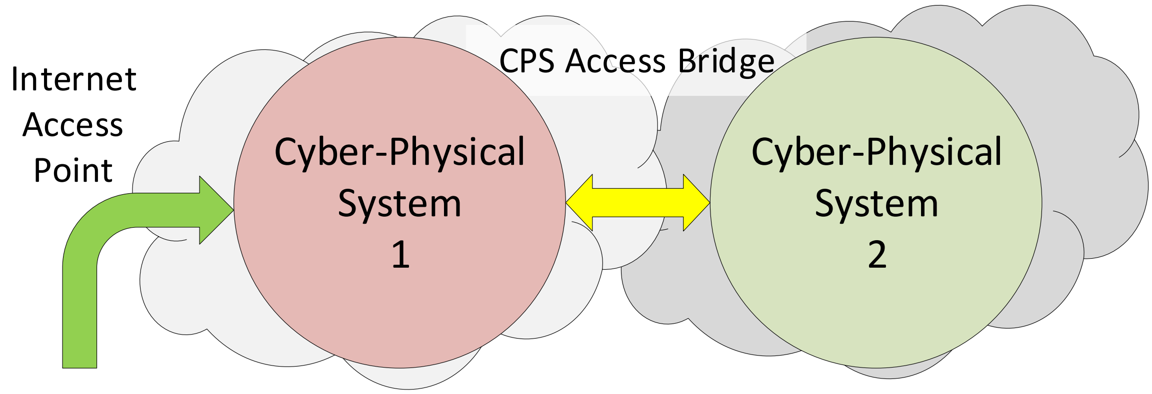


Figure 7-1: CPS Cyber Security (Trembley, 2018)

7.4 Critical Systems using AI

The definition of a safety critical system is a system whose failure will result in death or serious injury to people and damage to equipment and property (Trembley, 2018). To evaluate a CPS using AI, there are three attributes that must be addressed: security, criticality, and robustness. Each attribute combines both the usefulness and reliability of such a system, and there are trade-offs involved. It is the tradeoffs that must be carefully researched in order for the goals of the system to be accomplished. Any CPS design will require all three attributes to be addressed and quantified for a level of cyber-security to prevent disruption from attacks. The cyber security guidelines and standards for network and data protection is needed ensure the confidentiality, integrity, and availability of the CPS. This is addressed in the preceding, see Section 6.

The criticality is the value placed on the importance of the “things” in the system. Researching CPS criticality is a result of its function within the larger system as a measure of its associated risk in the system. For example, the CPS can monitor or control processes with high degrees of criticality. Since criticality is a difficult number to assign, in critical systems, redundancy and failsafe emergency shutdowns are often employed to increase the reliability of the system in case of unanticipated issues. This is where the research needs to understand the criticality of the system and how it behaves.

The robustness of the CPS system has to be capable of adapting to changes in the operational environment without suffering physical damage or loss of the critical features of its functionality. The design of the CPS needs to be capable of detecting equipment malfunctions, false alarms, and cyber-attacks. This is where the artificial intelligence becomes a necessary agent to the orderly and safe operation of the system. A CPS that is constantly exposed to a complex set of environmental stimuli will need to navigate the complex nature of faults, false alarms, and mismatches. The engineered reliability of the component devices that comprise the CPS is where the research into the areas of robustness, security, and criticality is paramount to the overall success of the CPS.

References

Alippi, Cesare, Stavros Ntalampiras, and Manuel Roveri. 2017. "Model-Free Fault Detection and Isolation in Large-Scale Cyber-Physical Systems." *IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE* 1 (1): 61-72.

Arne, Worm. 2000. "Information-centered human-machine systems analysis for tactical command & control systems modeling and development." *IEEE International Conference on Systems, Man, and Cybernetics.* IEEE. 2240-2246.

Ashby, William R. 1956. *An Introduction to Cybernetics.* London: Chapman & Hall.

Backhaus, Scott, Russell Bent, James Bono, Ritchie Lee, Brendan Tracey, David Wolpert, Dongping Xie, and Yildiray Yildiz. 2013. "Cyber-Physical Security: A Game Theory Model of Humans Interacting Over Control Systems." *IEEE TRANSACTIONS ON SMART GRID* 4 (4): 2320-2327.

Bajpayee, Divyajeet, and Jitendra Mathur. 2015. "A Comparative Study About Autonomous Vehicles." *IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems.* Tamil Nadu: IEEE. 1-6.

BEA. 2012. *Final Report On the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France flight AF 447 Rio de Janeiro - Paris.* Accident, Bureau d'Enquêtes et d'Analyses pour la sécurité de l'aviation civile, Paris: Ministèredel' Écologie, du Développement durable, des Transportset du Logement.

Billings, Charles E. 1997. *Aviation Automation: The Search For A Human-Centered Approach.* Mahwah, New Jersey: Lawrence Erlbaum Associates.

Blackwelder, Britt, Katherine Coleman, Sara Colunga-Santoyo, Jeffrey S Harrison, and Danielle Wozniak. 2016. "The Volkswagen Scandal." *Case Study* (University of Richmond: Robins School of Business) 1-25.

Boisot, Max, and Bill McKelvey. 2011. "Complexity and Organization–Environment Relations: Revisiting Ashby's Law of Requisite Variety." *Complexity and Managment.* Washington, D.C.: Sage.

Bomey, Nathan. 2018. *Volkswagen reveals futuristic AI driverless concept vehicle at Geneva Motor Show.* USA TODAY. Geneva, March 5. https://www.usatoday.com/story/money/cars/2018/03/05/volkswagen-id-vizzion-geneva-motor-show/396096002/.

Borenstein, Jason, Joseph Herkert, and Keith Miller. 2017. "Ethical Responsibilities of Design Engineers: Self-Driving Cars." *IEEE Technology and Society Magazine* 67-75.

Bosetti, Paolo, Mauro Da Lio, and Andrea Saroldi. 2015. "On Curve Negotiation: From Driver Support to Automation." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 16 (4): 2082-2093.

Carey, Bill. 2017. "AIN Online." *AIN Online.* October 15. Accessed March 11, 2018. https://www.ainonline.com/aviation-news/air-transport/2017-10-15/transport-canada-reports-first-collision-drone-airplane.

Chen, Ching-Han, Ming-Yi Lin, and Chung-Chi Liu. 2018. "Edge Computing Gateway of the Industrial Internet of Things Using Multiple Collaborative Microcontrollers." *IEEE Network* 24-32.

Chen, Jessie Y. C., Michael J Barnes, and Michelle Harper-Sciarini. 2011. "Supervisory Control of Multiple Robots: Human-Performance Issues and User-Interface Design." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS* 41 (4): 435-455.

Corno, Matteo, Simone Formentin, and Sergio M Savaresi. 2016. "Data-Driven Online Speed Optimization in Autonomous Sailboats." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 762-771.

DARPA. 2017. *DARPA META Program.* Accessed April 2, 2018. https://cps-vo.org/group/avm/meta.

Dhanalaxmi, B, and G. Apparao Naidu. 2017. "A Survey on Design and Analysis of Robust IoT Architecture." *International Conference on Innovative Mechanisms for Industry Applications.* Bengaluru: IEEE. 375-378.

DHS. 2002. *Command and Control of Wildland/Urban Interface Fire Operations for the Structural Chief Officer.* Washington: Department of Homeland Security.

Drumea, Andrei, and Alexandru R Dobre. 2014. "Modelling, Simulation and Testing of an Autonomous Embedded System Supplied by a Photovoltaic Panel." *2014 IEEE 20th International Symposium for Design and Technology in Electronic Packaging (SIITME).* Bucharest: IEEE. 309-312.

Dundar, Aysegul, Jonghoon Jin, Berin Martini, and Eugenio Culurciello. 2017. "Embedded Streaming Deep Neural Networks Accelerator With Applications." *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS* 28 (7): 1572-1583.

European Parliamentary Research Service. 2016. "Automated vehicles in the EU." *Briefing.* January.

FAA. 2018. *Federal Aviation Administration.* January 2016. Accessed March 11, 2018. https://www.faa.gov/news/updates/?newsId=84384.

Frankot, Robert T. 2012. "Performance of Target Acquisition Systems with Ordered Commitment." *IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS* 48 (3): 2154-2166.

Gao, Jinghua Guo, Ping Hu, and Rongben Wang. 2016. "Nonlinear Coordinated Steering and Braking Control of Vision-Based Autonomous Vehicles in Emergency Obstacle Avoidance." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 17 (11): 3230-3240.

Garrido, Angel. 2012. "A Brief History of Fuzzy Logic." *BRAIN: Broad Research in Artificial Intelligence and Neuroscience* 3 (1): 71-77.

Geyer, Fabien, and Georg Carle. 2016. "Network Engineering for Real-Time Networks: Comparison of Automotive and Aeronautic Industries Approaches." *IEEE Communications Magazine* (IEEE) 106-112.

Gobbo, Federico, and Marco Benini. 2013. "From Ancient to Modern Computing: A History of Information Hiding." *IEEE Annals of the History of Computing* (IEEE Computer Socie) 33-39.

Gubbi, Jayavardhana, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013. "Internet of Things (IoT): A vision, architectural elements, and future directions." *Future Generation Computer Systems* 29 (7): 1645-1660.

Henryk F. R. Arciszewski, Tjerk E. de Greef, and Jan H. van Delft. 2009. "Adaptive Automation in a Naval Combat Management System." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS* 39 (6): 1188-1199.

Hernandez, Carlos, Mukunda Bharatheesha, Jeff van Egmond, Jihong Ju, and Martijn Wisse. 2018. "Integrating Different Levels of Automation: Lessons from Winning the Amazon Robotics Challenge 2016." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 1-11.

James, H. B. 1953. "The Use of an Analogue Comuter to Optimize the Transient Response of an Aircraft-Type Generator-Regulator System." *IEEE* 363-368.

Jin, Chong, Rui Kang, and Ruiying Li. 2018. "VTB-RTRRP: Variable Threshold Based Response Time Reliability Real-Time Prediction." *IEEE Access* 6: 60-71.

Kang, Mengya, Chenglin Wen, and Chenxi Wu. 2018. "A Model Predictive Scheduling Algorithm in Real-Time Control Systems." *IEEE/CAA JOURNAL OF AUTOMATICA SINICA* 5 (2): 471-478.

Karsai, Gabor. 2003. "Toward Fault-adaptive Control of Complex Dynamic Systems." In *Software-enabled Control: Information Technology for Dynamical Systems*, by Tariq Samad and Gary Balas, 347-368. Piscataway: John Wiley & Sons.

Katzourakis, Diomidis I, Joost C de Winter, Mohsen Alirezaei, Matteo Corno, and Riender Happee. 2014. "Road-Departure Prevention in an Emergency Obstacle Avoidance Situation." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS* 44 (5): 621-629.

Khorrami, Farshad, Prashanth Krishnamurthy, and Ramesh Karri. 2016. "Cybersecurity for Control Systems: A Process-Aware Perspective." *Perspectives* 75-83.

Kircher, Katja, Annika Larsson, and Jonas A Hultgren. 2014. "Tactical Driving Behavior With Different Levels of Automation." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 158-167.

Klien, Gary, David D Woods, Jeffrey M Bradshaw, Robert R Hoffman, and Paul J Feltovich. 2004. "Ten Challenges for Making Automation a “Team Player” in Joint Human-Agent Activity." *IEEE INTELLIGENT SYSTEMS* 91-95.

Klir, George J. 2001. *Facets of Systems Science.* Vol. 15. New York, New York: Kluwer Academic/Plenum Publishers.

Korssen, Tim, Victor Dolk, Joanna van de Mortel-Fronczak, Micheland Reniers, and Maurice Heemels. 2018. "Systematic Model-Based Design and Implementation of Supervisors for Advanced Driver Assistance Systems." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 19 (2): 533-544.

Kuprijanow, Alex. 1970. "The Effect of Airspace Loading by Controlled Air Traffic on the Air Traffic Control System." *Proceedings of the IEEE* 381-391.

Kushner, David. 2013. "The Real Story of Stuxnet." *IEEE Spectrum*, February 26: 1-10.

Layadi, Toufik M, Gérard Champenois, and Mohammed Mostefai. 2015. "Modeling and Design Optimization of an Autonomous Multisource System Under a Permanent Power-Supply Constraint." *IEEE TRANSACTIONS ON SUSTAINABLE ENERGY* 6 (3): 872-880.

Lazanyi, Kornelia, and Greta Maraczi. 2017. "Dispositional trust – Do we trust autonomous cars?" *IEEE 15th International Symposium on Intelligent Systems and Informatics.* Subotica: IEEE. 135-140.

Lee, Edward A. 2015. "The Past, Present, and Future of Cyber-Physical Systems: A Focus on Models." *Sensors* 4837-4869.

Lee, Edward A., and S. A. Seshia. 2012. *Introduction to Embedded Systems - A Cyber-Physical Systems Approach.* Berkley: LeeSeshia.org. http://cyberphysicalsystems.org/Cyber-Physical_Systems.html.

Lindsey, Ronald. 2009. "Positive Train Control in North America: What It Is and What It Isn't." *EEEVEHICULARTECHNOLOGYMAGAZINE* 22-26.

Liu, Liang-Chien, Chiung-Yao Fang, and Sei-Wang Chen. 2017. "A Novel Distance Estimation Method Leading a Forward Collision Avoidance Assist System for Vehicles on Highways." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 937-949.

Mansouri, Nazanin. 2016. "A Case Study of Volkswagen Unethical Practice in Diesel Emission Test." *International Journal of Science and Engineering Applications* 5 (4): 211-216.

Marquez, Jessica J, and Margarita Ramirez. 2014. "Level of Automation and Failure Frequency Effects on Simulated Lunar Lander Performance." *Aerospace Conference.* Big Sky: IEEE. 1-10.

Martin, Iain, Steve Parkes, Martin Dunstan, and Nick Rowell. 2014. "Asteroid Modeling for Testing Spacecraft Approach and Landing." *IEEE Computer Society* 52-62.

Moradi-Pari, Ehsan, Neda Nasiriani, Yaser P Fallah, Parviz Famouri, Steve Bossart, and Keith Dodrill. 2014. "Design, Modeling, and Simulation of On-Demand Communication Mechanisms for Cyber-Physical Energy Systems." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 10 (4): 2330-2339.

Mullen, Jethro, and Mark Thompson. 2016. *Samsung takes $10 billion hit to end Galaxy Note 7 fiasco.* October 11. Accessed March 11, 2016. http://money.cnn.com/2016/10/11/technology/samsung-galaxy-note-7-what-next/index.html.

NASA. 2015. "New Horizons: The First Mission to the Pluto System and the Kuiper Belt." *NASA Fact Sheet.* National Aeronautics and Space Administration, May 26.

National Research Council. 2006. *C4ISR for Future Naval Strike Groups.* Washington: The National Academies Press.

National Transportation Safety Board. 2015. "NTSB Press Release." *NTSB Calls for Immediate Action on Collision Avoidance Systems for Vehicles; Cites Slow Progress as Major Safety Issue.* Washington: National Transportation Safety Board Office of Public Affairs, June 8.

Noh, Samyeul, and Kyounghwan An. 2018. "Decision-Making Framework for Automated Driving in Highway Environments." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 19 (1): 58-71.

Novak, Petr, Petr Kadera, and Manuel Wimmer. 2017. "Agent-Based Modeling and Simulation of Hybrid Cyber-Physical Systems." *IEEE Conference Publishing* 1-8.

NTSB. 2015. *Derailment of Amtrak Passenger Train 188 Philadelphia, Pennsylvania May 12, 2015.* Accident, Washington: National Transportation Safety Board.

NTSB. 2014. *Descent Below Visual Glidepath and Impact With Seawall Asiana Airlines Flight 214 Boeing 777-200ER, HL7742 San Francisco, California July 6, 2013.* Accident, National Transportation Safety Board, Washington: National Transportation Safety Board.

NTSB. 2017. *PRELIMINARY REPORT RAILROAD Amtrak Passenger Train 501 Derailment.* Accident, Washington: National Transportation Safety Board.

NTSB. 2015. *The Use of Forward Collision Avoidance Systems to Prevent and Mitigate Rear-End Crashes.* Special Investigation Report, Washington, D.C.: National Transportation Safety Board, 63.

Ohta, Hiroki, Naoki Akai, Eijiro Takeuchi, Shinpei Kato, and Masato Edahiro. 2017. "Pure Pursuit Revisited: Field Testing of Autonomous Vehicles in Urban Areas." *Cyber-Physical Systems, Networks, and Applications.* Nagoya: IEEE. 7-12.

2015. "On Curve Negotiation: From Driver Support to Automation." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 2082-2093.

Palmer, James D, Andrew P Sage, Thomas B Sheridan, Michael H Smith, and James M Tien. 2003. "The IEEE Systems, Man, and Cybernetics Society: Historical Development, Current Status, and Future Perspectives." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS* 33 (1): 13-23.

Parasuraman, Raja, Thomas B Sheridan, and Christopher D Wickens. 2000. "A Model for Types and Levels of Human Interaction with Automation." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS* 286-297.

Parnell, Gregory S, Patrick J Driscoll, and Dale L Henderson. 2011. *Decision Making in Systems Engineering and Management.* Hoboken: John Wiley & Sons, Inc.

Petit, William A. 2009. "Interoperable Positive Train Control." *IEEE VEHICULAR TECHNOLOGY MAGAZINE* 27-34.

Popken, Luitjens. 2007. "Rescuing the Huygens Mission From Fiasco." *Proceedings of the IEEE* (IEEE) 95 (11): 2248-2258.

Quintas, Joao, Paulo Menezes, and Jorge Dias. 2017. "Information Model and Architecture Specification for Context Awareness Interaction Decision Support in Cyber-Physical Human–Machine Systems." *IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS* 47 (3): 323-331.

Reinholtz, Kirk, and Keyur Patel. 2008. "Testing Autonomous Systems for Deep Space Exploration." *IEEE A&E SYSTEMS MAGAZINE* 22-27.

Robertson, Charles L. 2010. *Determining Appropriate Levels of Automation.* FITS SRM Automation Management Research, Bismarck : UNIVERSITY OF NORTH DAKOTA.

Rouse, Sandra H, William B Rouse, and John M Hammer. 1982. "Design and Evaluation of an Onboard Computer-Based Information System for Aircraft." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS* SMC-12 (4): 451-463.

Rovere, Santiago L, Michael J North, Guillermo P Podesta, and Federico E Bert. 2016. "Practical Points for the Software Development of an Agent-Based Model of a Coupled Human-Natural System." *IEEE Access* 4: 4282-4298.

Ruff, Heath A, S Narayanan, and Mark H Draper. 2002. "Human Interaction with Levels of Automation and Decision-Aid Fidelity in the Supervisory Control of Multiple Simulated Unmanned Air Vehicles." *PRESENCE* 335-351.

SAE INTERNATIONAL. 2014. "Automated Driving: Levels of Driving Automation are Defined in New SAE International Standard J3016." SAE INTERNATIONAL.

Safety, Board National Transportation. 2015. *The Use of Forward Collision Avoidance Systems to Prevent and Mitigate Rear-End Crashes.* Special Investigation Report, Washington: NTSB.

Sanders, Mark S, and Ernest J McCormick. 1993. *Human Factors in Engineering and Design.* New York: McGraw-Hill, Inc.

Schnelle, Scott, Junmin Wang, Hai-Jun Su, and Richard Jagacinski. 2017. "A Personalizable Driver Steering Model Capable of Predicting Driver Behaviors in Vehicle Collision Avoidance Maneuvers." *IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS* 47 (5): 625-635.

Shannon, Claude E. 1948. "A Mathematical Theory of Communication." *The Bell System Technical Journal*, July 1: 379-423.

Sheng, Guo, Xinya Zhao, Hangdong Zhang, Lv Zhihan, and Song Houbing. 2017. "Mathematical Models for Simulating Coded Digital Communication: A Comprehensive Tutorial by Big Data Analytics in Cyber-Physical Systems." *IEEE Access* 1-9.

Sheridan, Thomas B. 2002. *Humans and Automation: System Design and Research Issues.* Santa Monica: John Wiley & Sons, Inc.

Sheridan, Thomas S. 2011. "Adaptive Automation, Level of Automation, Allocation Authority, Supervisory Control, and Adaptive Control: Distinctions and Modes of Adaptation." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS* 662-667.

Sheridan, Tom. 2017. "Recollections from the Early Days of SMC." *IEEE Systems, Man, & Cybernetics Magazine* (IEEE) 25.

Shi, Dawei, Robert J Elliott, and Tongwen Chen. 2017. "On Finite-State Stochastic Modeling and Secure Estimation of Cyber-Physical Systems." *IEEE TRANSACTIONS ON AUTOMATIC CONTROL* 62 (1): 65-80.

Shneiderman, Ben. 2007. "Human Responsibility for Autonomous Agents." *IEEE INTELLIGENT SYSTEMS* 60-61.

Smets, Nanja J, Jurriaan van Diggelen, Mark A Neerincx, Jeffrey M Bradshaw, Catholijn M Jonker, Lennard de Rijk, Pieter A Senster, Otten Thije, and Maarten Sierhuis. 2010. "Assessing Human-Agent Teams for Future Space Missions." *IEEE INTELLIGENT SYSTEMS* (IEEE Computer Society) 46-53.

statista. 2017. *statista: The Statistics Portal.* May 23. Accessed March 11, 2018. https://www.faa.gov/news/updates/?newsId=84384.

Sun, Jiayi, Jun Tang, and Songyang Lao. 2017. "Collision Avoidance for Cooperative UAVs with Optimized Artificial Potential Field Algorithm." *IEEE Access* 5: 18382-18390.

Thalassinakis, Emmanuel j, Evangelos N Dialynas, and Demosthenes Agoris. 2006. "Method Combining ANNs and Monte Carlo Simulation for the Selection of the Load Shedding Protection Strategies in Autonomous Power Systems." *IEEE TRANSACTIONS ON POWER SYSTEMS* 1574-1582.

Trembley, Daniel K. 2018. "A Survey of Advanced Manufacturing with Legacy Machinery: The Internet of Other Things." *Proceedings of the 2018 IISE Annual Conference.* Norcross: IISE. 1-6.

Turner, P. R. 1985. "Autonomy and Automation for Space Station Housekeeping and Maintenance Functions." *Journal of Engineering for Industry* (ASME) 107 (1): 39-42.

Tweedale, Jeffrey W, and Lakhmi C Jain. 2011. *Embedded Automation in Human-Agent Environment.* Vol. 10. Berlin: Springer.

USAF. 2008. *Aircraft Accident Investigation: B-2A T/N 89-0127.* Accident, Andersen Air Force Base: United State Air Force.

Vahidi, Ardalanand, and Azim Eskandarian. 2003. "Research Advances in Intelligent Collision Avoidance and Adaptive Cruise Control." *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 4 (3): 143-153.

Von Bertalanffy, Ludwig. 1974. *Robots, Men, and Minds: Psychology in the Modern World.* New York: George Braziller.

Wan, Jjiang, Arquimedes Canedo, and Mohammad A Faruque. 2017. "Functional Model-Based Design Methodology for Automotive Cyber-Physical Systems." *IEEE SYSTEMS JOURNAL* 11 (4): 2028-2039.

Wang, Chun-Sheng, and Min Wu. 2013. "Hierarchical Intelligent Control System and Its Application to the Sintering Process." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 9 (1): 190-197.

Wang, Yue. 2016. "Regret-Based Automated Decision-Making Aids for Domain Search Tasks Using Human-Agent Collaborative Teams." *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY* 1680-1695.

Wehrmeister, Marco A, Carlos P Eduardo, and Franz J Rammig. 2013. "Aspect-Oriented Model-Driven Engineering for Embedded Systems Applied to Automation Systems." *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS* 9 (4): 2373-2386.

Wiener, Norbert. 1948. *Cybernetics: Control and Communication in the Animal and the Machine.* Cambridge: MIT Press.

Wilkins, Maurice J. 2003. "Seeking the Appropriate Level of Automation." *IEEE COMPUTING AND CONTROL ENGINEERING* 43-44.

Xin, Shujun, Qinglai Guo, Hongbin Sun, Boming Zhang, Jianhui Wang, and Chen Chen. 2015. "Cyber-Physical Modeling and Cyber-Contingency Assessment of Hierarchical Control Systems." *IEEE TRANSACTIONS ON SMART GRID* 6 (5): 2375-2385.

Xu, Z. and et al. 2016. "Multinominal Latent Logistic Regression for Image Processing." *IEEE Transactions on Image Processing* 973-987.

Yarmoluk, Dan. 2017. "The Industrial IoT can Monitor Critical Machinery - Measurement can change behavior, lead to new business models." *ISE Magazine*, February: 43-46.

Young, Stuart H, Thomas A Mazzuchi, and Shahram Sarkani. 2017. "A Framework for Predicting Future System Performance in Autonomous Unmanned Ground Vehicles." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS* 47 (7): 1192-1206.

Zadeh, Lotfi A. 2008. "Toward Human Level Machine Intelligence - Is It Achievable? The Need for a Paradigm Shift." *IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE* 11-22.

Zhang, Dan, Haiyu Song, and Li Yu. 2017. "Robust Fuzzy-Model-Based Filtering for Nonlinear Cyber-Physical Systems With Multiple Stochastic Incomplete Measurements." *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS* 47 (8): 1826-1838.

Zhang, Kun, Jonathan Sprinkle, and Ricardo G Sanfelice. 2016. "Computationally Aware Switching Criteria for Hybrid Model Predictive Control of Cyber-Physical Systems." *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING* 13 (2): 479-490.