Massachusetts Institute of Technology

Common Mistakes in STPA and CAST

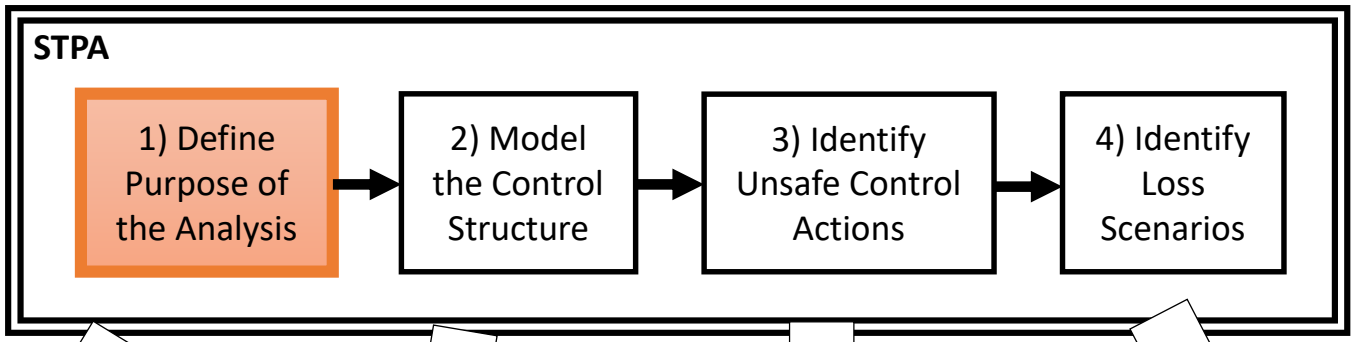
Dr. John Thomas

March 2019

Revised October 2019

Any questions? Please email: jthomas4@mit.edu

System-Theoretic Process Analysis (STPA)

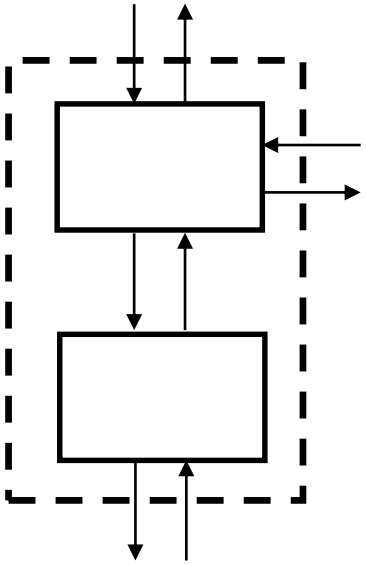


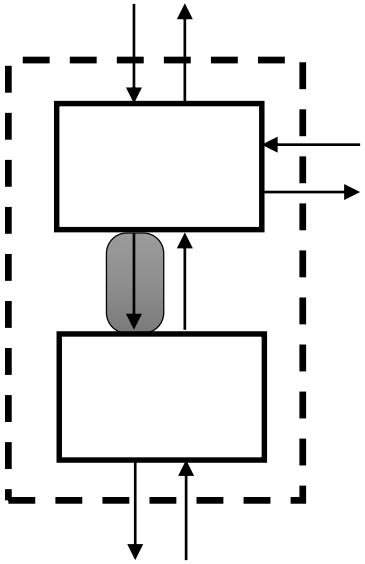
Identify Losses, Hazards

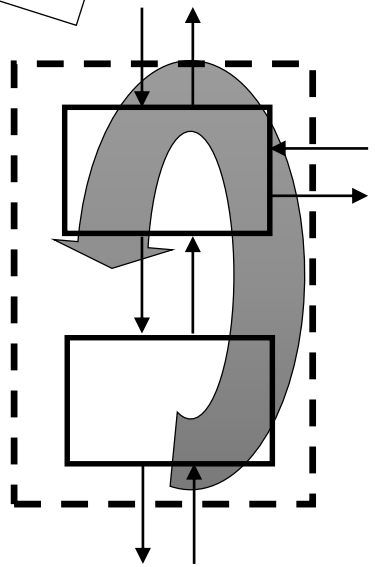
Define System boundary

Environment

System







Losses,
System-level Hazards

Incorrect Losses

- Loss of brake pressure
- Loss of engine RPM
- Loss of pressurizer pressure
- …

Incorrect System-level Hazards

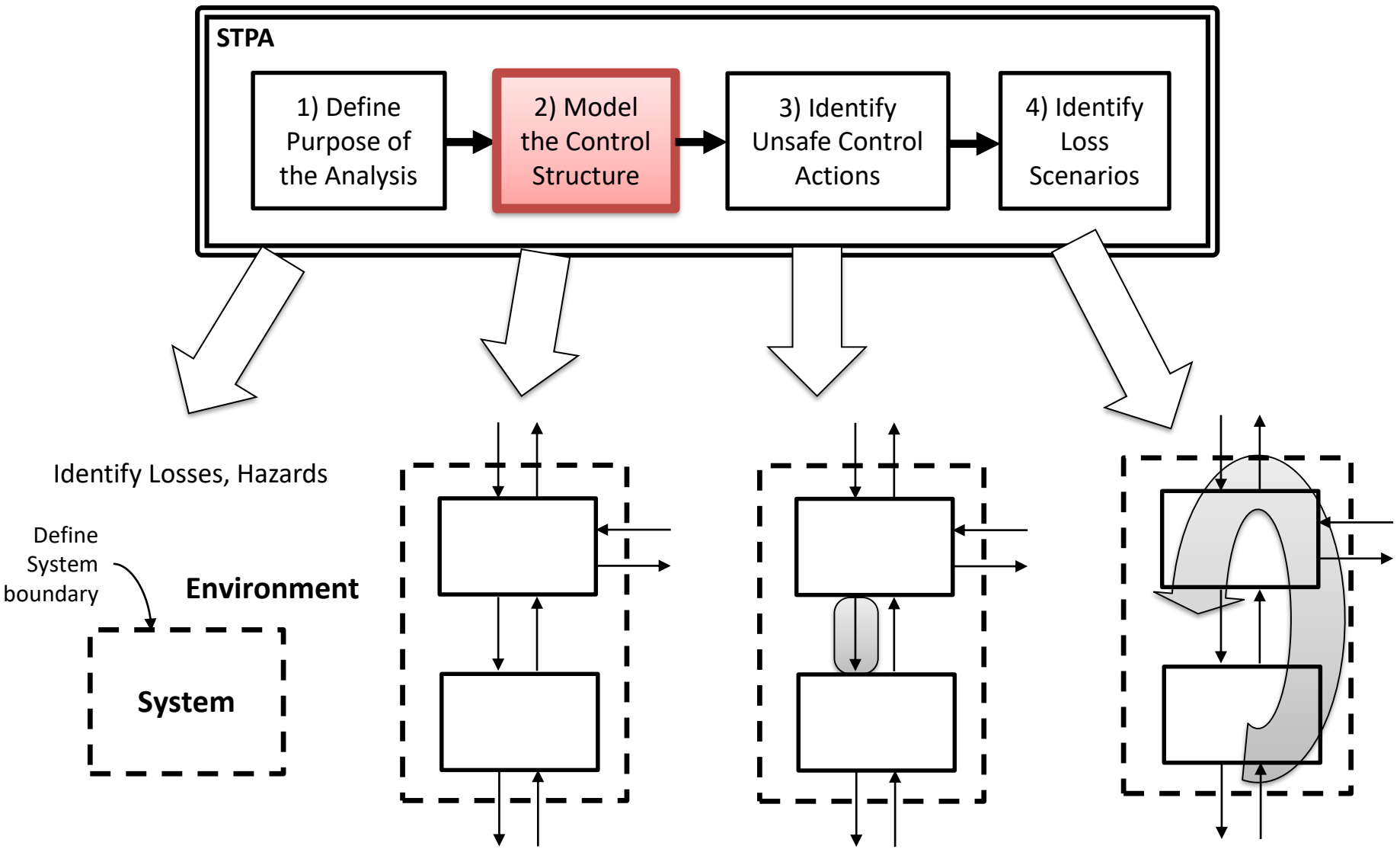
- Engine Flameout
- Cruise control does not notify driver of oncoming car
- Improper use of cruise control by driver
- Transmission controller reports incorrect gear to driver

Tips to prevent common mistakes when identifying hazards

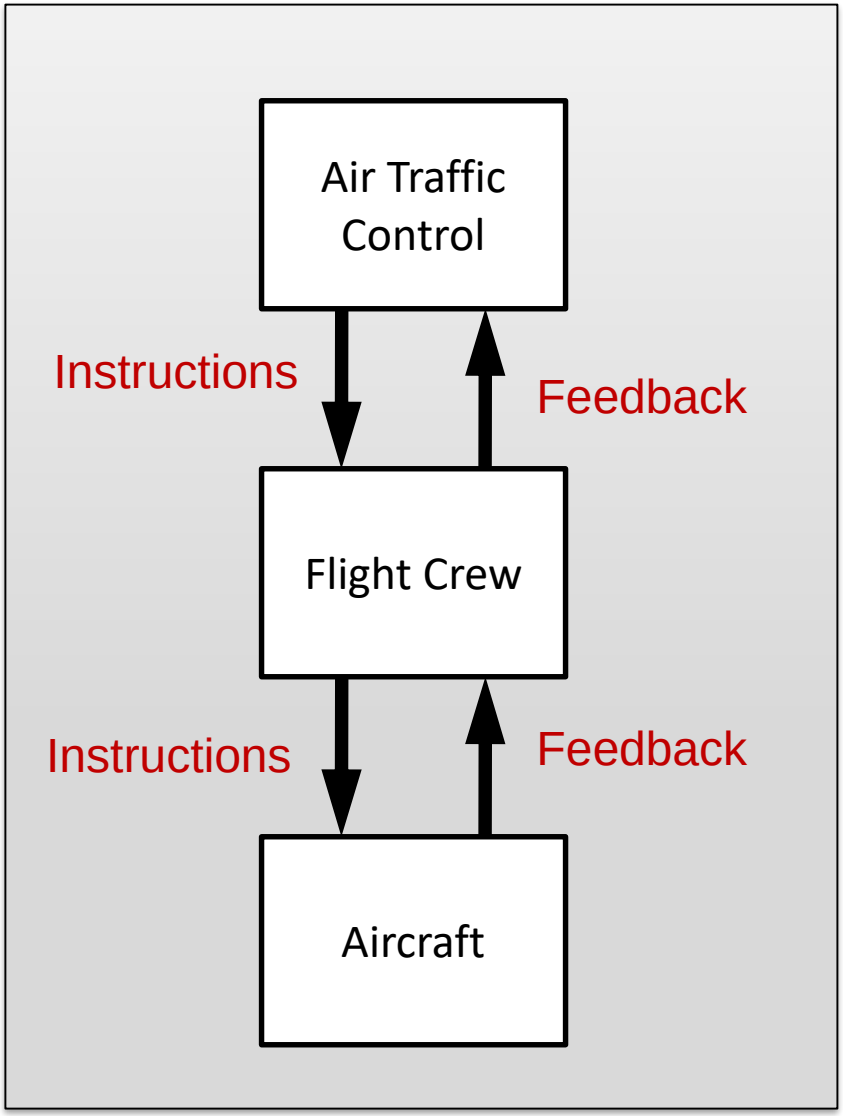
- Hazards should not refer to individual components of the system
- All hazards should refer to the overall system and system state
- Hazards should refer to factors that can be controlled or managed by the system designers and operators
- All hazards should describe system-level conditions to be prevented
- The number of hazards should be relatively small, usually no more than 7 to 10
- Hazards should not include ambiguous or recursive words like “unsafe”, “unintended”, “accidental”, etc.

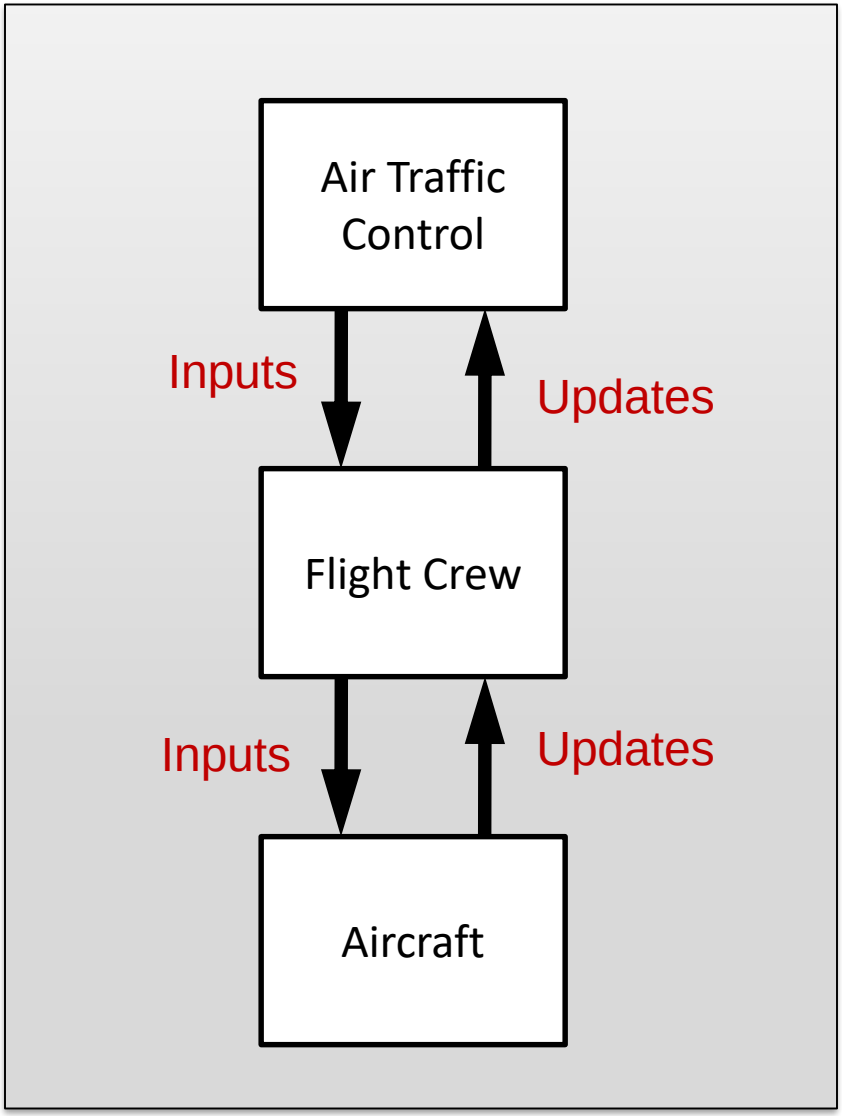
STPA Handbook

System-Theoretic Process Analysis (STPA)

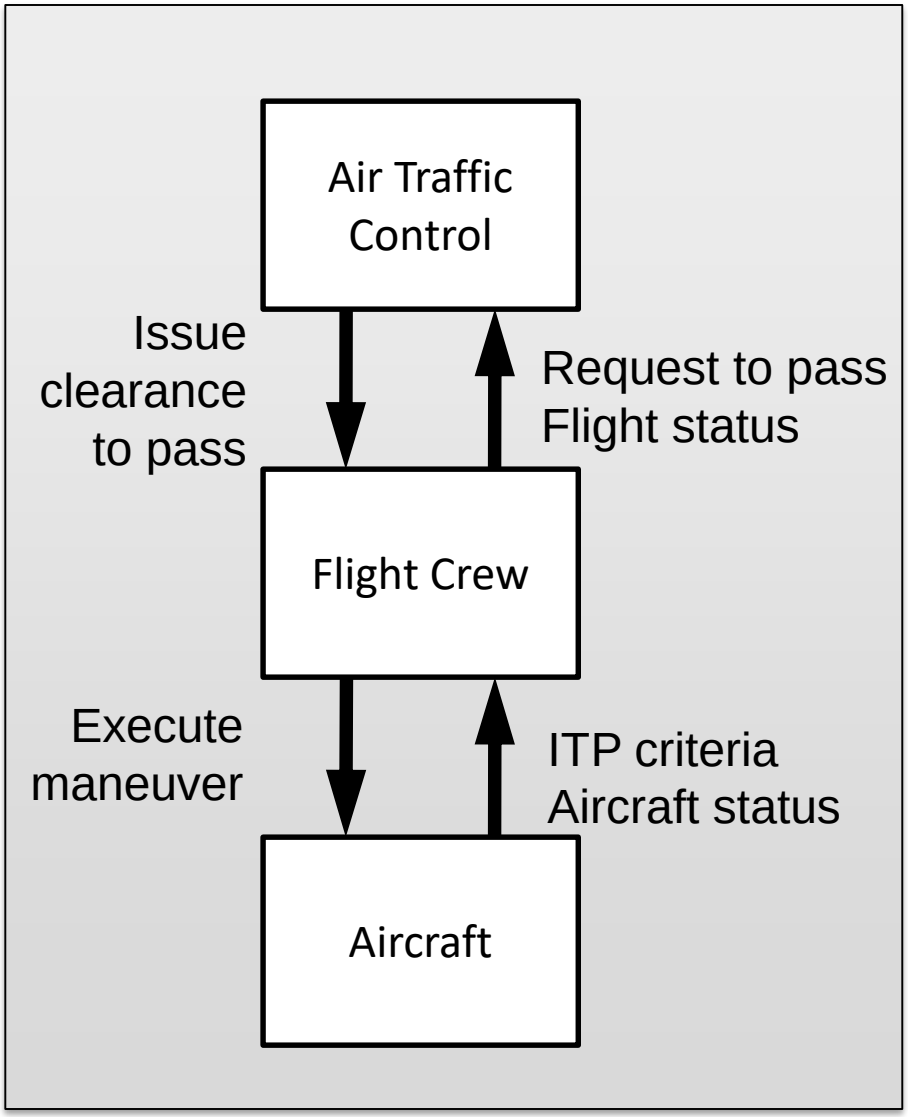


Control Structure that is too vague



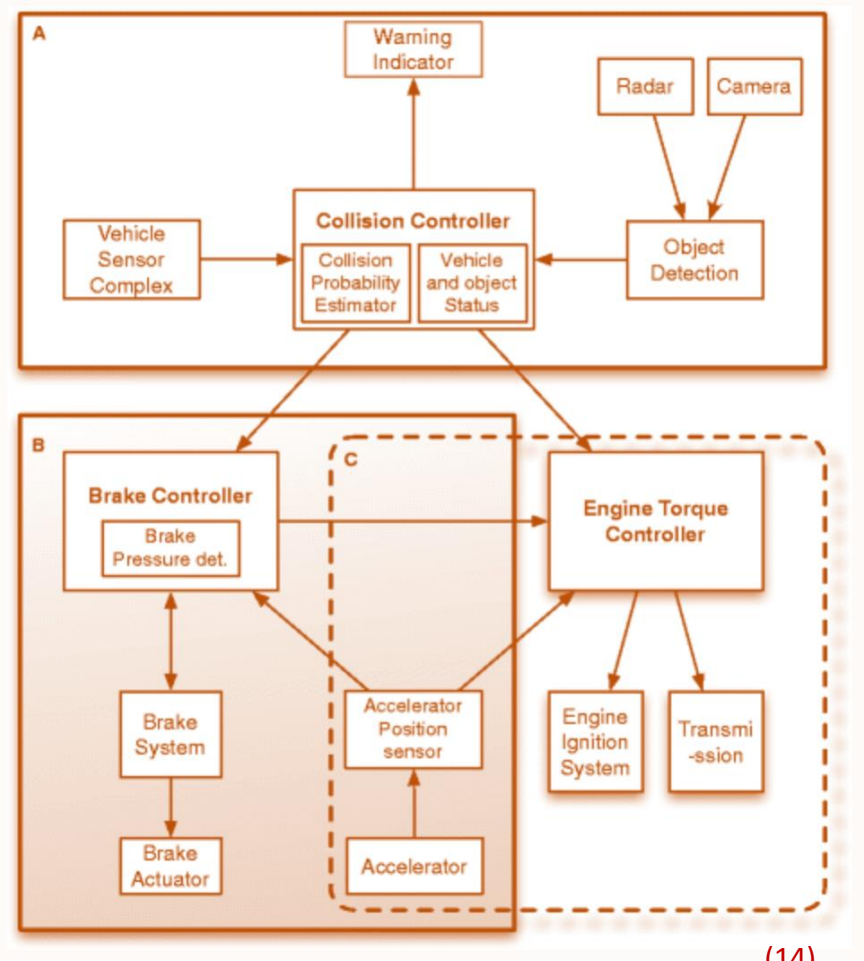


Better High-level Control Structure



- Note that “High-level” does not have to be vague!

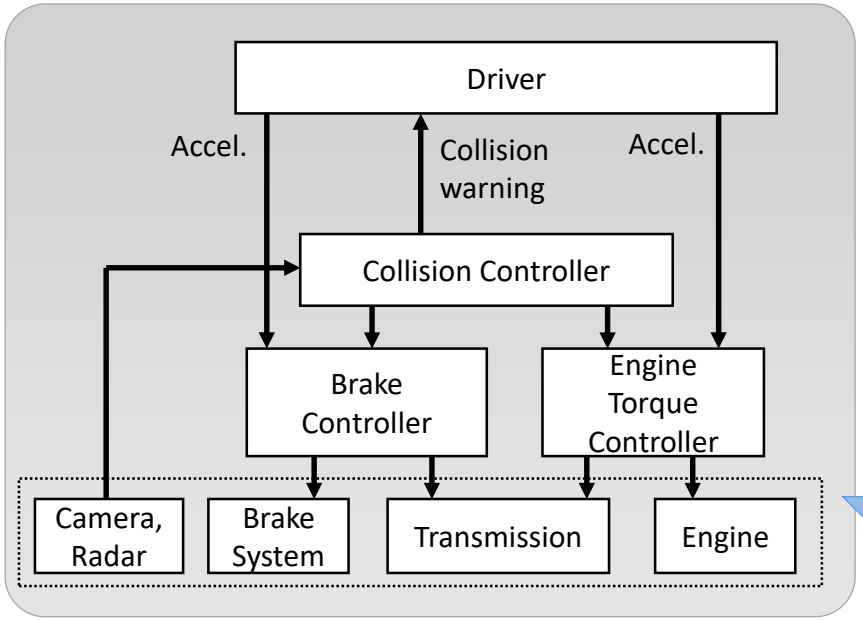
Incorrect control structure



(14)

- Missing or inconsistent control hierarchy
- Driver cmds, but no driver
- Sensors and actuators with no controller
- Controlled process?
- Control loops?

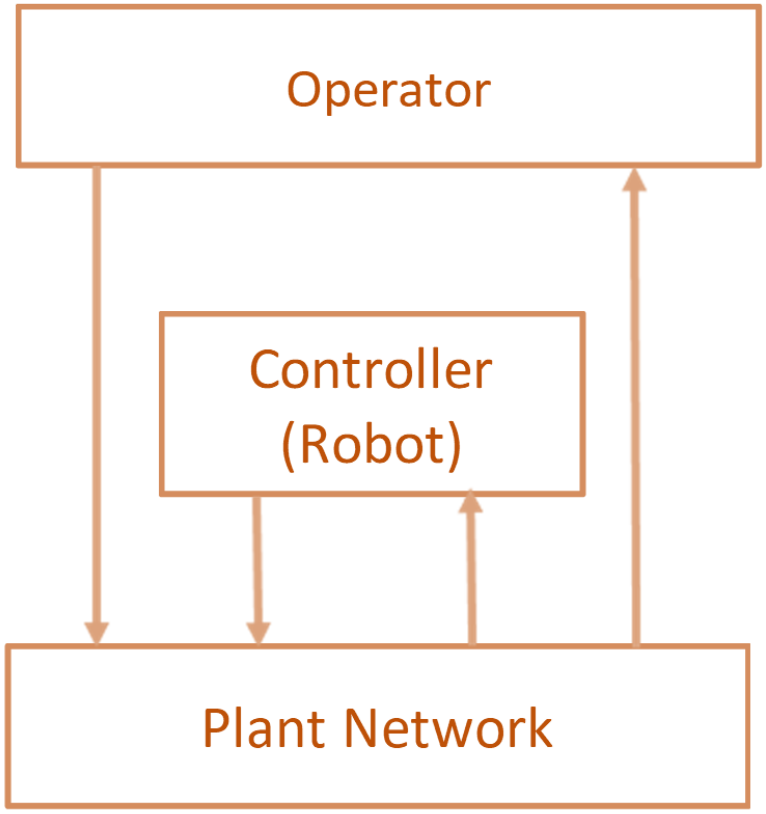
Better control structure (but incomplete)



(8)

- Defined control hierarchy
- Driver is included

Incorrect control structure



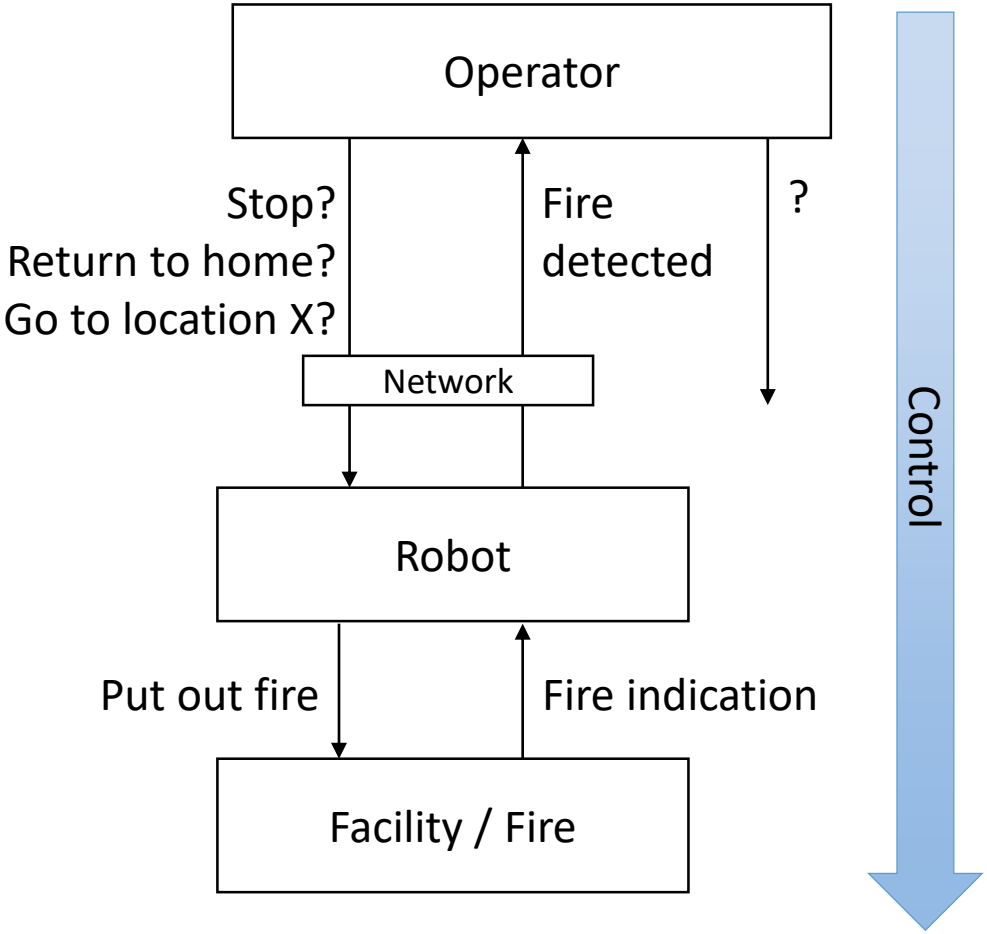
Control hierarchy?

Is the network really the ultimate controlled process

No commands to the Robot?

What are the commands/feedback?

Better control structure (but incomplete)

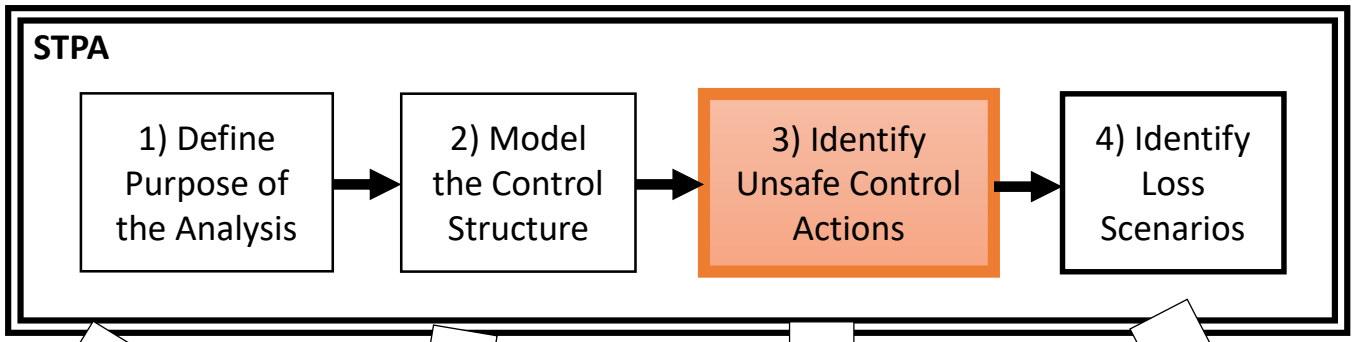


Properly defined control hierarchy

Controlled process is the facility/fire

Network is a “pass-through”, not generating its own control actions

System-Theoretic Process Analysis (STPA)

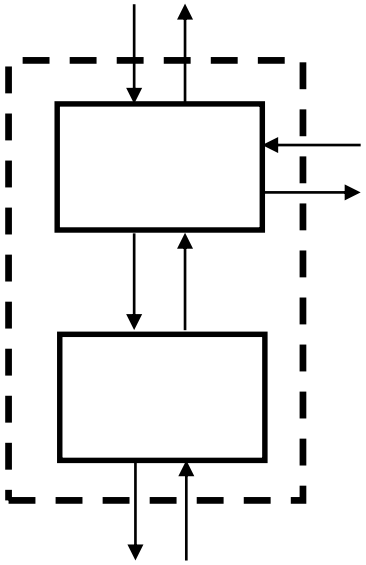


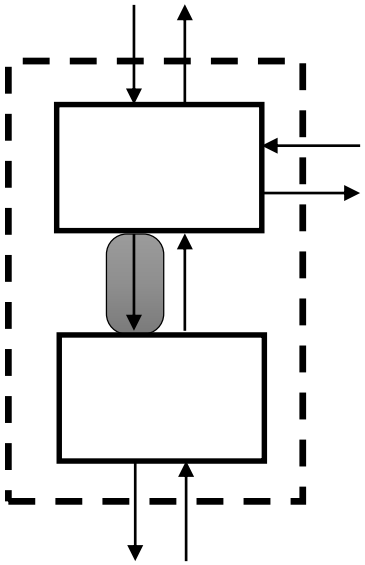
Identify Losses, Hazards

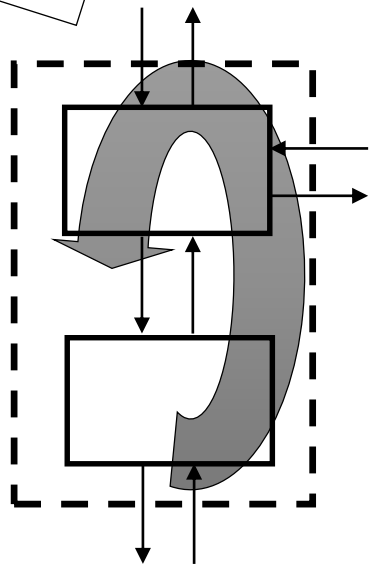
Define System boundary

Environment

System







Incorrect UCAs (Unsafe Control Actions)

- Pilot fails to recognize TCAS alert
- Does not monitor emergency brake operation
- Decreases funding

"Fails"
"Recognize"
"Monitor"
Missing action
Missing context

Better UCA

- UCA-1:

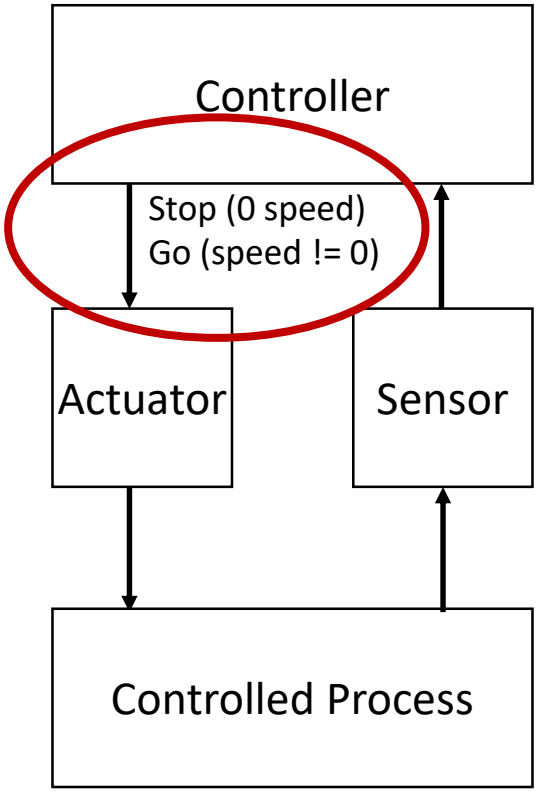
  Pilot
  does not provide
  pitch up cmd
  when TCAS provides climb TA
  [H-1]

Includes all necessary UCA elements:
- Source controller
- Type
- Control Action
- Context
- Traceability to hazards

Tips for Specifying Unsafe Control Actions

- Start every UCA with the source controller
- A UCA is not just a statement about the state of a component
- A UCA is not just a statement about the outcome
- A UCA should include an observable output of the controller (an action or inaction)
  - Not a thought or a process like "monitoring“ or “recognizing”.
  - Look at arrows on the control structure
- Do not use the word “fail” in a UCA
  - These are not necessarily failures. They may or may not be caused by failures, and we may not know all the causes when STPA Step 3 is performed.

Incorrectly interpreting UCA

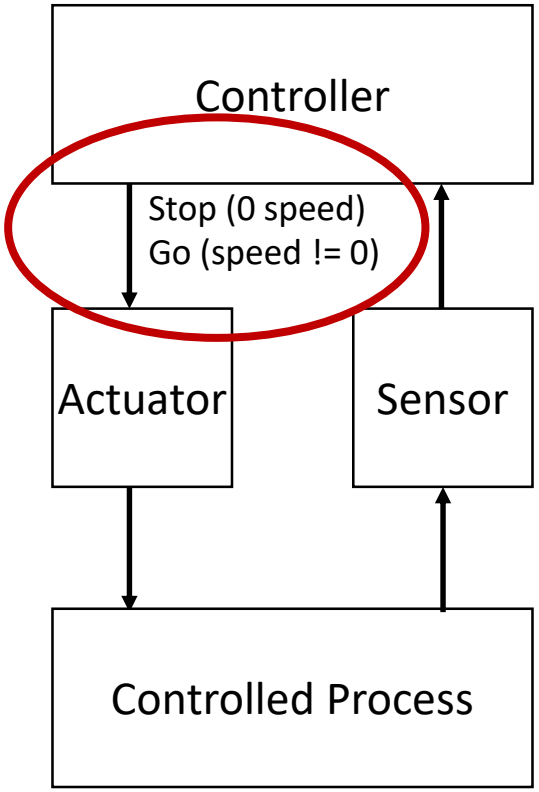


	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Go Cmd	Controller does not provide Go cmd when _____	Controller provides Go cmd when obstacle is in path	…	…
Stop Cmd	Controller does not provide Stop cmd when obstacle is in path	Controller provides Stop cmd when _____	…	…

"It must have provided the Go command"

Don't assume the Go cmd must have been provided. Maybe, maybe not.
It could also be that no cmd was provided.

Incomplete UCAs



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Go Cmd	Controller does not provide Go cmd when _____	Controller provides Go cmd when obstacle is in path	…	…
Stop Cmd	Controller does not provide Stop cmd when obstacle is in path	Controller provides Stop cmd when _____	…	…

Other UCAs are missing. What about:

- … Provides go with excessive speed…
- … Provides go with insufficient speed…
- … Provides go in opposite direction…
- … Provides go in unstable way (e.g. rapidly changing speed) …

“This research found that STPA was weaker on system failures: [link]”

A UCA contains five parts:

UCA-2: BSCU Autobrake <Source> provides <Type> Brake command <Control Action> during a normal takeoff <Context> [H-4.3] <Link to Hazards>

No.	Command or Event	Not Provided	Provided Unsafe	Provided			Stopped Too Soon
				Too Early	Too Late	Out of Sequence	
1	Vehicle Status Signal	Catastrophic- (Wrong brake pressure determination) [1a]	Catastrophic- (Wrong brake pressure determination) [1a]	N/A	Catastrophic- (Wrong brake pressure determination and wrong reaction time) [1a]	N/A	N/A
2	Object Status Signal	Catastrophic- (Wrong brake pressure determination) [2a]	Catastrophic- (Wrong brake pressure determination) [2a]	N/A	Catastrophic- (Wrong brake pressure determination and wrong reaction time) [2a]	N/A	N/A

Table 1. Inadequate Control Commands/Events

Published March 2019

"This research found that STPA was weaker on system failures: [link]"

HAZARD ANALYSIS

For hazard analysis the detailed control structure diagram of the system was acquired. Next, the first and the second author of this study analyzed the forward collision avoidance system and identified 14 inadequate control commands or events, including their causal factors. The results (both inadequate control commands or events and their causal factors) were analyzed and reviewed by the third and the fourth author. In this study, the authors have performed hazard analysis of the forward collision avoidance system by following their best interpretation/understanding of the STPA guidelines as presented by Leveson (2012) and Leveson et al. (2012). Table 1 shows an excerpt of the identified inadequate control commands or events[1] that could lead to hazardous states.

No.	Command or Event	Not Provided	Provided Unsafe	Provided			Stopped Too Soon
				Too Early	Too Late	Out of Sequence	
1	Vehicle Status Signal	Catastrophic- (Wrong brake pressure determination) [1a]	Catastrophic- (Wrong brake pressure determination) [1a]	N/A	Catastrophic- (Wrong brake pressure determination and wrong reaction time) [1a]	N/A	N/A
2	Object Status Signal	Catastrophic- (Wrong brake pressure determination) [2a]	Catastrophic- (Wrong brake pressure determination) [2a]	N/A	Catastrophic- (Wrong brake pressure determination and wrong reaction time) [2a]	N/A	N/A

Table 1. Inadequate Control Commands/Events

- STPA Steps 1 & 2?
- Incorrect STPA Step 3
- STPA Step 4?

Conclusions despite mistakes

- "STPA has proved to be an effective and efficient hazard analysis method“
- "With regard to software error type hazards, STPA found more hazards than FMEA of unique hazards“
- "STPA considers more types of hazard causes than the other traditional hazard analysis methods. Therefore, STPA is more complete than existing traditional hazard analysis methods"

Command /event	Not provided	Provided unsafe		Provided …		Stopped too soon
			Too early	Too late	Out of seq.	
Object detection signal	Catastrophic-system dysfunction [collision] (1a)	Catastrophic-system malfunctioning (1b)	N/A	Catastrophic-system dysfunction [collision] (1a)	N/A	N/A
Vehicle complex signal	Catastrophic-problem in calculation of vehicle status and collision probability (2a)	Catastrophic-problem in calculation of vehicle status and collision probability (2a)	N/A	Catastrophic-problem in calculation of vehicle status and collision probability (2a)	N/A	N/A
Collision warning signal	Negligible (if every thing is working properly, then the active safety will be saved from collision) (3a)	N/A	Negligible (if every thing is working properly, then the active safety will be saved from collision) (3a)	Negligible (if every thing is working properly, then the active safety will be saved from collision) (3a)	N/A	Negligible (warning will be stopped too soon that can cause accident. If everything works properly, then the active safety will be saved from collision) (3b)

Incorrect unsafe control actions

Incorrect UCAs

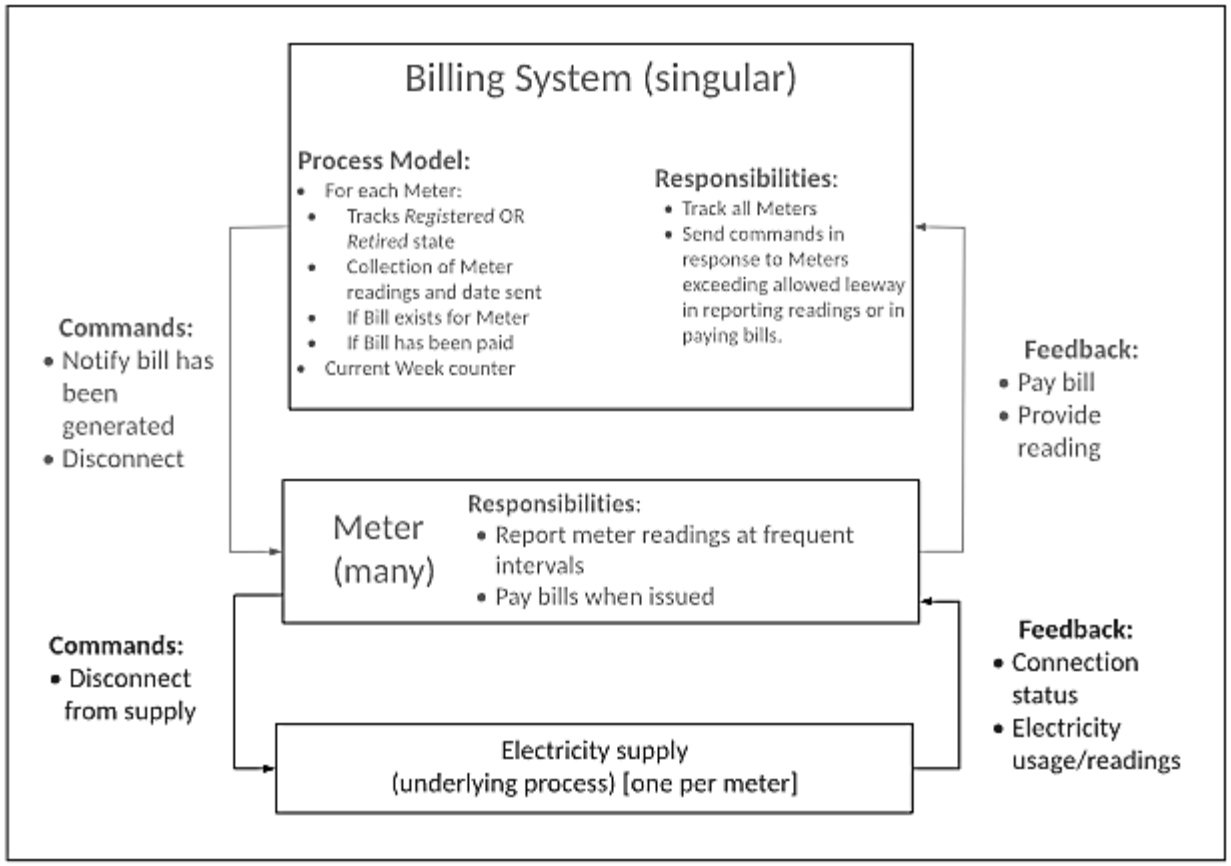
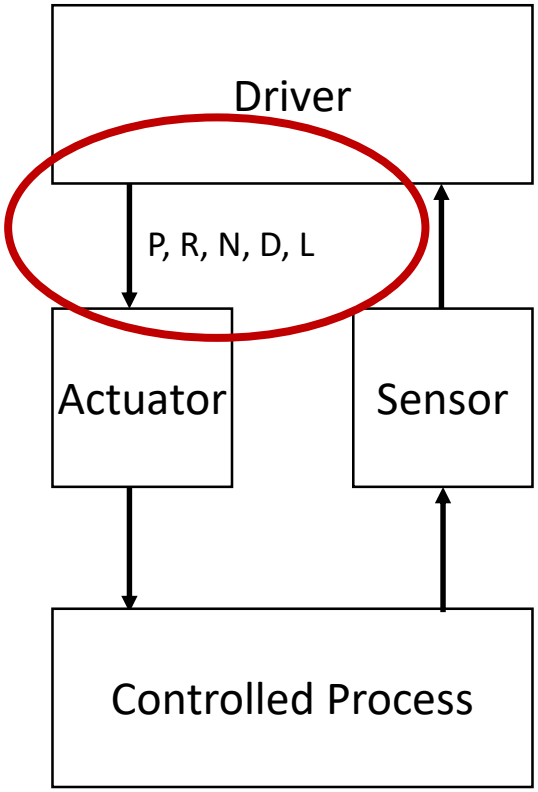


Figure 1: Functional control structure for smart meter example.

Control Action	*Is issued*	*Is not issued*	*Is issued out of sequence*	*Is issued for incorrect duration*
Register Meter	An invalid meter is re-registered.	A meter fails to be registered.	A meter is registered multiple times.	N/A - registration is discrete.

Table 2: Control action analysis results

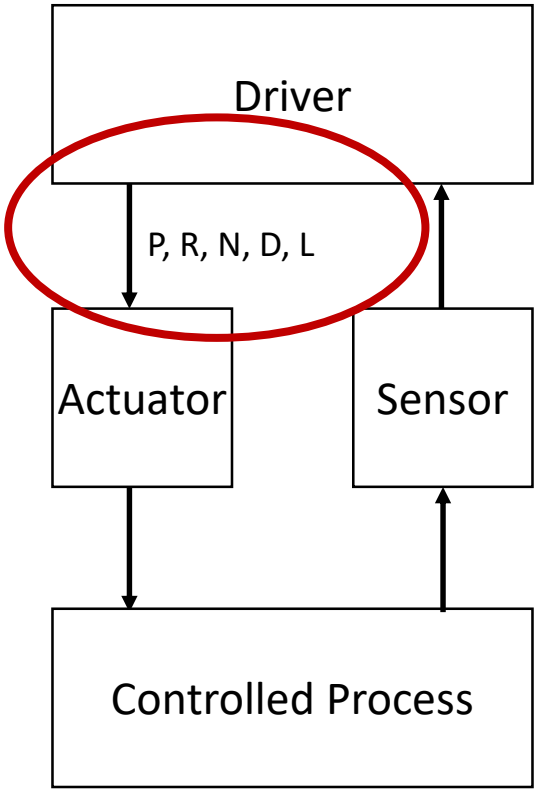
Incorrect UCAs



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Park Cmd	Driver does not provide Park Cmd	Driver provides Park Cmd erroneously	…	…
Reverse Cmd	Driver does not provide Reverse Cmd when not needed	Driver provides Reverse Cmd by mistake	…	…

- UCA must specify the context that makes the control action unsafe
- What does “erroneously” mean? What makes it unsafe?

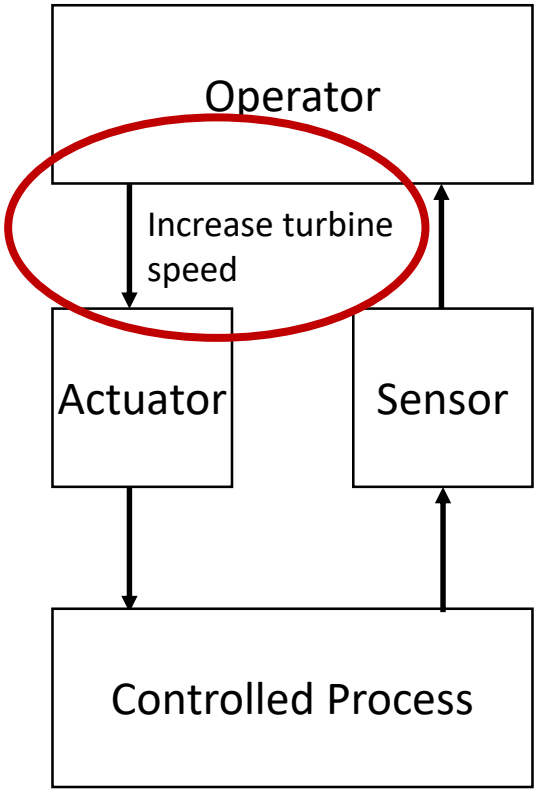
Indirect context



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Park Cmd	…	Driver provides Park when they incorrectly believe vehicle is stopped	…	…
Drive Cmd	Controller does not provide Stop cmd when _____	Controller provides Stop cmd when _____	…	…

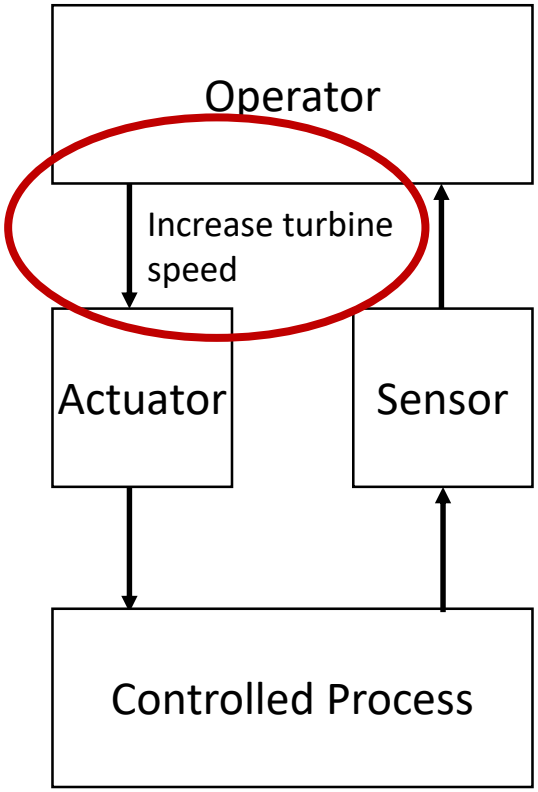
- Controller beliefs belong in another step
- Ask: what is the condition that makes the park command itself unsafe?

Vague context, assumptions



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Increase turbine speed	Operator does not provide increase turbine speed cmd when required	…	…	…

Defining UCAs relative to procedures

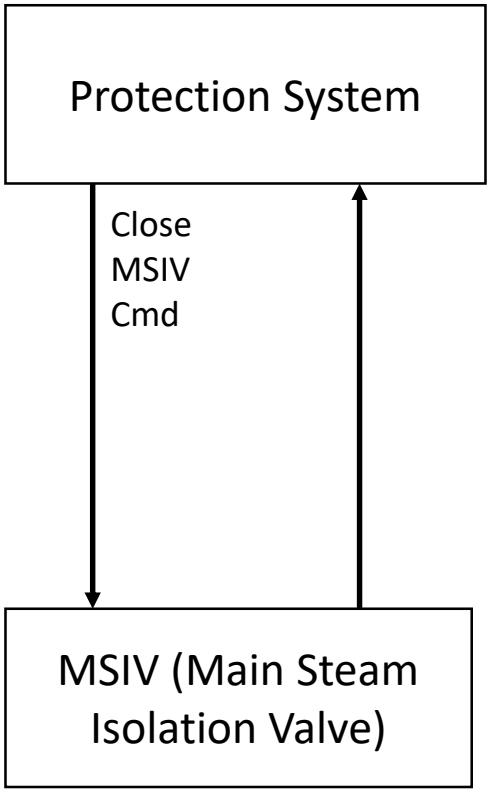


	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Increase turbine speed	…	Operator provides increase turbine speed cmd when procedure specifies decreasing	…	…

STPA does not assume the existing procedure is fully correct and complete. Better UCA:

- Operator provides increase turbine speed cmd when turbine speed exceeds X rpm

Confusing UCAs with Failure Effects



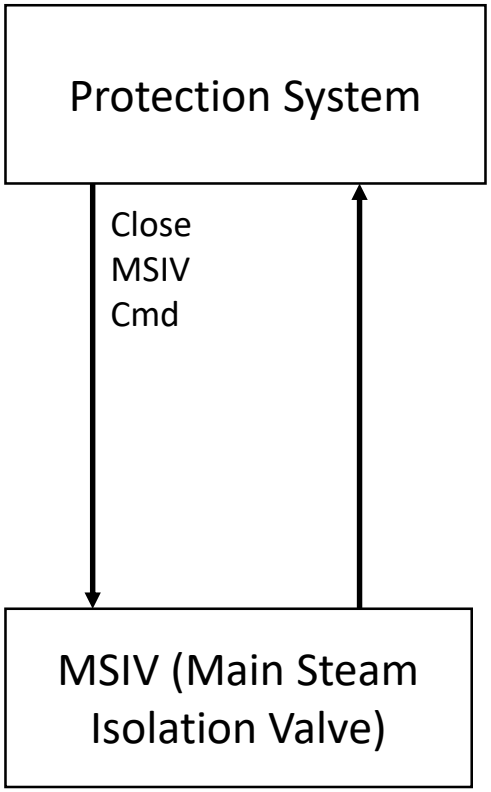
	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Close MSIV Cmd	…	Contamination in secondary cooling Turbine damage	…	…

Are these correct? Hard to review. These were reviewed incorrectly.

Tips:

- UCAs are control actions in a context that makes them unsafe
- UCAs are not just effects
- UCAs are not just hazardous states
- UCA contexts might be non-hazardous without the control action.

Confusing UCA contexts with hazardous states



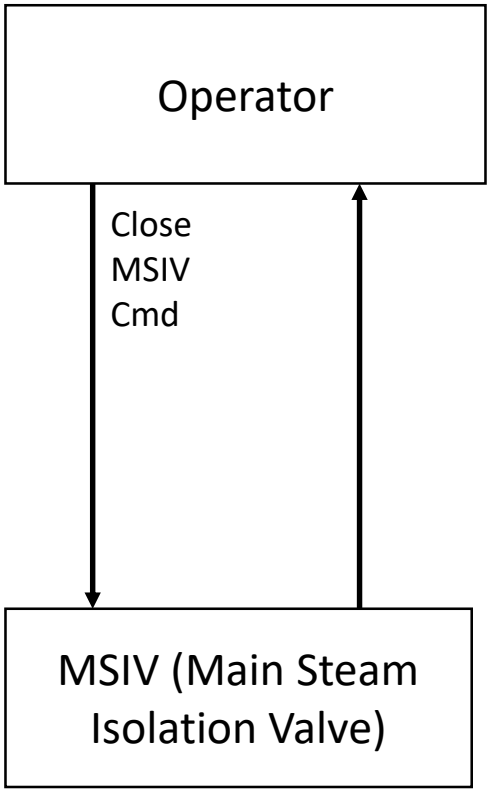
	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Close MSIV Cmd	Steam Generator Tube Ruptures	Secondary cooling systems failed	…	…

Potential confusion: UCA contexts are not simply the hazardous states.

A UCA is an action that is unsafe in some context. Confusion can be avoided by writing whole UCA.

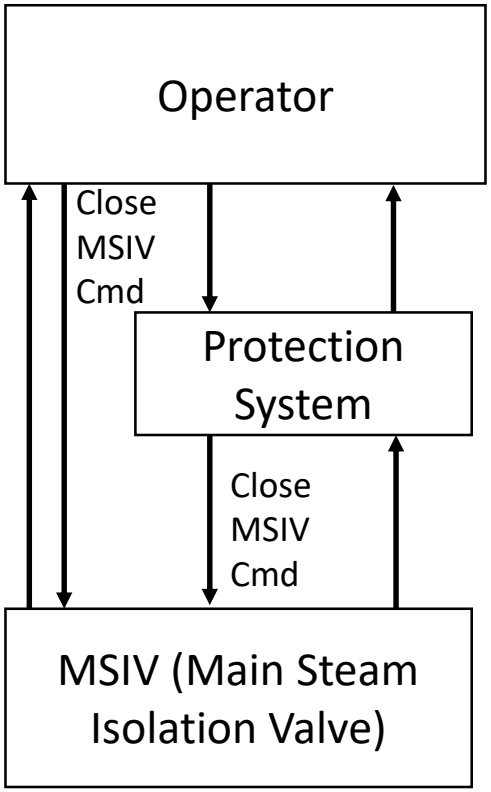
UCA-1: Protection System does not provide Open MSIV Cmd when Steam Generator Tube Ruptures [H-1,2]

Potential confusion



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Close MSIV Cmd	when SGTR and other cooling systems not operational	…	…	…

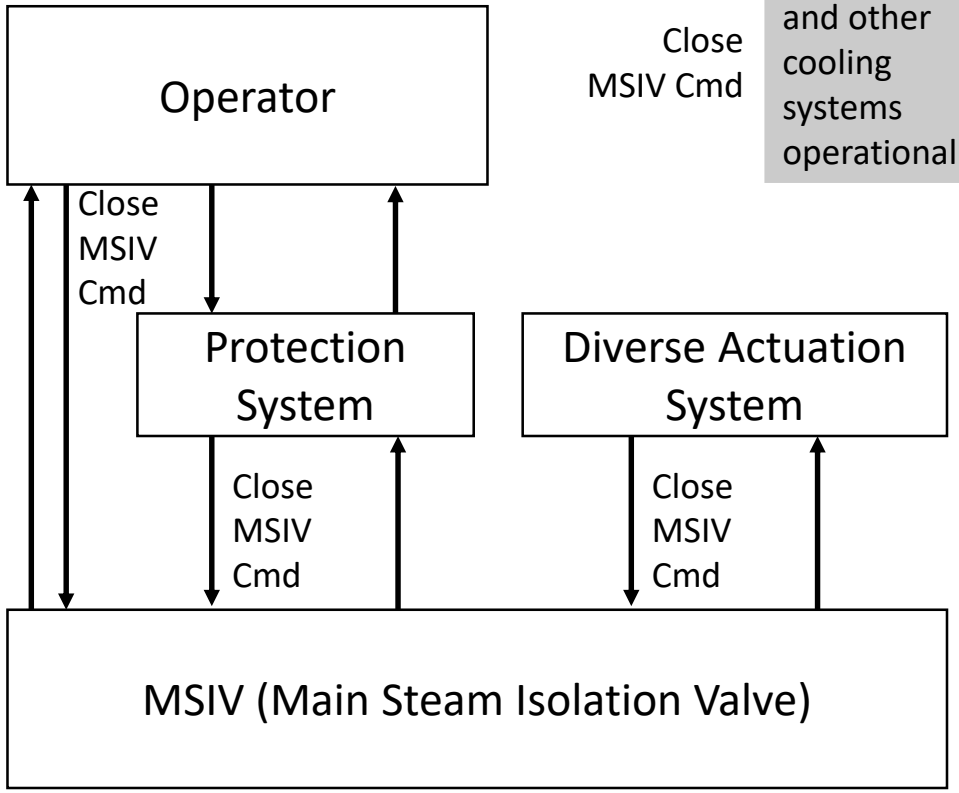
Potential confusion



	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Close MSIV Cmd	when SGTR and other cooling systems operational	…	…	…

Confusing control actions from multiple controllers

	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Close MSIV Cmd	when SGTR and other cooling systems operational	…	…	…



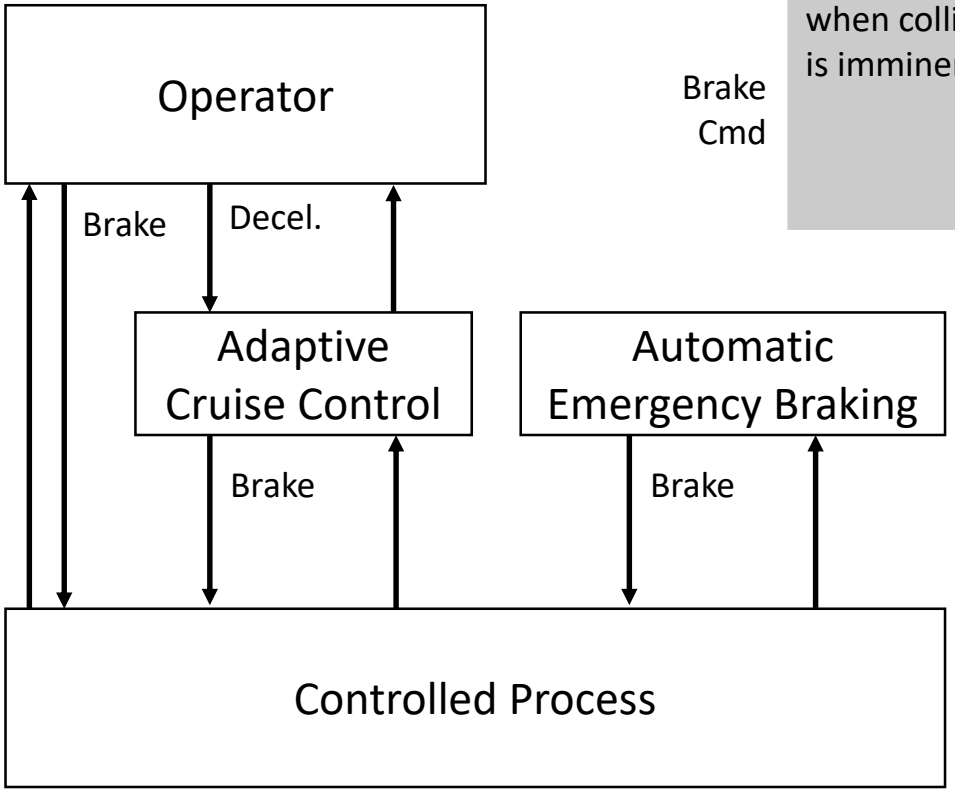
Author identified a valid UCA, but it was not adequately communicated to reviewers and others.

Confusion can be avoided by writing whole UCA.

UCA-1: Operator does not provide Close MSIV Cmd when SGTR and other systems operational [H-1,2]

Confusing control actions from multiple controllers

	Not providing causes hazard	Providing causes hazard	Too early, too late, out of order	Applied too long, Stopped too soon
Brake Cmd	when collision is imminent	…	…	…



Author identified a valid UCA, but it was not adequately communicated to reviewers and others.

Confusion can be avoided by writing whole UCA.

UCA-1: Automatic Emergency Braking does not provide Brake Cmd when Collision is Imminent [H-1,2]

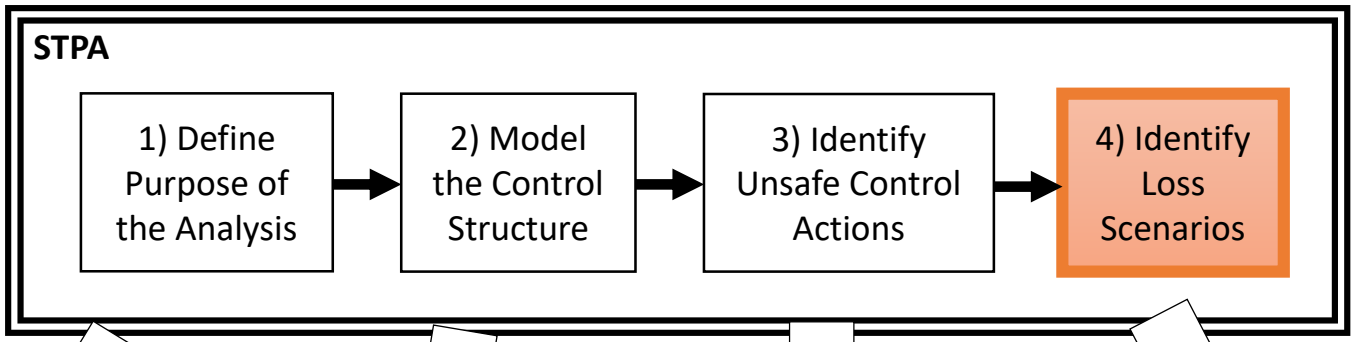
Current guidance

Tips to prevent common mistakes when identifying UCAs

- Ensure every UCA specifies the context that makes the control action unsafe.
- Ensure UCA contexts specify the actual states or conditions that would make the control action unsafe, not potential beliefs about the actual states.
- Ensure the UCA contexts are defined clearly.
- Ensure the UCA contexts are included and not replaced by future effects or outcomes.
- Ensure traceability is documented to link every UCA with one or more hazards.
- Review any control action types assumed to be N/A, and verify they are not applicable.
- For any continuous control actions with a parameter, ensure that excessive, insufficient, and wrong direction of the parameters are considered.
- Ensure any assumptions or special reasoning behind the UCAs are documented

STPA Handbook

System-Theoretic Process Analysis (STPA)

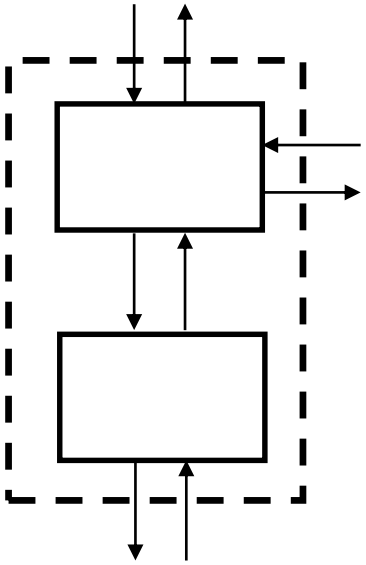


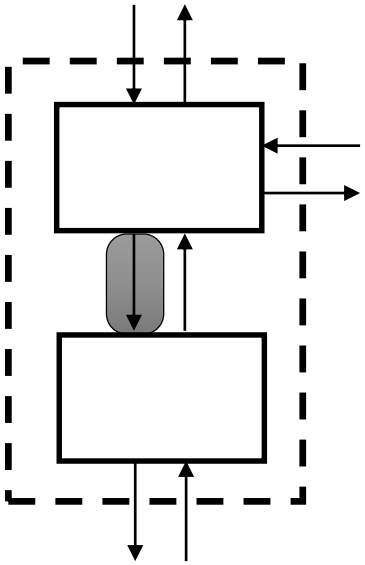
Identify Losses, Hazards

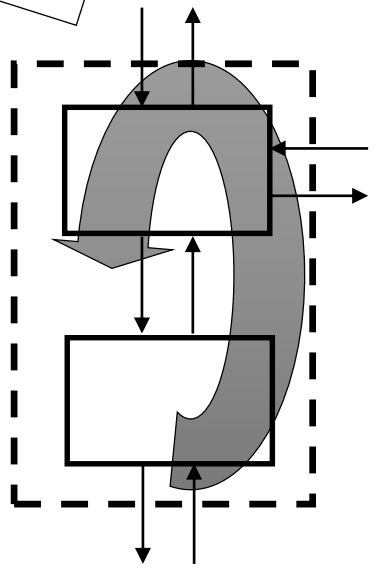
Define System boundary

Environment

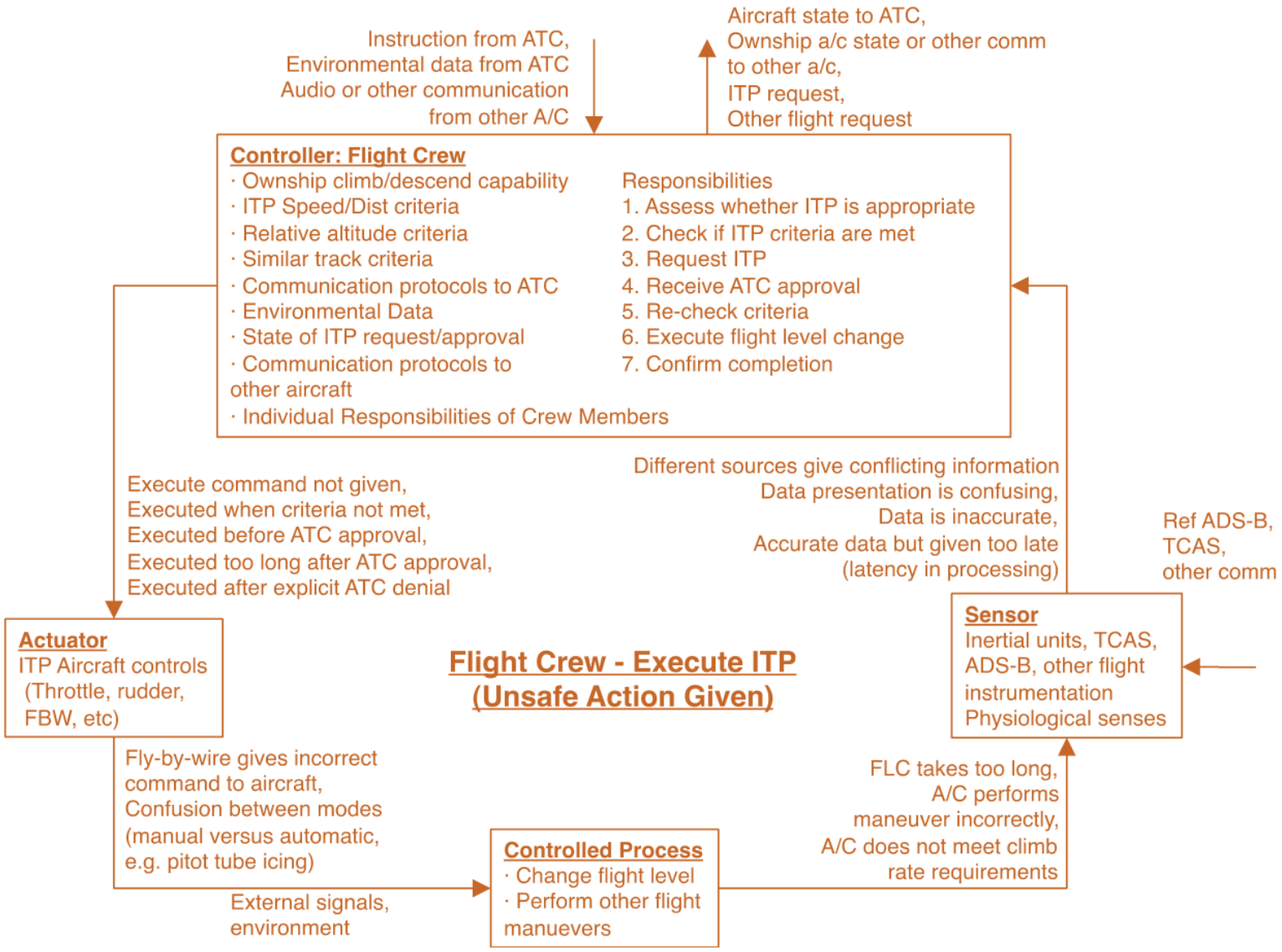
System







Identifying causal factors without interactions



Causal factors should be more than failures and malfunctions

Step 1 no.	Hazards	Severity	Causal factors
1a	System dysfunction due to failure of object detection system	Catastrophic	Object detection component failure (camera, radar, or motion sensors)
			Communication error (no signal)
1b	Malfunctioning of the system due to incorrect input from object detection system	Catastrophic	Corrupted communication (wrong signal)
			Malfunctioning of camera, radar, and motion sensors
			Communication system does not work on time
2a	Incorrect and missing calculation of vehicle status and collision probability due to failure or malfunctioning of vehicle complex sensors	Catastrophic	Failure of vehicle sensors

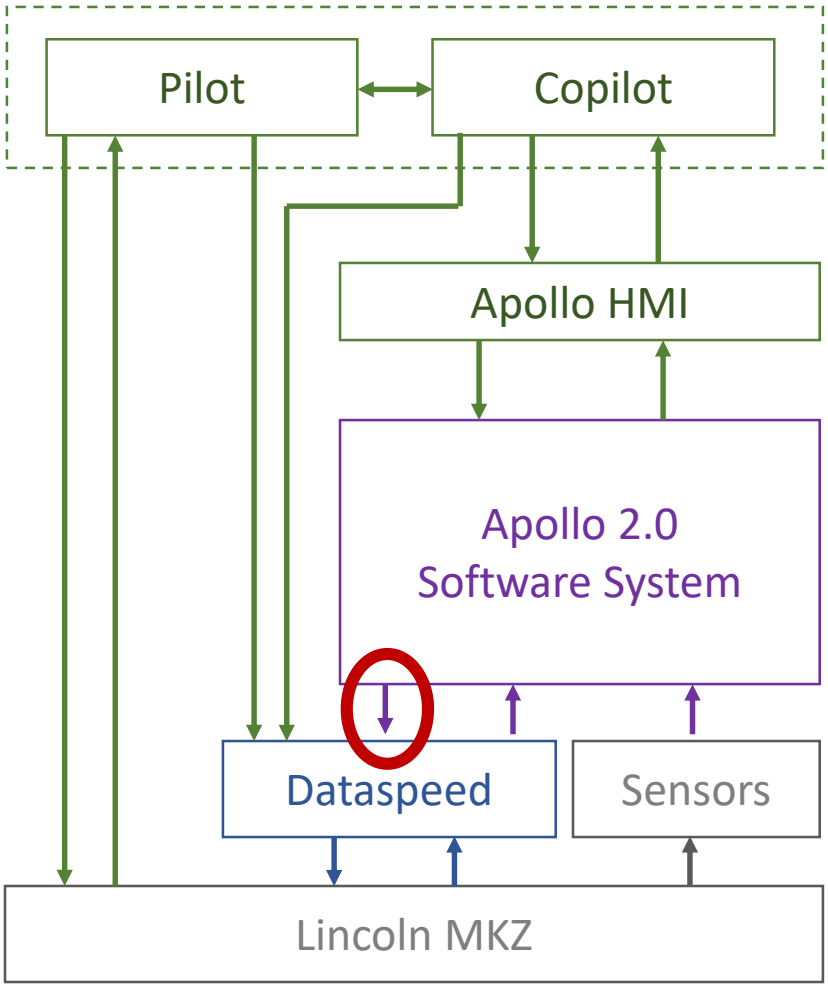
Current guidance

Tips to prevent common mistakes when identifying Scenarios

The most common mistake is to identify individual causal factors rather than a scenario. For example, you may be tempted to create list of factors like “wheel speed sensor failure”, “wheel speed feedback is delayed”, “loss of power”, etc. The problem with listing individual factors outside the context of a scenario is that it’s easy to overlook how several factors interact with each other, you can overlook non-trivial and non-obvious factors that indirectly lead to UCAs and hazards, and you may not consider how combinations of factors can lead to a hazard. Considering single factors essentially reduces to a FMEA where only single component failures are considered.

STPA Handbook

Better Scenario Example



UCA-1: Apollo provides throttle cmd when forward collision is imminent

- Can occur if Apollo incorrectly believes forward collision is not imminent (Process Model Flaw)
- Feedback: Apollo is not designed to detect automatic emergency braking or disable throttle commands.

Resulting potential requirements

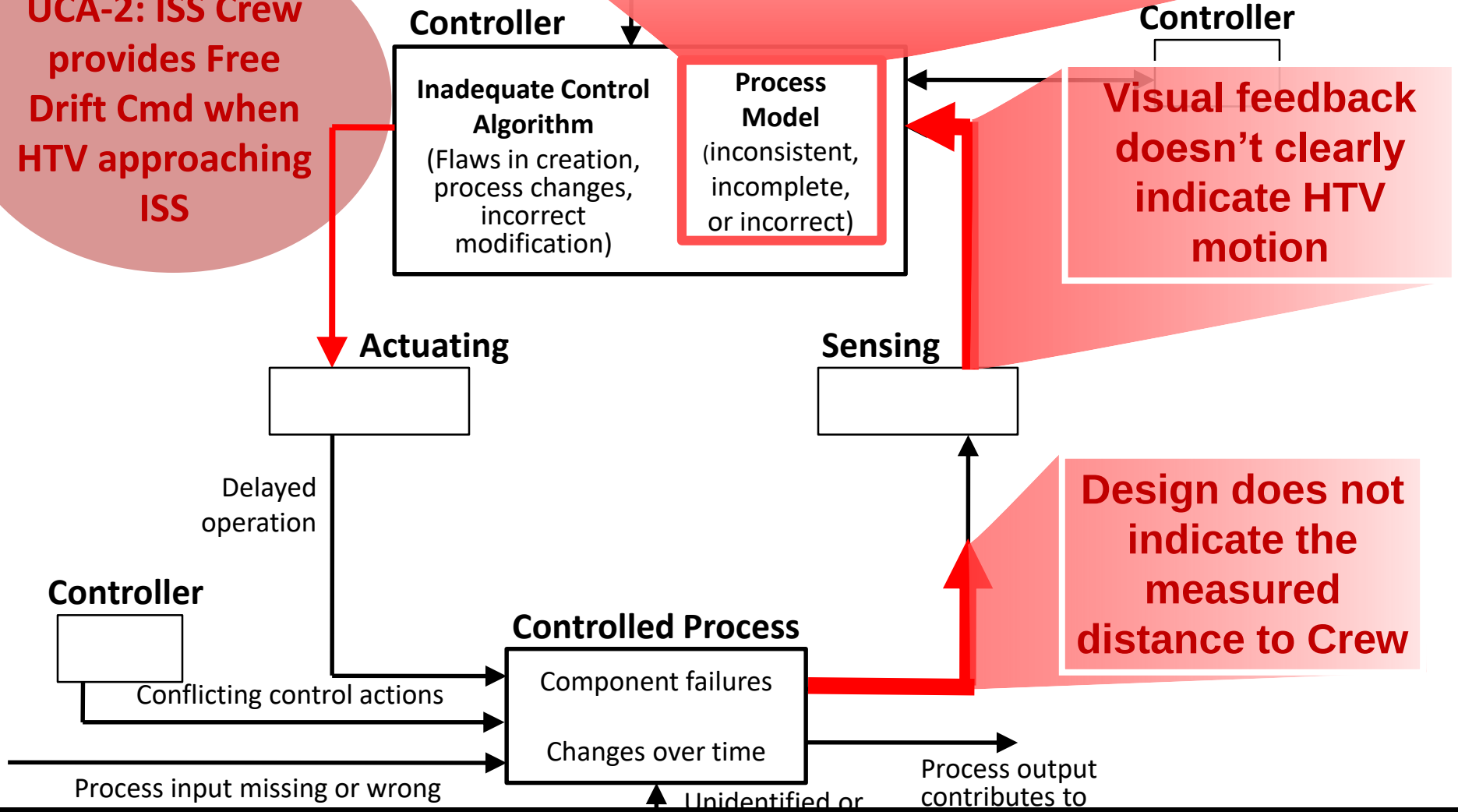
- R-1: Apollo must not provide throttle cmd when AEB engages
- ..

Actual design: The vehicle is designed to override automatic emergency braking if throttle commands are received

Better Scenario Example

Flawed Process Model:
ISS Crew incorrectly believes HTV is not approaching ISS

UCA-2: ISS Crew provides Free Drift Cmd when HTV approaching ISS



Better Scenario Example

Driver accelerates when vehicle is not in appropriate range (e.g. reverse instead of drive)

Driver incorrectly believes vehicle is in Drive

MM not updated because vehicle ignored cmd to shift to Drive (stayed in reverse)