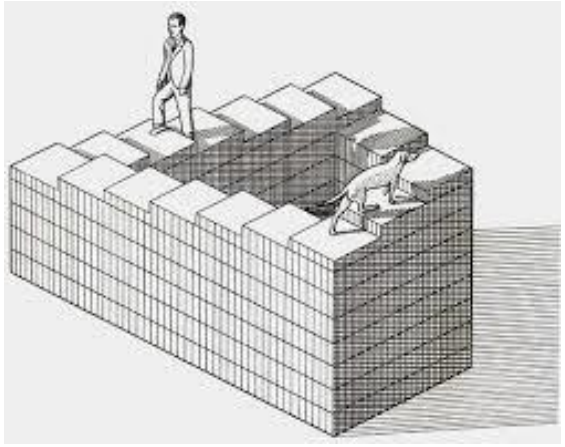


Using STPA to Create a Conceptual Architecture

Prof. Nancy G. Leveson

Aeronautics and Astronautics
MIT

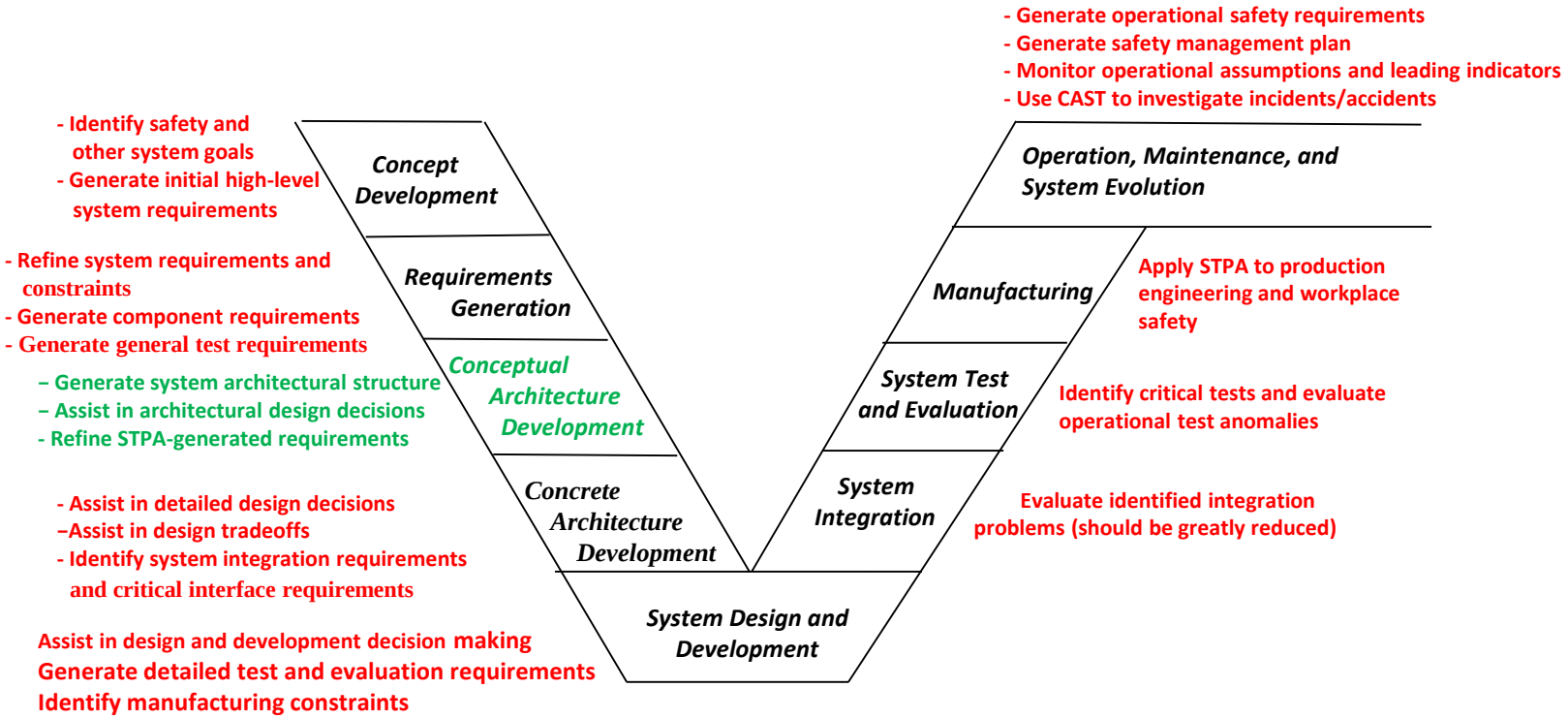
The Problem

- Architectures often created before
  - Know all requirements and constraints
  - Independent of specific system requirements or constraints
- Often just use standard architectures
  - Not necessarily reflective of system type being developed
  - Reflect some goals/constraints but not others
- Often dive into details prematurely
  - Decompose into standard functional components
  - Specify logical and physical details of connections between components (network structures, interface specs)
  - Design physical interactions before know what connections are important or needed
  - Usually little or no tracing to detailed requirements and desired system-level properties

The Problem (2)

- Results in systems where unique requirements only vaguely tied to architecture
- Safety engineering efforts reduced to producing a lot of paper with no real impact on actual system design
- Security efforts delayed until little impact on system design and system cannot be protected against adversaries
- Makes it harder and costlier to ensure safety/security/etc. are satisfied by implementation if haven't designed these qualities into the system from beginning and may even be infeasible
- Maintenance and upgrades may be enormously expensive
- MBSE does not capture unique architectural requirements for CPS (Cyber-Physical Systems) and control systems

New step in V-model: Conceptual Architecture Development



- Bridges gap between "shall statements" and detailed architecture development
- Provides concrete tracing between requirements and design

Spacecraft Physical/Logical Design vs. Control Structure

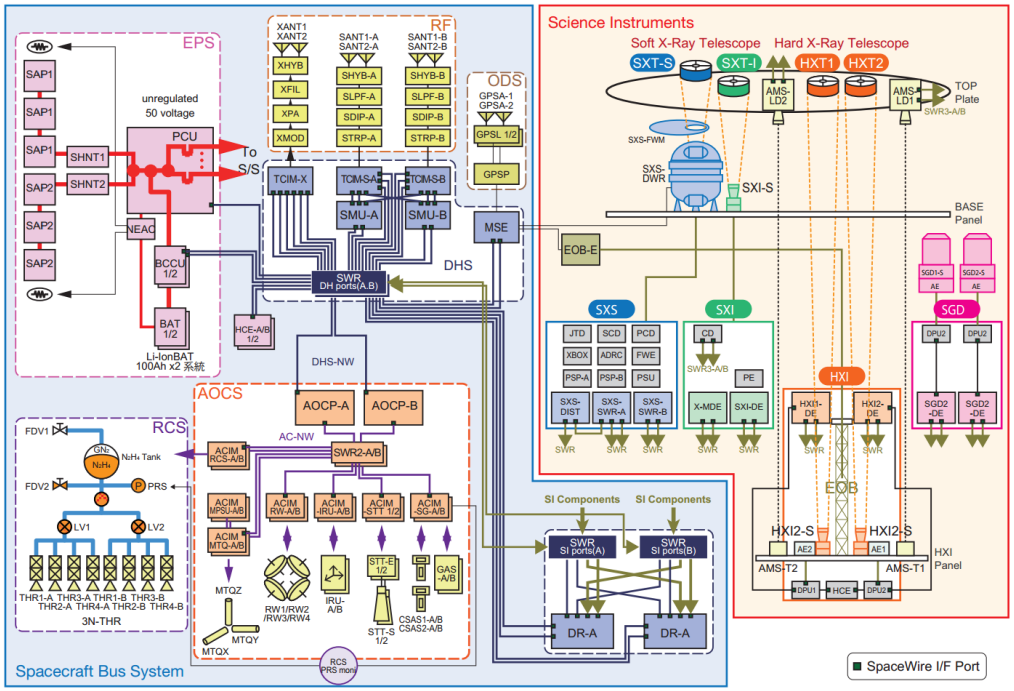
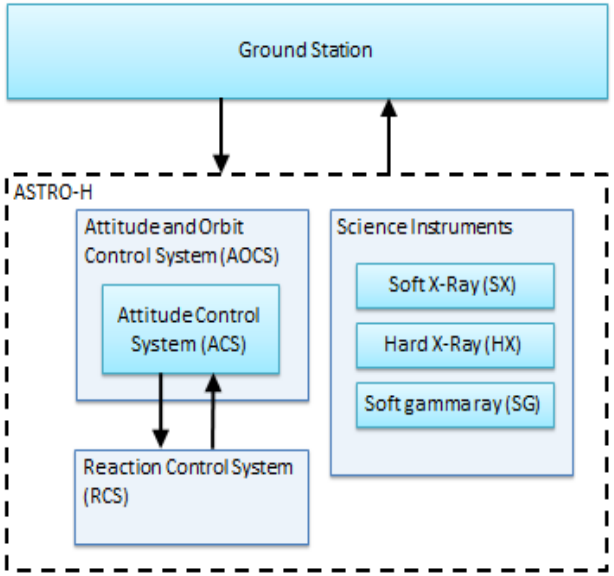


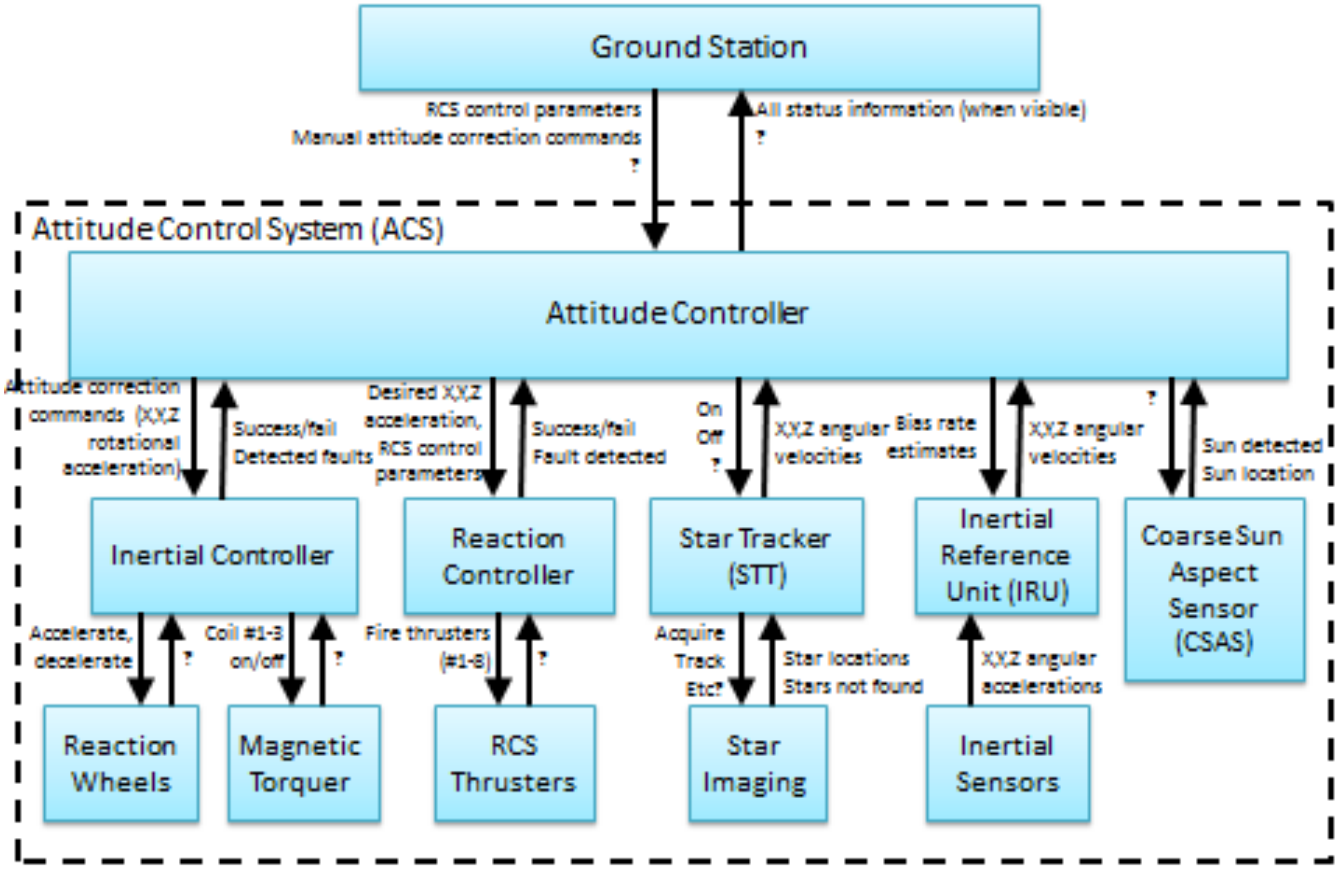
Figure 3.9: System block diagram. A is the primary and B is the redundant system.

High-level control structure



- MBSE should not mean only one model used (and one that is 40 years old and not appropriate for CPS)
- Different models useful for different types of analysis (provide different abstractions)

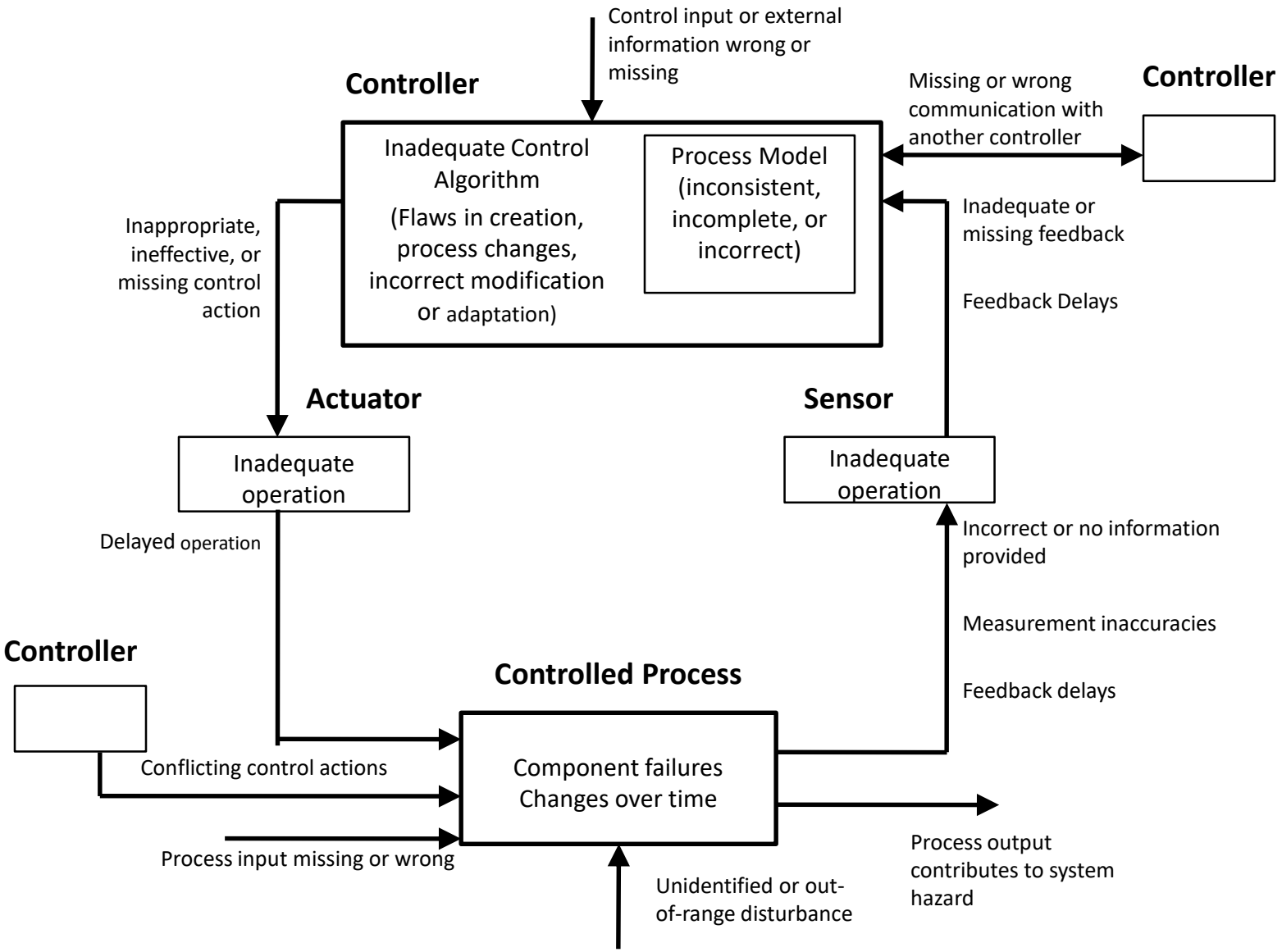
Create more detailed control structure alternatives (architectures) as design proceeds



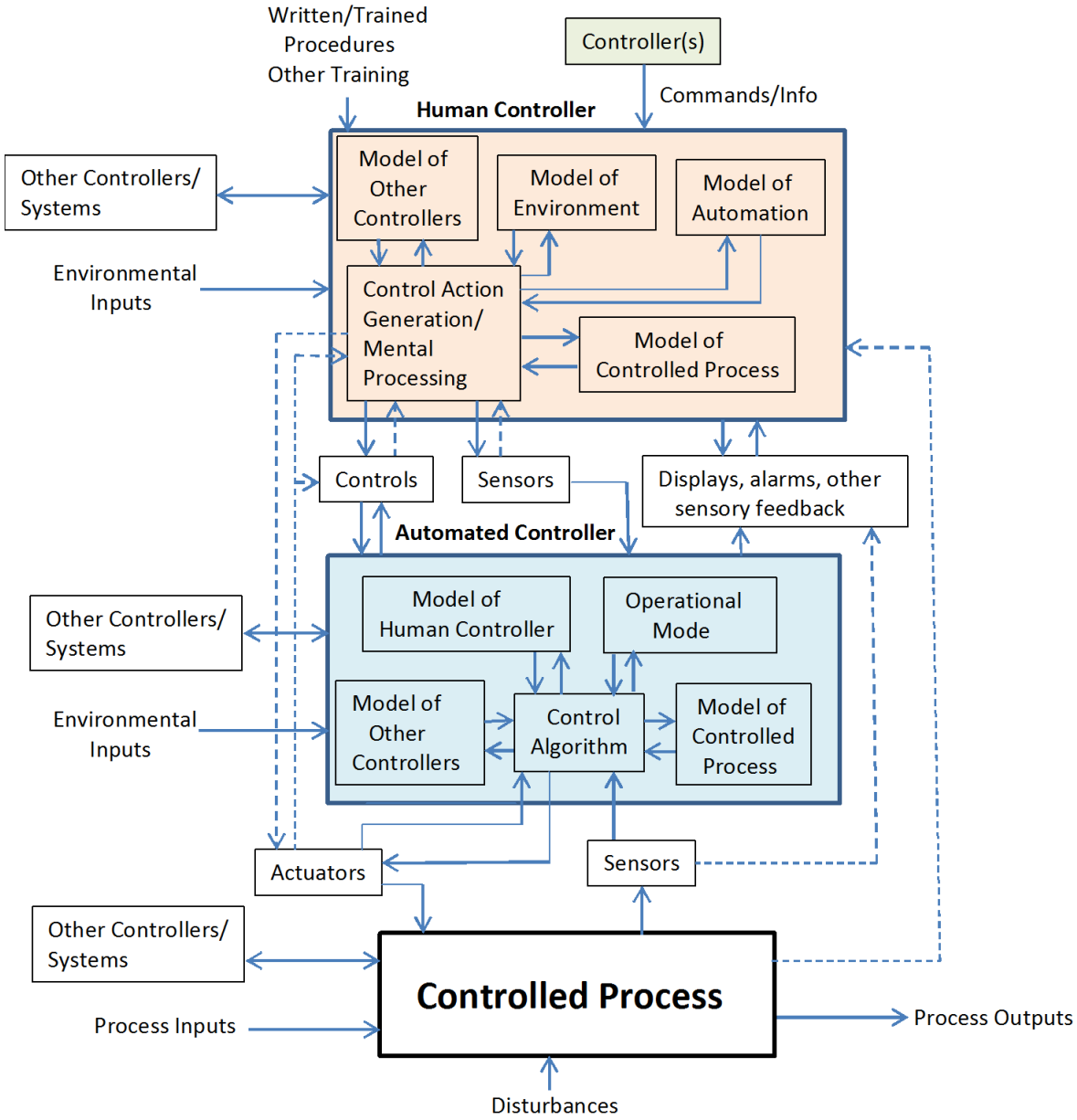
Design Process using Conceptual Architecture

1. Establish system goals
2. Create CONOPS
3. Identify high-level requirements and system hazards
4. Define basic control structure
5. Derive high-level safety constraints using STPA and the control structure
6. Assess risk using STPA scenarios and the risk matrix (optional)
7. Create the initial conceptual architecture (from the basic control structure) and refine it using the STPA results.
8. Create physical/logical architecture from the conceptual architecture
9. Create a detailed systems design using additional STPA analysis in decision making
10. etc.

Identifying Causal Scenarios



General Format of Conceptual Architecture (Control Structure)

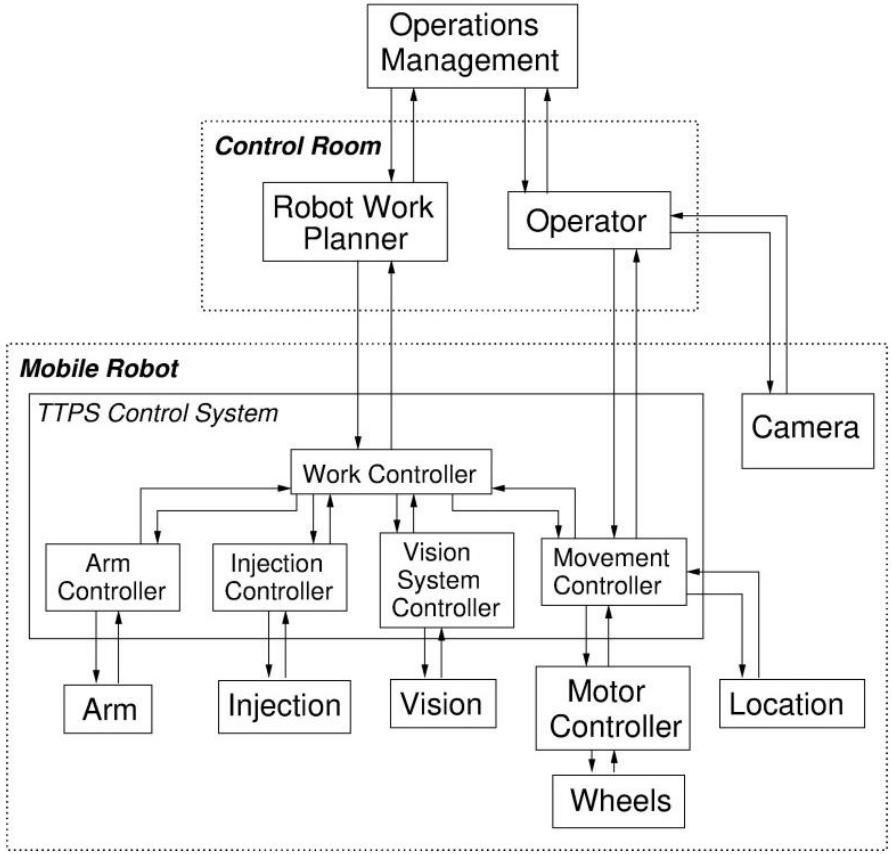


Start at higher level of abstraction and refine using results of STPA and other analyses

Note potential for human-centered design

Example 1: Industrial Robot

- Design at CMU to service the thermal tiles on the Space Shuttle. Never used.
- I redid the design using STPA. Details in the paper.
- Initial conceptual architecture

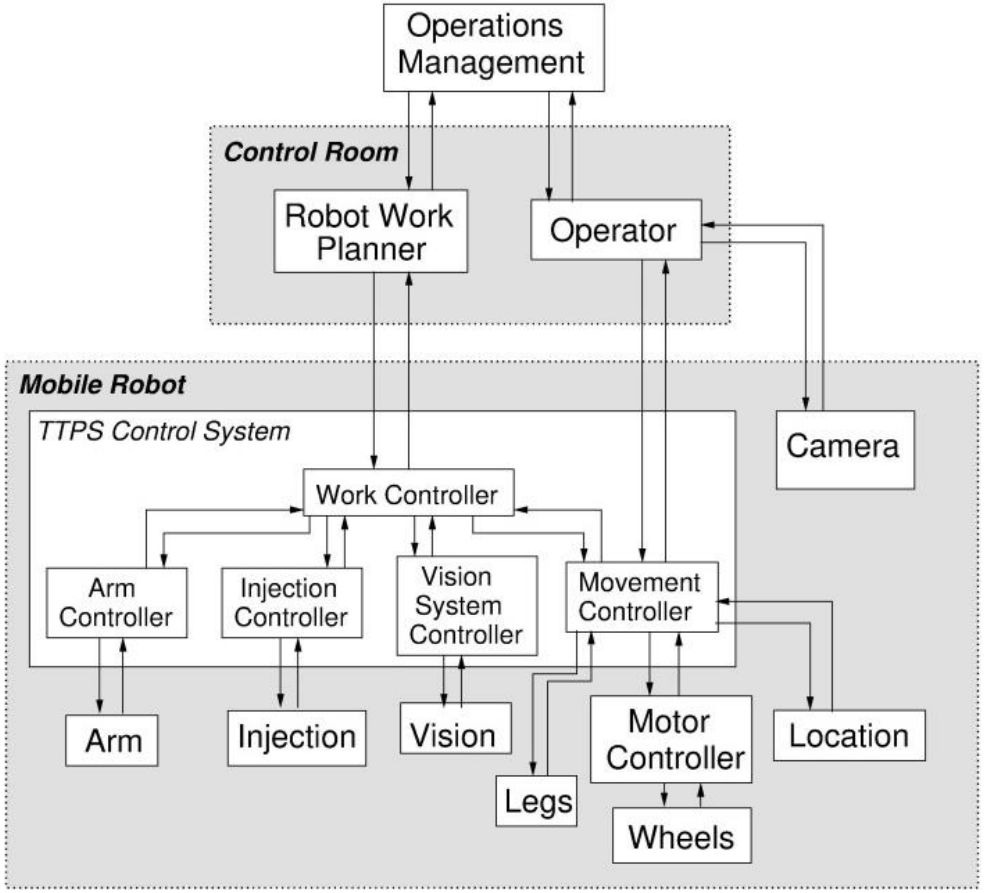


System-Level Hazard

- **H1**: Violation of minimum separation between the robot mobile base and objects (including the orbiter and humans) [**A1-2**, **A2-1, A3**].
- **H2**: **Unstable robot base** [**A1-2**, **A2-1, A3**].
- **H3**: Movement of the robot causing injury to humans or damage to the orbiter [**A1-2**, **A2-1, A3**].
- **H4**: **Conditions that could lead to damage to the robot** [**A3-4**].
- **H5**: Conditions that could lead to fire or explosion [**A1-2, A-2, A-3**].
- **H6**: Contact of human with DMES waterproofing chemical [**A2-2, A3-1**].
- **H7**: Inadequate orbiter thermal tile protection [**A1-1**].

Industrial Robot (2)

- Identified that unstable when manipulator arm extended, so added stabilizer legs.



New hazards and constraints after legs added:

H2: Unstable robot base [**A1-2**, **A2-1, A3**] is refined into

H2.1: The manipulator arm is extended while the stabilizer legs are not fully extended.

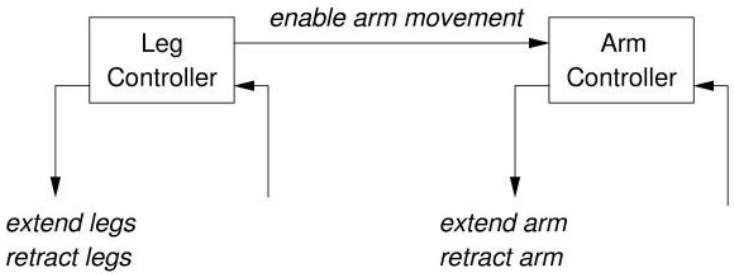
H4: Conditions that could lead to damage to the robot [**A3-4**]

H4.1: Legs extended during movement of robot

Two new refined constraints on the design:

1. The manipulator arm must never be extended if the stabilizer legs are not fully extended.
2. The mobile base must not move with the stability legs extended.

Doing the STPA Analysis of these refined hazards:



Control action	Not Provided	Provided	Early/late/wrong order	Stopped too soon
Extend legs	Legs not extended before arm extended H1	Extend legs during movement H2	Extend arm before legs extended H1	Stop before legs fully extended H1
Retract legs	Legs not retracted before movement H2	Retract legs while arm extended H1	Retract legs before arm fully stowed H1	Stop while legs still partially extended H1

Control action	Not Provided	Provided	Early/late/wrong order	Stopped too soon
Extend arm	(tile processing hazard)	Extend arm when legs retracted H1	Extend arm before legs fully extended H1	(tile processing hazard)
Retract arm	Not retracted before movement starts H2	(tile processing hazard)	(tile processing hazard)	Stop before arm fully stowed and movement starts or legs retracted H1 H2

Combining similar entries for H1 and H2:

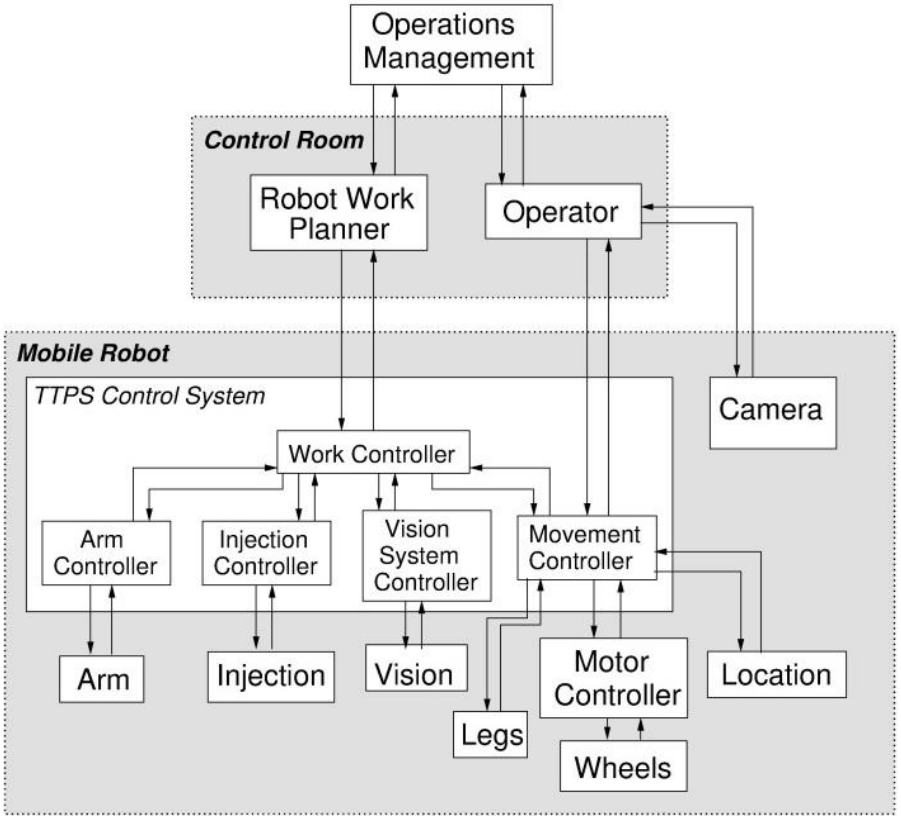
- Unsafe control actions by leg controller:
  1. The leg controller does not command a deployment of the stabilizer legs before the arm is extended.
  2. The leg controller commands a retraction of the stabilizer legs before the manipulator arm is fully stowed.
  3. The leg controller commands a retraction of the stabilizer legs after the arm has been extended or commands a retraction of the stabilizer legs before the manipulator arm is stowed.
  4. The leg controller stops extension of the stabilizer legs before they are fully extended.
- And one for the arm controller:
  1. The arm controller extends the manipulator arm when the stabilizer legs are not extended or before they are fully extended.

And restated as safety constraints:

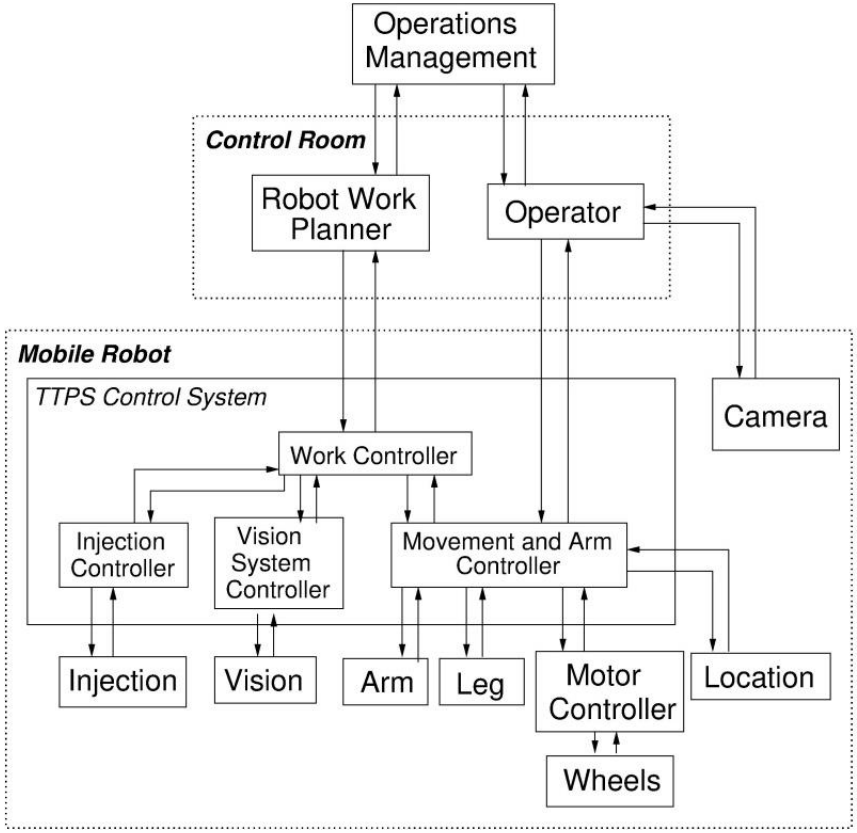
1. The leg controller must ensure the stabilizer legs are fully extended before arm movements are enabled.
2. The leg controller must not command a retraction of the stabilizer legs when the manipulator arm is not in a fully stowed position.
3. The leg controller must command a deployment of the stabilizer legs before arm movements are enabled; the leg controller must not command a retraction of the stabilizer legs before the manipulator arm is stowed.
4. The leg controller must not stop the leg extension until the legs are fully extended.

Industrial Robot: Old and New Architecture

Old



New



Example 2: UAV (Major David Horney)

- Generating a conceptual architecture for tethered UAVs
- First high-level architecture makes no assumptions about whether piloted aircraft controls the software-controlled aircraft.
  - Control decisions may change in different phases of flight and formations may change throughout a mission
  - Two candidate architectures evaluated.
    1. Human PIC determines formation shape and tethered aircraft responsible to implement command by maintaining their position in specified formation.
    2. Tethered aircraft determine optimal formation shape based on current conditions and phase of flight.
  - Shows how tradeoffs in architectural design might be analyzed and resolved.

Example 3: Manned-Unmanned Aircraft Teaming (Jeremiah Johnson)

- Several generic architectures have been produced for this problem
  - Define decomposed set of functional components and their interactions (all are different)
  - None designed considering specific safety and security requirements
  - Each designed for specific and different payloads, mission sets, and communication capabilities. Apply to specific missions and not all potential missions.
- Instead, Robertson used STPA to create a generic MUM-T conceptual architecture to satisfy requirements for general design of a swarm architecture that is safe when coordinating with a manned aircraft.
  - Includes safety and security requirements for an entire system, including software, manned aircraft, ATC, Ground Station, etc.

Open Research Questions

- How to generate a physical/logical architecture from the conceptual architecture
- Assuring physical/logical architecture satisfies safety/security constraints in conceptual architecture (hopefully much easier)
- Potential role of conceptual architectures in certification

Current Conceptual Architecture Validation Projects

- Autonomous automobile certification (Michael Schmid, M.S. thesis completed)
- Army UAV design (AF Lt. Elias Johnson, in process M.S. thesis)
- Army FARA design and risk assessment (Navy Lt. Dro Gregorian and Army Cpt. Sam Yoo and Lincoln Lab)
- NASA Urban Air Mobility Research
- Army human-centered design research for helicopter degraded visual environments

Questions?

The paper on this topic can be downloaded from:

http://psas.scripts.mit.edu/home/nancys-white-papers/

Two examples follow:

- Creating a conceptual architecture for a defense system UAV (Lt. Elias Johnson)
- Use in certification of autonomous autos (Michael Schmid)