Chapter 8

Some More Conceptual Machinery

We have already addressed the need for rigor in system analysis, and seen that a rigorous approach to describing systems demonstrates subtleties in the definition of hazard, and difficulties in the proposed analysis, whereby one attempts to calculate overall risk as a function of hazard, severity, and likelihood that a hazard will lead to an accident. We now introduce some further notions which will help in the analysis of systems.

8.1 System Properties in the Large

Causality in Aviation It is required by international treaty (the 1948 Chicago Convention, setting up the International Civil Aviation Organisation) that accidents to commercial aircraft be investigated, and a *probable cause* and *contributing (causal) factors* for the accident determined. Commercial aviation represents a significantly complex system, involving complex systems such as air traffic control (considered by Perrow [Per84]) and invididual commercial airliners as parts. We may conclude that the causal influences of complex system parts on each other and on the environment, and vice versa, is an important and significant feature of such systems.

Commercial Aircraft As Complex Systems Commercial aircraft themselves are highly complex systems, with functioning parts that are mechanical (engines, control surfaces), parts that are electrical (lighting, control systems and control system signalling), digital electonics (avionics) and human (pilots).

What's a Part? We include, say, pilots as parts of the aircraft, because the aircraft's behavior is presaged on the (formal) behavior of the pilots; and the behavior of the pilots is specified as part of the aircraft's operation. In general, one can consider an object O with behavior to be a part of a system insofar as the physical behavior of the system is coupled to the behavior of the object O. So we would consider pilots to be part of the aircraft system, because they physically manipulate objects in the cockpit which have a direct effect on the behavior of the aircraft. We do not consider air traffic control to be part of the aircraft because Air Traffic Control (ATC) behavior is mediated through pilot understanding and compliance, and they have no direct influence, as do pilots, on the behavior of the aircraft. ATC is an aviation-system part which communicates with aircraft-system parts. We are free to draw the boundary of a system where we like (that is, to include or exclude certain objects) and criteria which we may use include

- tradition
- the intensity of interoperation
- the mode of interoperation

Interactive Complexity and Tight Coupling: Perrow We have identified *complexity* as a feature of systems. It is an intuitive notion that most people understand (although they may judge differently depending on a number of factors, including the intellectual tools they use for understanding). Other large-scale features of systems are important, including two singled out by Perrow:

- *Interactive complexity.* This feature represents the degree to which system parts need to communicate and interact with each other during normal system operation.

- *Coupling.* Coupling is further classified as *tight coupling* and its contrary *loose coupling*. This feature represents the degree to which a change in state or behavior of a system part causes changes in state or behavior in other system parts, and the degree of such changes. In a tightly coupled system, many other parts of a system will be sensitive to small changes in a single system part, and the associated state or behavioral changes will be significant.

Perrow claimed that tightly-coupled, interactively complex systems were unusually subject to what he called *system accidents*, which is an accident which one cannot attribute causally to failure of any one precise system part. All parts seemed to function as they were designed to do, but nevertheless an accident occurred. Perrow went so far as to claim that when systems are highly tightly-coupled and interactively complex, system accidents were virtually inevitable. He provided a number of studies in [Per84] to support his claim. A significant in-depth study of one highly complex system, the U.S. military's system of nuclear weaponry, was investigated by Sagan [Sag93], who came to conclusions supporting Perrow's contention.

System Decomposition We have already noted that one should not conflate reliability (the property of a system to continue to perform its intended function, or not) and safety (the avoidance of accidents). It follows that a system accident in Perrow's sense may not represent a system failure (a failure to perform its desired function), unless avoiding accidents was an explicit function of the system. It may not be.

According to the failure reasoning in Figure 3.15, a failure in the system as a whole may be put down to a failure in some system component, provided that the components form an adequate decomposition. Assuming that an accident represents a system failure (as it should if the required safety properties of the system are included in the system requirements specification), an adequate decomposition will determine in which part of the system a failure is located. Perrow suggests the DEPOSE composition; his contention that there exist system accidents suggests that DEPOSE is not an adequate decomposition. But he provides no reasoning to suggest that adequate decompositions do not exist. Intuitively, they do. Features of systems are found which contribute to accidents. These features are part of some decomposition of a system. No accident has ever occurred in which investigators have simply given up, and said that although they know everything there is to know about how the accident occurred, nevertheless they cannot say anything about any system part which contributed.

Heterogeneity I call a system *heterogeneous* if it includes parts of widely disparate types: for example, mechanical, electrodigital, human, procedural. An aircraft includes electromechanical, electrodigital, and human parts and its correct operation requires procedural parts also. An air traffic control system has similar parts in different proportions; minimal ATC systems involve radios and recorders as the sole electromechanical parts, have no electrodigital parts, and are humanly and procedurally intensive. The importance of heterogeneity lies in the different operational and failure modes of the different types of parts. For example

- electrodigital systems are functionally reliable, do not adapt to situations they are not explicitly designed to handle, and fail in unpredictable ways;
- humans are functionally unreliable (relatively speaking), adapt to situations they were not explicitly trained to handle, and fail in predictable ways
- procedures do not have behavior, they specify behavior, hence the notion of functional reliability does not apply; they do not adapt to situations they are not explicitly designed to handle, and they fail in unpredictable ways.

Openness An *open system* is a system whose constitution or behavior is comparatively affected by the environment. In constrast, a *closed system* is one whose constitution or behavior is relatively unaffected by the environment. For example

- Computer communication subsystems may be closed or open:

  - Communication in a computer network connected by appropriately shielded cables is relatively unaffected by the location of other objects in the space, by temperature, by light, by radio signals and by electromagnetic fields.
  - Communication in a network connected by infrared sensors is affected by the location of other objects (it is "line-of-sight"), and by the presence of other infrared radiation such as that generated by spotlights.
  - Communication in a network connected by radio is relatively unaffected by the location of other objects except for building structures which are relatively radio-opaque, and is highly affected by the presence of other radio signals, of which there are many.

  A communication subsystem connected by cabling is therefore relativel closed; one connected by radio or infrared is relatively open

- A pressure tank subsystem of a chemical plant may be affected by the ambient temperature, but this may also be suitable controlled by a cooler which belongs to the system. Else, it is affected mainly by the inflows and outflows, which are part of the overall system, and which may be regulated within certain specified limits. The pressure tank may be adversely influenced by bombs, nuclear explosions, earthquakes (to some extent) and large-scale plant fires, and little else. It is a relatively closed system.

- An aircraft in flight is significantly affected by the motions of the air mass it is flying through, and by the presence or absence of terrain and other non-gaseous physical objects in its flight path. It is a relatively open system.

A Proposal Connecting Hazard Definition And System Properties It is worth considering if the appropriate definition of hazard for a safety analysis of a system can partly be determined from system properties. For example, Hazard-1 was developed in the commercial nuclear and chemical industries, both of which deal with relatively closed systems. In a relatively closed system, it makes sense to focus on the system state as the major component of a risky situation, since the system state is the major factor affecting subsequent behavior. In contrast, many aviation and other safety analysts seem to prefer Hazard-2, in which properties of the environment are singled out as the significant contributors to a hazardous situation. In a case in which system behavior is largely affected by the environment state, this focus seems to make sense.

8.2 Causality

Formal System Descriptions Suffice to Define Causality We have construed systems as consisting of objects with behavior, and have described in Section 3 how we may consider these formally:

- system state is described through state predicates
- state predicates can be written in, say, a first-order logical language
- behavior is construed via a discrete unending sequence of states
- temporal operators such as $\Box$ and $\Diamond$ are used to make assertions about future states in behaviors
- states can be considered to be *near* and *far* from each other
- altenatives behaviors can be considered to be *near* and *far* from each other

Construing systems in this way allows us precisely to define causality. We shall now see how.

8.2.1 Hume

Hume's Second Definition David Hume gave two definitions of causality over 200 years ago. Here is his second.

> ....we may define a cause to be *an object, followed by another, and where all the objects similar to the first are followed by objects similar to the second.* Or, in other words *where, if the first object had not been, the second never had existed.*

[Hum75, Section VII, Part II, paragraph 60].

We may consider the word '*object*' to refer also to events, maybe states, as noted in the work of John Stuart Mill [Mil73a].

David Lewis notes [Lew73a] that there are two definitions given by Hume, and over the course of the subsequent couple of hundred years, the consequences of these notions has been explored. The first definition, in terms of observable regularities, leads to a psychological explanation of causality and is of less interest for our purposes. The second definition, above, is *counterfactual* – it talks of what might have been but was not.

8.2.2 The U.S. Air Force

This is what the U.S. Air Force says about accident explanations [Uni94]:

> **3-11. Findings, Causes, and Recommendations.** The most important part of mishap investigation is developing findings, causes and recommendations. The goal is to decide on the best preventive actions to preclude mishap recurrence. To accomplish this purpose, the investigator must list the significant events and circumstances of the mishap sequence (findings). Then they must select from among these the events and conditions that were causal (causes). Finally, they suggest courses of action to prevent recurrence (recommendations).
>
> **3-12. Findings:**
>
> a. Definition. The findings ..... are statements of significant events of conditions leading to the mishap. They are arranged in the order in which they occurred. Though each finding is an essential step in the mishap sequence, each is not necessaily a cause factor......
>
> **3-13. Causes:**
>
> a. Definition. Causes are those findings which, singly or in combination with other causes, resulted in the damage or injury that occurred. A cause is a deficiency the correction, elimination, or avoidance of which would likely have prevented or mitigated the mishap damage or significant injuries. A cause is an act, an omission, a condition, or a circumstance, and it either starts or sustains the mishap sequence.....

In the paragraph defining causes, the counterfactual definition is used.

8.2.3 Lewis

Lewis's Formal Definition of Causal Factor Suppose A and B are state predicates or state changes (which we shall call *events* from now on). David Lewis's definition of causal factor proceeds as follows. *A is a (necessary) causal factor of B* just in case, had A not occurred, B would not have occurred either. This definition is counterfactual. Before we explain the formal meaning of counterfactual expressions, also due to Lewis [Lew73b], we illustrate the definition of causal factor. Consider a system in which there is a programmable digital component which contains a bit, stored in a variable named X. With systematic ambiguity, we shall refer to this bit as X. Suppose the electronics is wired such that, when X is set, a mechanism (say, an interlock) is thereby set in motion. Suppose the interlock has been well enough designed so that it can only be set in motion by setting X. Then X is a causal factor in any setting in motion of the interlock according to the Lewis definition: *had X not been set, the interlock* *would not have moved.* Furthermore, let us suppose that the digital component is well-designed, so that X can only be set by a specific operation O of a processor to set it, and that this operation is performed by executing a specific program instruction I. Then,

- *had the operation O not been performed, X would not have been set*, and
- *had the instruction I not been executed, the operation O would not have been performed.*

It follows that

- Performance of O is a necessary causal factor in setting X, and
- Executing I is a necessary causal factor in performing O

The Meaning of A Counterfactual Lewis also gives a formal meaning to a counterfactual. The counterfactual *had A not occurred, B would not have occurred* is interpreted as follows [Lew73b]. We have construed the real world as a behavior, and we have a relation of nearness amongst behaviors. Now, in the real world, B occurred, as did A. But we can consider the *nearest behaviors* to the real world in which A did not occur. The counterfactual *had A not occurred, B would not have occurred* is defined to be true (in the real world) just in case, in all these nearest behaviors in which A did not occur, B did not occur either.

The Semantics Applied to the Example We can consider behaviors near enough to the real world such that I was not executed. We're focusing on system predicates and environment predicates of this system, so we may presume that the more of them that are the same, the nearer the states of the alternative behavior are to the real world. It follows that in the nearest behaviors the design and intended operation of the system can be assumed to be identical to its design and intended operation in the real world. For these behaviors, then, in which I was not executed, O was not performed. And in these behaviors in which O was not performed, X was not set. And in these behaviors in which X was not set, the interlock was not set in motion. So consideration of the nearest behaviors shows that the counterfactuals are to be evaluated as true. Consequently, the assertions of causality (or, rather, *causalfactorality*) are true.

A Comment on the Relation of Nearness The relation of nearness between behaviors is ternary, and comparative: behavior B is nearer than behavior C to behavior A. For reasons that we shall not go into here, Lewis's formal semantics for counterfactuals requires that the nearness relation have a certain form. Fix A, then the relation *B is nearer than C to A* is a binary relation between B and C. Lewis's requirement is that this binary relation must be *ordinal*: it must define an order relation. That is:

Comparability every two worlds B and C are comparable: either B is nearer to A than C, or vice versa, or they are both equally near.

Asymmetry if B is nearer to A than C, then it cannot be the case that C is also nearer to A than B

Irreflexivity B is not nearer than itself to A

Transitivity if B is nearer than C to A, and C is nearer than D to A, then B is nearer than D to A

There are two further conditions on the order relation, that it be closed under arbitrary upper bounds and lower bounds, that need not concern us.

The Notion of Causal Factor is Not Transitive Lewis points out that his notion of causal factor is not transitive, that is

- If A is a causal factor of B, and B is a causal factor of C, this does not necessarily mean that A is a causal factor of C.

Since the intuitive idea of a cause is something that propagates through a "chain" of causal factors, Lewis proposes to define "cause" as the "transitive closure" of the relation of causal factor. The *transitive closure* of a relation R is the smallest (or "tightest", most narrowly defined) relation R^* which

- is transitive, that is, if aR^*b and bR^*c, then aR^*c, and
- contains R, that is, if aRb then aR^*b.

Another way of defining the transitive closure is by a recursive definition; it is demonstrable that the two definitions are equivalent for any relation R. The recursive definition is as follows. aR^*b if and only if

1. aRb, or
2. there is a c such that aRc and cR^*b, and
3. aR^*b only if this can be shown by (repeated) application of Rules 1 and 2 above.

We won't concern ourselves further with the notion of transitive closure. It suffices to know that

- there is a purely formal way of obtaining a unique transitive relation from a given binary relation, called the transitive closure, and
- the intuitive notion of "cause" appears to be transitive, so
- we may rigorously define "cause" as the transitive closure of "causal factor".

We shall need the notion of cause when it comes to discussing the "probable cause" of an aviation accident, in Part III.

8.2.4 Aside: Causality and Computers

Relation Between Instruction and Execution is Causal This example also illustrates that, according to the formal definition, the design of a digital system ensures that the relation between the form of an instruction and and its execution is causal. The instruction I says to increment regiester R. I is executed; R is incremented. Had the instruction not been to increment register R, then R would not have been incremented. Therefore, the form of I, that I is an instruction to increment R, is a causal factor in incrementing R when the instruction is executed.

Debugging is Causal Analysis This observation entails that debugging computer programs is a form of causal analysis. We shall use this observation later to motivate a method, *Why-Because Analysis*, of causal analysis of complex system failures. One can consider it akin to 'debugging' complex systems. Not only by analogy, but formally.