The Spectrum of Overwhelming Systems
Simple, Complex and Complicated

Complexity and Complicatedness: two different ways in which systems become **challenging** for us to understand and build.

Such systems risk **overwhelming** us.

The also correspond very closely to **two major scientific and design paradigms**:

- Complex systems science
- Systems-based approaches, like engineering.
Simplicity and overwhelmingness

• Simplicity and overwhelmingness has nothing to do with order *per se*

• Disordered simple systems; statistically describable (e.g. ideal gas)

• Ordered simple systems; keep repeating some pattern (e.g. materials)
Systems that are not simple

• The bird flock is complex – it has high *dynamical complexity*
• The computer is complicated – it has high *structural complexity*

They pack a very large number of degrees of freedom
The complex system: a bird flock simulation

• Perfect for agent-based simulation; fit perfectly into the complexity paradigm.
• A regular space where agents move
• A set of interaction rules for the agents:
  • Separation, alignment and cohesion: adjusting movement vectors dynamically
  • Initialize, update states according to rules.
• Enjoy the macroscopic patterns.
The complicated system: a computer

- NAND gate in CMOS technology: low level of physical computer organizations.
- Specific components arranged in a static and exactly designed fashion.
- Each component is very well known, it does very specific things.
- Together they do something very different but equally specific – something to which we attribute a function (here NAND)
- Meaningful only in a bigger context.
- CMOS components are parts in a toolbox from which we may compose even larger systems with emergent functionalities!
Moving up...

• We may design something like this for example, an EPROM chip.

• Or any of all the functionally distinct chips we find in electronic technologies.

• More than CMOS tech needed, but the toolbox is still quite limited and standardized.

• **Immense** Design Space that we may explore in the search for new designs that do new things...

• We may expand and improve the pieces in the toolbox.

• So what do we get? We get pieces in *yet another toolbox*...
Moving up again...

• Motherboard with lots of IC’s, wiring and other types of components.
• New emergent function again.
• All neatly delimited and well documented, all adapted to be fit together... not in a unique but in an open-ended fashion.
• We may also unpack any of these components in a similar way as what we just did.
...and again...

• A computer is assembled
• The complicated system is a nested hierarchy
• Components are configurations in the Design Space of a lower-level toolbox
• ...and they are part of higher level toolboxes.
...almost done...

• The computer is part of a workplace system
• A **sociotechnical** system!
• Toolboxes may combine social and technological components
....and finally (at least for now)

• Just to illustrate that we may still go on!
• The workplace itself may be a cog in an even bigger machinery... and then we may have a department in a corporation... and so on.
• We will get back to these chimera systems in coming sessions.
Simplicity again...

Notice how both types of systems become complex by being simple in some other way.

The complex system is not very complicated

The complicated systems is not very complex

We may understand them and design them precisely on that account!
Near-Decomposability

• Formal model: Herbert Simon’s “near-decomposability.”
• Tremendously influential as an archetype of how one designs and understands complex systems.
• Complex systems, in 1961, by the way, meant “complicated systems” in our parlance.
Levels and design

• A formalization of what we have been talking about?
The short run...

...is the time scale over which these ontological assumptions are valid.

There may or may not be a relevant short run.

**Traffic**

<table>
<thead>
<tr>
<th>Seconds</th>
<th>Hours</th>
<th>Years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only the internal environment dynamics moves – the objects of interest do not move in a relevant way</td>
<td>Good “short run” – outer environment remains fixed, and internal environment dynamics can be expressed in terms of behavior of the vehicle</td>
<td>The outer environment may change dramatically, and for reasons that are not understandable within the modelled dynamics</td>
</tr>
</tbody>
</table>
Central examples: Herds, traffic, social networks.

Main signifying features:

1. Many (even immensely so) components on same organizational level.
2. Many components but few component classes.
3. High redundancy: components may step in for other components of the same class (compare removing *an* ant with removing *the* liver).
4. Loose exogenous constraints on formation and dissolution of interactions between components. Exogenous structuring constraints apply to interactions between types of components; e.g. how do cars and trucks behave in traffic.
5. Strong endogenous structuring of component interactions (emergent patterns) may arise from the dynamics (shoals, traffic jams, paths, etc.)
Simplicity hook: If we deal successfully with emergence among very large numbers of interacting entities (which e.g. simulation helps us do) then, from the view of component classes, complex systems are much simpler than they may appear. Emergent patterns can be explained in those terms.

Desirable adaptive affordances:
1. Resilience (dampening of disturbances, redundancy)
2. Adaptation
3. Distributed action, monitoring and processing provides affordances unavailable to complicated systems.
4. Self-assembly/organization as path to building adapted systems.
Main challenges:
1. Chaos in massively parallel dynamics: (i) unpredictability; (ii) amplification of disturbances.
2. Emergence (e.g. macroscopic patterns) in strongly parallel and distributed dynamical systems.
3. Harnessing complex systems for adapted purposes invokes the same demand for “slaving” components as for complicatedness.

Main approaches: Computation and dynamical systems theory (e.g. chaos theory, synergetics). Simulation crucially allows mass dynamics to play out explicitly “in silico”.

Generation/maintenance: Generally, emergent complex patterns arise “suddenly” as interacting components come together, and dissolve if components seize to interact.
Complicated

**Central examples:** Technology, organisms.

**Main signifying features:**

1. Scale-separated level hierarchies.
2. Potentially very tall hierarchies, spanning from small to large scales.
3. Components have relatively few sub-components.
4. About as many component types as component instances.
5. Sub-components are co-adapted to specific complementary functions in a whole with emergent affordances and functions.
6. Low redundancy: components cannot generally take over the roles of other components.
7. Sub-components are “slaved”: they often make no sense separately.
8. Near-Decomposability essentially resets the number of degrees of freedom between sub-component and component.
9. Phased lifecycle:
   - **Assembly:** System assembled/developed with high precision in protected space, free from functional demands.
   - **Use:** Systems expresses intended set of functions, may undergo diagnostics and repairs to maintain function.
   - Transition between phases may be gradual, as in organisms.
**Simplicity hook:** The full system may pack very large numbers of components into delineable compartments organized in a level hierarchy. This strongly structures the patterns of permitted interactions and enables strong simplifying assumptions. We hardly need *any* knowledge about the embedding system to operate locally on its components.

**Desirable adaptive affordances:** Allows systematic exploration of design spaces: innovation and assembly may act in a strongly distributed and layered fashion; detailed designs (strong specialization), controllability, repeatability, scalability, precise and economic assembly, division-of-labor.
Main challenges:
1. Controlling and predicting the External Environment.
2. Alignment of goals and aims of components ("slaving").
3. Fine-tuned, non-redundant organization causes sensitivity to breakdowns and is an obstacle to dynamic use-phase adaptation.

Main approaches: Engineering, early “waves” of systems theory (cybernetics, operations research, control theory etc.), overall the “standard way” we think about design and governance.

Generation/maintenance: Complicated systems are assembled or, in biology, developed in morphogenesis.
Discussion points:

1. Is Near-Decomposability a good model of design and understanding in complex system?
2. Are there cases and conditions under which we depart from it in complicated systems?
3. What happens if we have a **complicated** complex model?
4. Does near-decomposability say something about human cognition?