A Preliminary Investigation of Design Simplification by Analogy

Marton E. Balazs
David C. Brown

Worcester Polytechnic Institute
Department of Computer Science

Simpler Design

Simpler structure:

- simpler attribute values (e.g. simpler shape)
- fewer components
- simpler relations (e.g. fewer components involved)

Simpler behavior:

- shorter causal chains
- fewer causal chains (e.g. fewer different causal chains corresponding to different inputs)

Simpler function:

- simpler inputs (e.g. simpler motion)
- fewer inputs (e.g. fewer forces applied)

Why Simplification by Analogy?

1. **reuse** of previous simplifications

2. possibility to:

   - **borrow** from other problems/domains
   - **produce** innovative simplifications
   - **discover** simplification rules

Potential Contributions

Simpler: More effective use of resources

- **structure:** less material...
- **behavior:** less wear...
- **function:** higher user productivity...

Simplification is a less researched area

Related Work

Simplification:

- **N. Suh**, Reduction of Information Content: information theoretical approach
- **Design For X**: rules for reducing the complexity of different phases of the life-cycle of a design

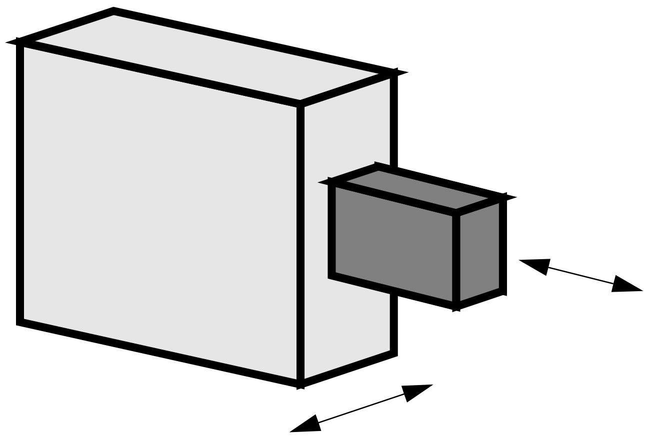
Functional Reasoning:

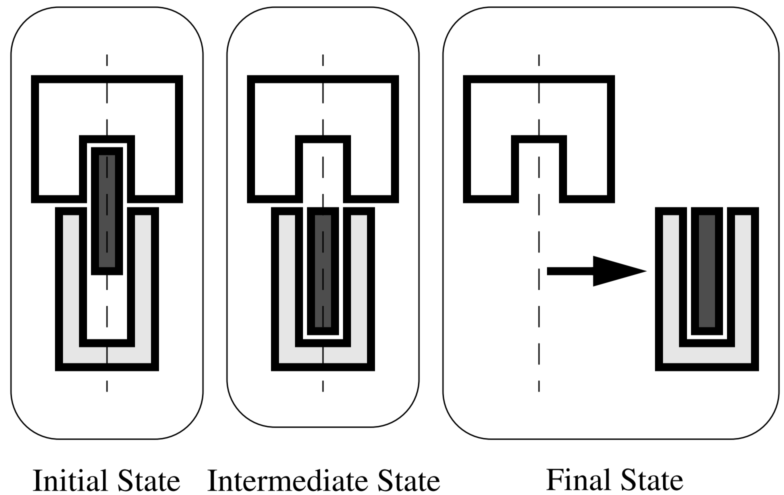
- Representation and reasoning about design function [**Chandrasekaran**] [**Goel**][**Chittaro**]
- Using functional reasoning [**Tomiyama**][**Chakrabarti**] [**Stroulia&Goel**]

Analogical Reasoning in Design:

- analogical design [**Goel**][**Qian&Gero**]
- purpose-directed analogical reasoning [**Kedar-Cabelli**]
- creative design [**Gero**][**Goel**]

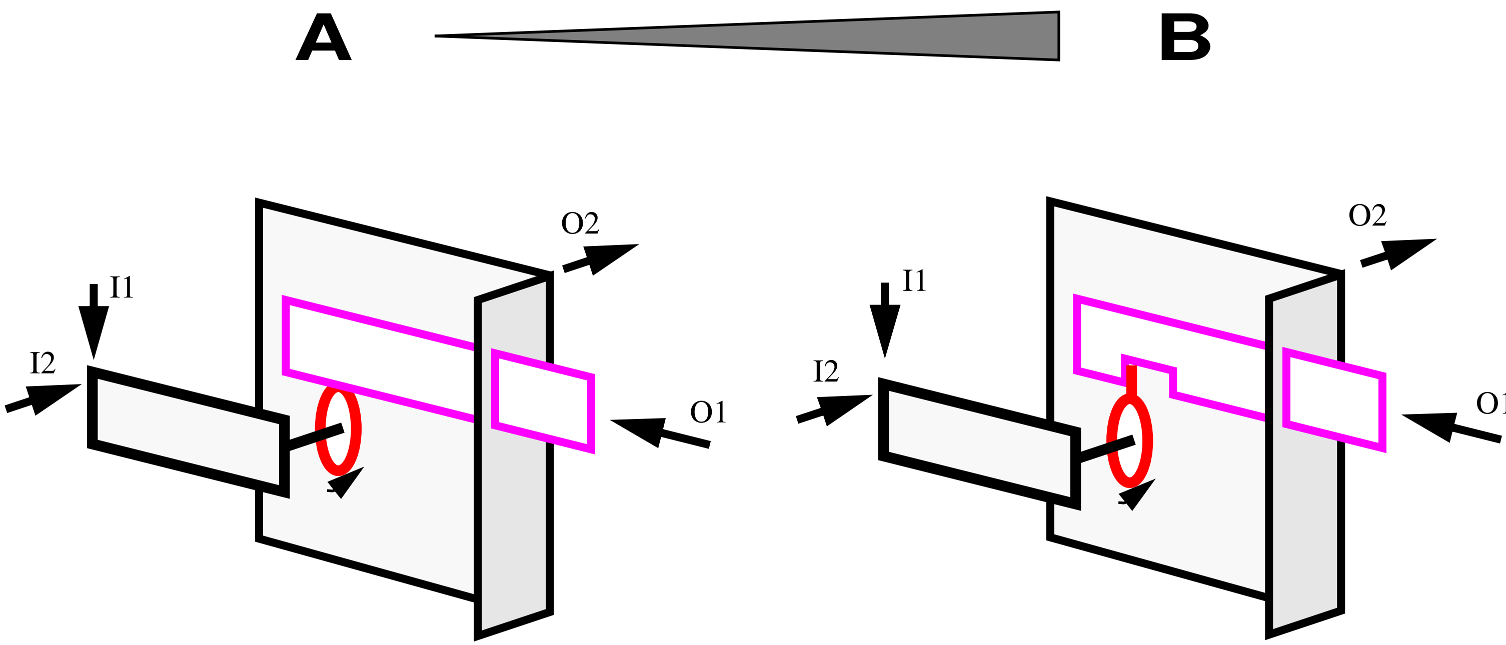
Door Locks





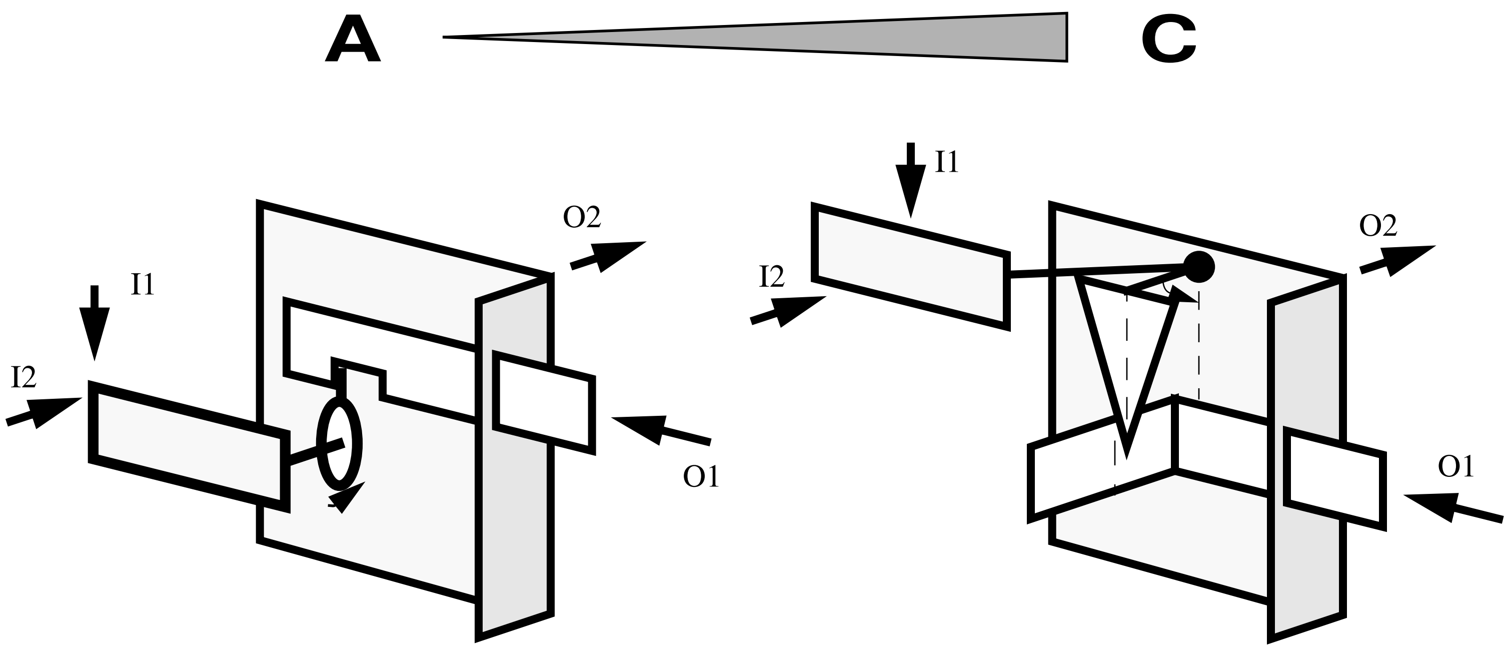
behavior of a door lock

Structurally Simpler



simpler components

Structurally Simpler



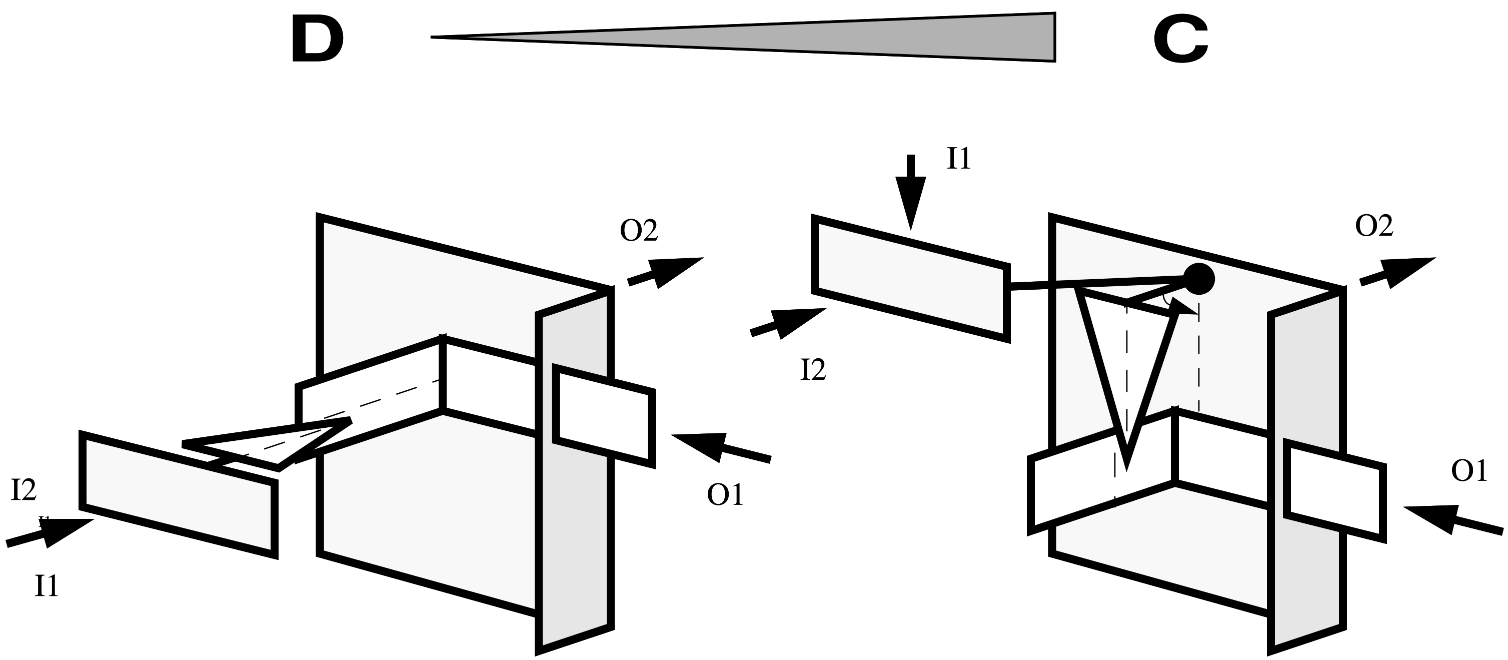
fewer components

Designs Representation

Designs are represented as:

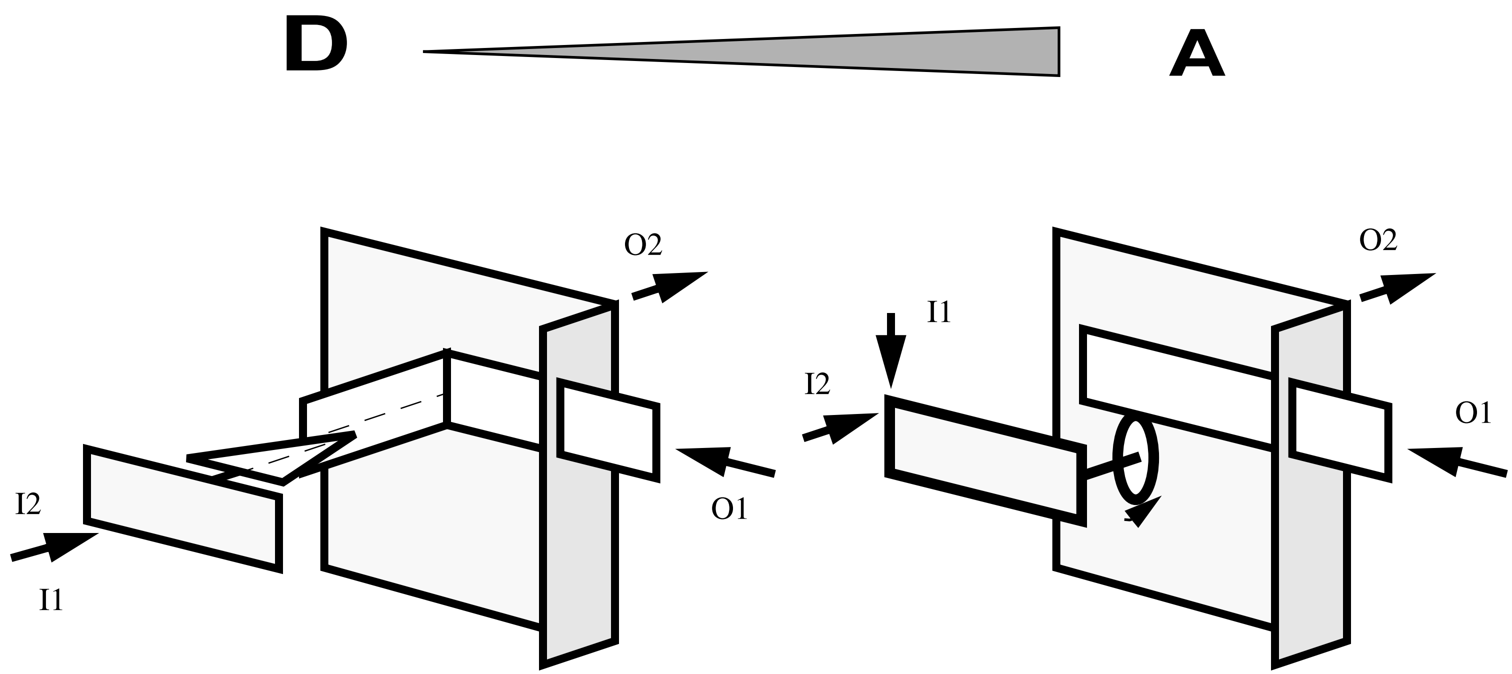
- Structure: attributes, components, relations
- Behavior: state transitions, causal chains
- Function: sequences of input-output pairs

Behaviorally Simpler



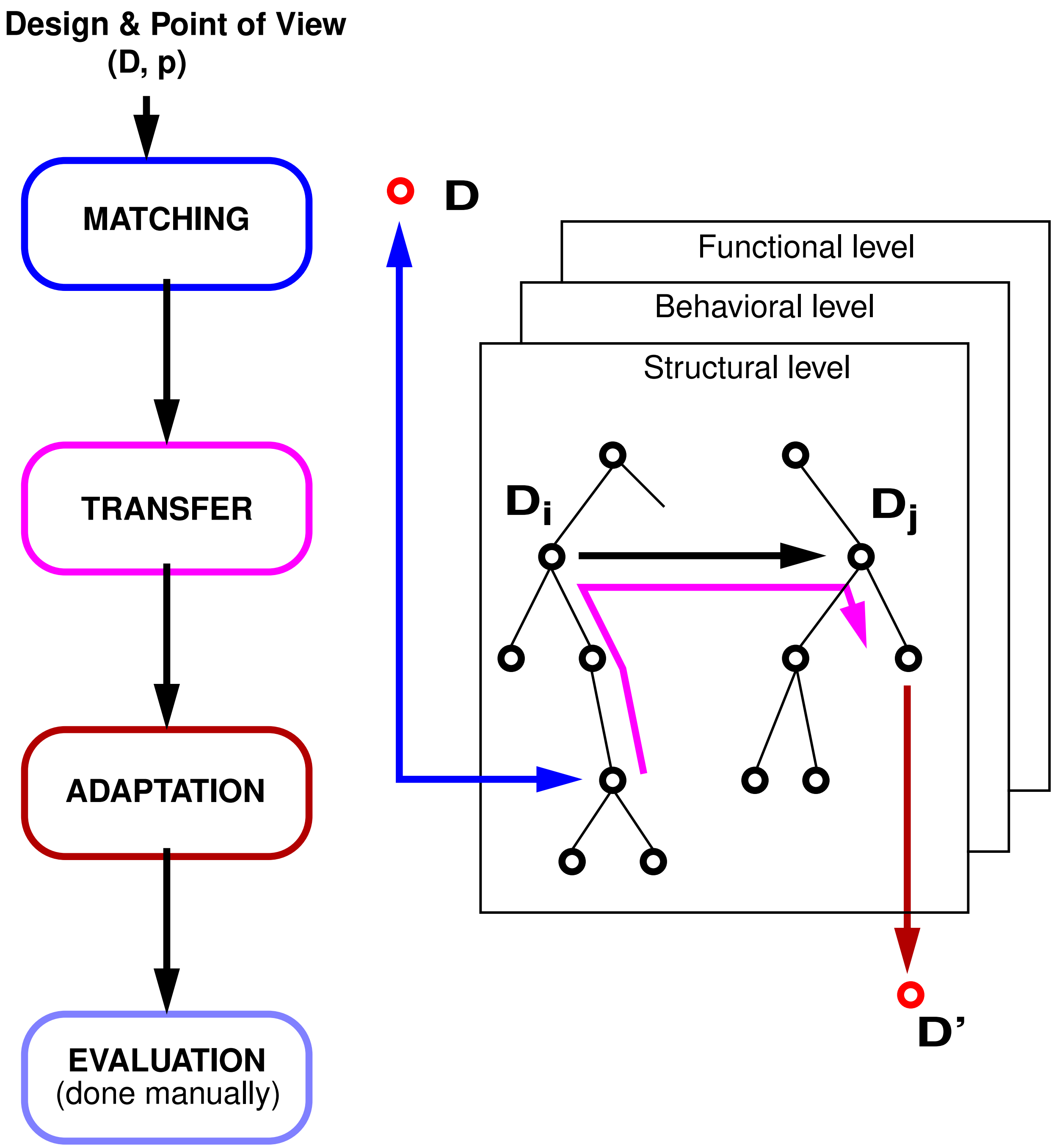
shorter causal chain

Functionally Simpler



simpler input combination

Simplification System



Key Issues

Abstraction

Where to use abstraction?

- **organizing the simplification database** into abstraction hierarchies - *abstractions over designs and simplifications*
- finding matches for designs - use of **goal-directed abstraction**
- **applying** "old" **simplification processes** to new simplification problems
- **generalizing** over "old" simplifications and "new" simplifications

Key Issues

Applying the Found Simplification

Ways to apply simplification:

1. **Transformational** - transfer and apply differences between the two designs involved in the simpler relation to the design to be simplified

   - use abstraction over the designs (directed by aspects relevant to the simplification)

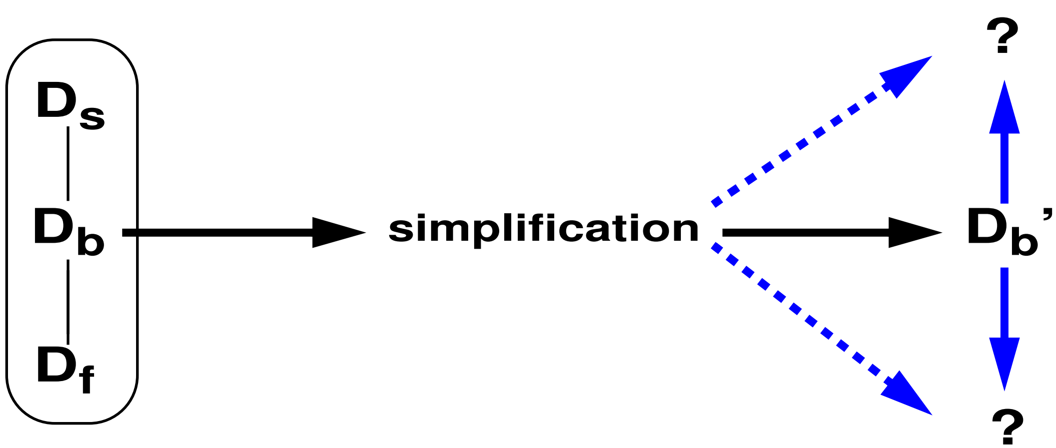
2. **Derivational** - transfer and apply the simplification process to the design to be simplified

   - use abstraction over the simplification process for transfer

Key Issues

Propagation of Simplifications

Simplification of a design at one level may result in modification of other levels

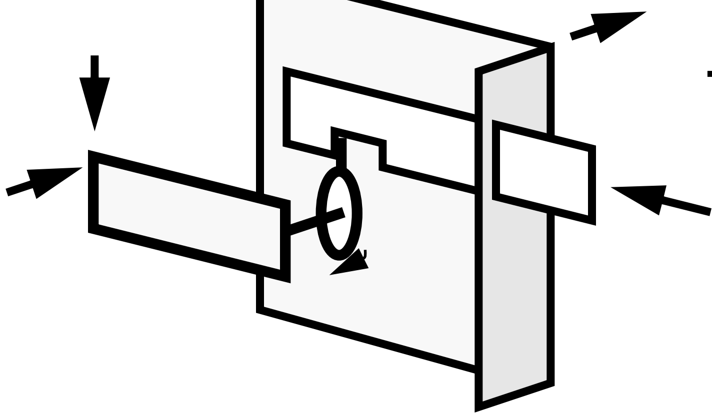


Problems:

- how does propagation happen
- building of a model of propagation

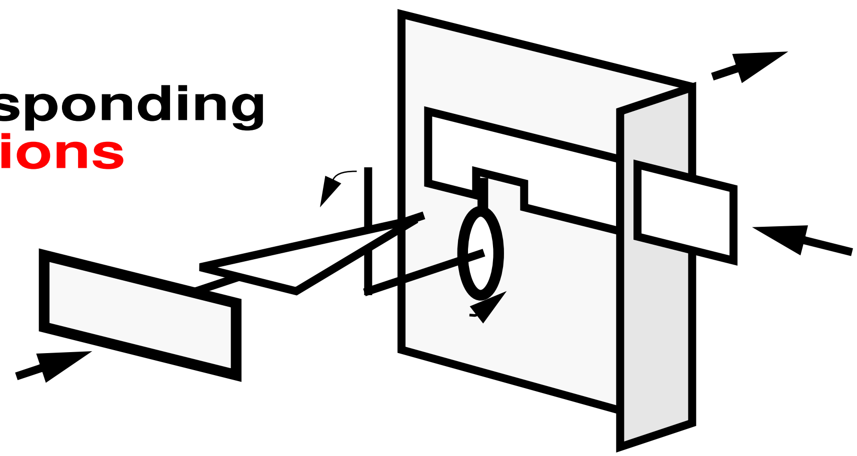
Key Isues - Propagation (continued)

A



To **functionally simplify** A find simplification that reduces inputs

and make the corresponding **structural modifications**



Research Directions

- Theory of design simplification (new)
- Propagation of design simplification across levels (new)
- Analogical reasoning about design behavior and function (new application)
- Using goal-driven abstraction in analogical reasoning (new application)
- Simplification by goal-based analogical reasoning (new application)
- Design simplification system (new)