Abstractions

- State-spaces often get very large, with many variables
- Often, many of the variables are irrelevant for the analysis to be performed,
- or only some particular property of the variable is relevant.
- **Idea:** Create a simpler model, which “abstracts away” the irrelevant detail in the original model.
- Ideally, the “abstracting away” should be performed automatically, by a “compiler”.
- The abstract model can be analyzed by a model-checker.
Example:

**Action System** *Dining Mathematicians*

**declare** \( n : \text{integer} \)**

**initially** \( n \geq 1 \)

\[
\begin{align*}
\text{think}_1 & \quad | \\
& \quad n := 3n + 1 \quad \text{odd}(n) \quad | \quad n := n/2 \quad \text{even}(n) \quad \text{even} \\
\text{eat}_1 & \quad | \\
& \quad | \\
\text{think}_2 & \\
\text{eat}_2
\end{align*}
\]

end

If we are only interested in control:

**Action System** *Abstract Mathematicians*

**declare** \( \text{even} : \text{boolean} \)**

**initially**

\[
\begin{align*}
\text{think}_1 & \quad | \\
& \quad \text{even} := \neg \text{even} \quad | \\
& \quad | \\
\text{think}_2 & \quad | \\
& \quad \text{even} := ? \quad \text{even} \quad | \\
\text{eat}_1 & \quad | \\
& \quad | \\
\text{eat}_2
\end{align*}
\]

end
How to relate the abstract to the more detailed?

• Let \( \langle S, S^0, \rightarrow \rangle \) be the detailed transition system,
  Let \( \langle T, T^0, \rightarrow \rangle \) be the abstract transition system,

• **Idea:** Each abstract state represents a set of concrete states.

• **Formally:** This is represented by a *concretization function* 
  \( \gamma : T \mapsto \mathcal{P}(S) \), which maps each abstract state to a set of 
  “concrete” states.

• We also want to “compute” on the set of abstract values. E.g., 
  we want to take “union”, “intersection”, and so on.

• Define an ordering \( \sqsubseteq \) on abstract states by 
  \[
  t_1 \sqsubseteq t_2 \iff \gamma(t_1) \subseteq \gamma(t_2)
  \]

• For each set \( F \subseteq S \), define a “best approximation” 
  \( \alpha(F) \in T \), such that 
  \[
  \alpha(\gamma(t)) = t \quad \text{for each } t \in T \\
  \gamma(\alpha(F)) \supseteq F \quad \text{for each } F \subseteq S
  \]

We say that \( (\alpha, \gamma) \) forms a *Galois Insertion* from \( (\mathcal{P}(S), \subseteq) \) 
  to \( (T, \sqsubseteq) \).
Approximating the Transition Relation

• The relation → is a safe abstraction of the relation → between concrete states if

whenever \( s \in \gamma(t) \) and \( s \rightarrow s' \)

there is \( t' \) such that \( s' \in \gamma(t') \) and \( t \rightarrow t' \)

• We say that \( \langle T, T^0, \rightarrow \rangle \) is a safe abstraction of \( \langle S, S^0, \rightarrow \rangle \) if

\( \rightarrow \) is a safe abstraction of \( \rightarrow \), and

\( \forall s^0 \in S^0. \exists t^0 \in T^0. s^0 \in \gamma t^0 \)

• **Theorem:** If \( \langle T, T^0, \rightarrow \rangle \) is a safe abstraction of \( \langle S, S^0, \rightarrow \rangle \), then for each computation

\[ s^0 \rightarrow s^1 \rightarrow s^2 \rightarrow s^3 \rightarrow s^4 \rightarrow s^5 \ldots \]

of \( \langle S, S^0, \rightarrow \rangle \), there is a computation

\[ t^0 \rightarrow t^1 \rightarrow t^2 \rightarrow t^3 \rightarrow t^4 \rightarrow t^5 \ldots \]

of \( \langle T, T^0, \rightarrow \rangle \) such that

\[ s_i \in \gamma(t_i) \]

for each \( i \).
Relating Concrete and Abstract Behaviors and their Properties

For a system, and a temporal property $\phi$, we want

$$\langle T, T^0, \rightarrow \rangle \models \phi \quad \text{implies} \quad \langle S, S^0, \rightarrow \rangle \models \phi$$

This can work, if we define

- For “simple” state formulas $p$:

  $$t \models p \quad \text{iff} \quad \forall s \in \gamma(t). \ s \models p$$

Note that “the law of excluded middle” need not hold!

- Extend the above definition to properties of behaviors in the natural way.

- **Theorem:**

  $$\langle T, T^0, \rightarrow \rangle \models \phi \quad \text{implies} \quad \langle S, S^0, \rightarrow \rangle \models \phi$$
Induced Operations and Transitions

Often, we want to compute an abstraction directly, based on the syntax.

- Assume that the original model has a set of actions (e.g., corresponding to “basic statements”)
- Assume that states are approximated in an abstract domain, as before:
- Approximate each action $A$ by its induced action
  \[ t \rightarrow \alpha(post(A, \gamma(t))) \]
- It is, more precisely, enough that we Approximate each action $A$ by an abstract action $B$ such that
  \[ \alpha(post(A, \gamma(t))) = post(B, t) \]
- **Example:** Let the set of concrete states be $\mathcal{N}$. Let the abstract domain be $\mathcal{P}\{\{even, odd\}\}$. Define

\[
\begin{align*}
\gamma(\text{even}) & = \{0, 2, 4, \ldots\} \\
\gamma(\text{odd}) & = \{1, 3, 5, \ldots\} \\
\gamma(\{\text{even, odd}\}) & = \mathcal{N} \\
\gamma(\emptyset) & = \emptyset
\end{align*}
\]

- **Induced Operations:** can be tabled as follows:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Result on even</th>
<th>Result on odd</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\cdot + 1$</td>
<td>odd</td>
<td>even</td>
</tr>
<tr>
<td>$3 * \cdot$</td>
<td>even</td>
<td>odd</td>
</tr>
<tr>
<td>$\cdot / 2$</td>
<td>${\text{even, odd}}$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>$\text{even}(\cdot)$</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>$\text{odd}(\cdot)$</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>
The table is extended to sets by

\[ \text{op}(t_1 \sqcup t_2) = \text{op}(t_1) \sqcup \text{op}(t_2) \]
Abstract System

Action System *Dining Mathematicians*

declare $b : \text{bit}$
initially $true$

\[
\begin{align*}
& \quad \text{think}_1 \\
& \quad b := \neg b \quad b = 1 \\
& \quad \text{eat}_1
\end{align*}
\]

\[
\begin{align*}
& \quad \text{think}_2 \\
& \quad b := ? \quad b = 0 \\
& \quad \text{eat}_2
\end{align*}
\]

end

**Problem:** Check whether

\[\neg(at \text{ eat}_1 \land at \text{ eat}_2)\]

is invariant.
What about Liveness and Fairness?