# Planning and Optimization

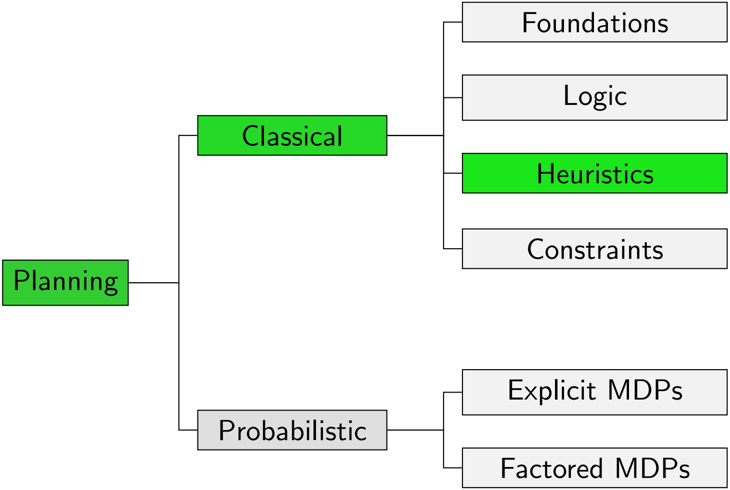
## D3. Abstractions: Additive Abstractions

Malte Helmert and Thomas Keller

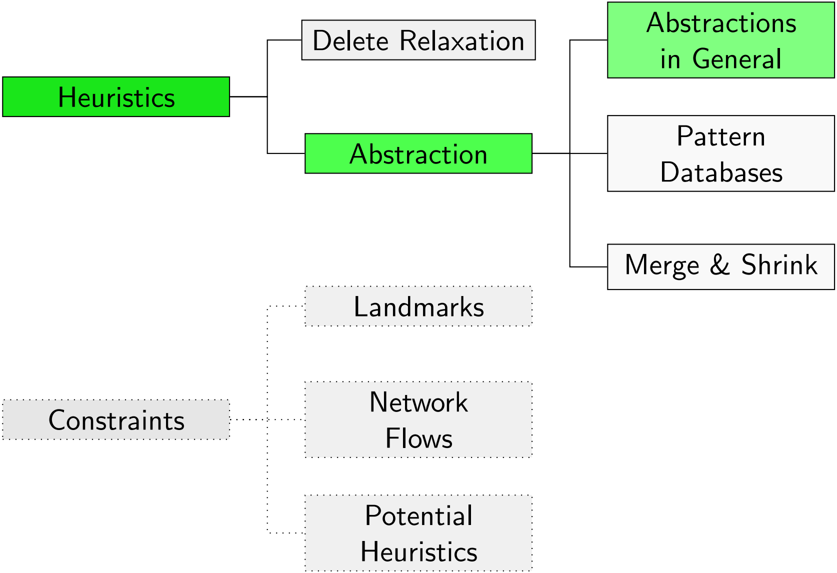
Universität Basel

October 30, 2019

# Content of this Course



# Content of this Course: Heuristics



# Additivity

# Orthogonality of Abstractions

**Definition (Orthogonal)**

Let $\alpha_1$ and $\alpha_2$ be abstractions of transition system $\mathcal{T}$.

We say that $\alpha_1$ and $\alpha_2$ are orthogonal if for all transitions $s \xrightarrow{\ell} t$ of $\mathcal{T}$, we have $\alpha_i(s) = \alpha_i(t)$ for at least one $i \in \{1, 2\}$.

# Affecting Transition Labels

**Definition (Affecting Transition Labels)**

Let $\mathcal{T}$ be a transition system, and let $\ell$ be one of its labels.

We say that $\ell$ affects $\mathcal{T}$ if $\mathcal{T}$ has a transition $s \xrightarrow{\ell} t$ with $s \neq t$.

**Theorem (Affecting Labels vs. Orthogonality)**

*Let $\alpha_1$ and $\alpha_2$ be abstractions of transition system $\mathcal{T}$.*

*If no label of $\mathcal{T}$ affects both $\mathcal{T}^{\alpha_1}$ and $\mathcal{T}^{\alpha_2}$,*
*then $\alpha_1$ and $\alpha_2$ are orthogonal.*

(Easy proof omitted.)

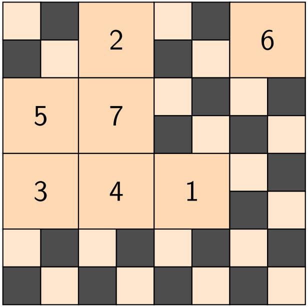
# Orthogonal Abstractions: Example

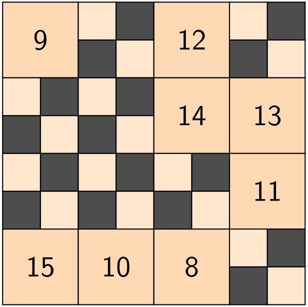
| | | | |
|---|---|---|---|
| | 2 | | 6 |
| 5 | 7 | | |
| 3 | 4 | 1 | |
| | | | |

| | | | |
|---|---|---|---|
| 9 | | 12 | |
| | | 14 | 13 |
| | | | 11 |
| 15 | 10 | 8 | |

Are the abstractions orthogonal?

# Orthogonal Abstractions: Example




Are the abstractions orthogonal?

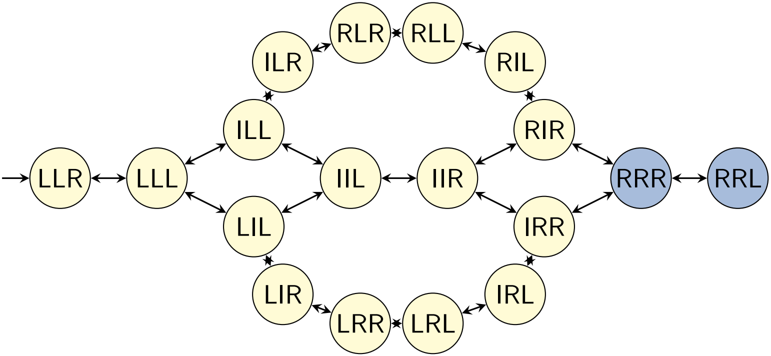
# Orthogonality and Additivity

**Theorem (Additivity for Orthogonal Abstractions)**

*Let $h^{\alpha_1}, \dots, h^{\alpha_n}$ be abstraction heuristics of the same transition system such that $\alpha_i$ and $\alpha_j$ are orthogonal for all $i \neq j$.*

*Then $\sum_{i=1}^{n} h^{\alpha_i}$ is a safe, goal-aware, admissible and consistent heuristic for $\Pi$.*

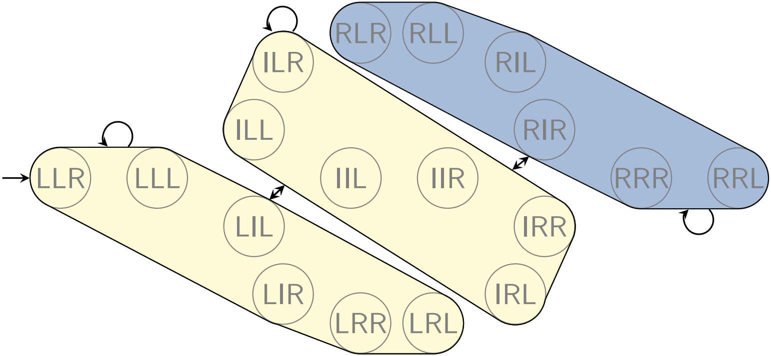
# Orthogonality and Additivity: Example




transition system $\mathcal{T}$

state variables: first package, second package, truck

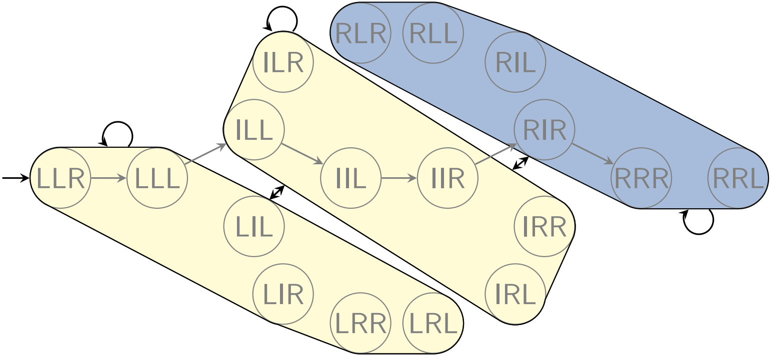
# Orthogonality and Additivity: Example




abstraction $\alpha_1$

abstraction: only consider value of first package

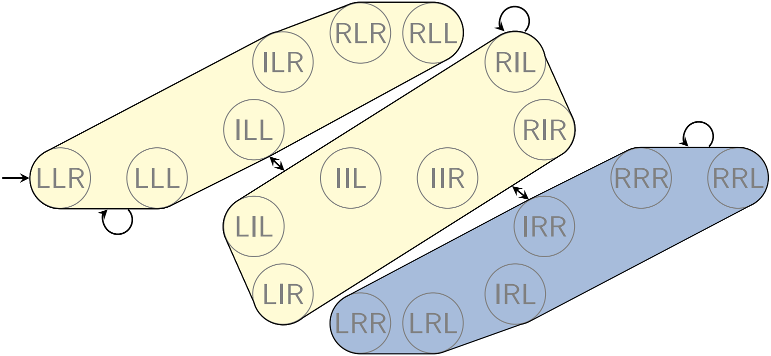
# Orthogonality and Additivity: Example




abstraction $\alpha_1$

abstraction: only consider value of first package

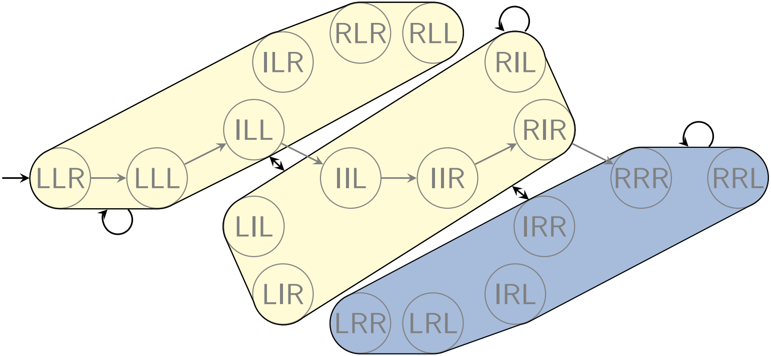
# Orthogonality and Additivity: Example




abstraction $\alpha_2$ (orthogonal to $\alpha_1$)

abstraction: only consider value of second package

# Orthogonality and Additivity: Example




abstraction $\alpha_2$ (orthogonal to $\alpha_1$)

abstraction: only consider value of second package

# Orthogonality and Additivity: Proof (1)

**Proof.**

We prove goal-awareness and consistency;
the other properties follow from these two.

Let $\mathcal{T} = \langle S, L, c, T, s_0, S_\star \rangle$ be the concrete transition system.

Let $h = \sum_{i=1}^{n} h^{\alpha_i}$.

# Orthogonality and Additivity: Proof (1)

**Proof.**

We prove goal-awareness and consistency;
the other properties follow from these two.

Let $\mathcal{T} = \langle S, L, c, T, s_0, S_\star \rangle$ be the concrete transition system.

Let $h = \sum_{i=1}^{n} h^{\alpha_i}$.

Goal-awareness: For goal states $s \in S_\star$,
$h(s) = \sum_{i=1}^{n} h^{\alpha_i}(s) = \sum_{i=1}^{n} 0 = 0$ because all individual abstraction heuristics are goal-aware. ...

# Orthogonality and Additivity: Proof (2)

**Proof (continued).**

Consistency: Let $s \xrightarrow{o} t \in T$. We must prove $h(s) \leq c(o) + h(t)$.

# Orthogonality and Additivity: Proof (2)

**Proof (continued).**

Consistency: Let $s \xrightarrow{o} t \in T$. We must prove $h(s) \leq c(o) + h(t)$.

Because the abstractions are orthogonal, $\alpha_i(s) \neq \alpha_i(t)$ for at most one $i \in \{1, \dots, n\}$.

# Orthogonality and Additivity: Proof (2)

**Proof (continued).**

Consistency: Let $s \xrightarrow{o} t \in T$. We must prove $h(s) \leq c(o) + h(t)$.

Because the abstractions are orthogonal, $\alpha_i(s) \neq \alpha_i(t)$ for at most one $i \in \{1, \dots, n\}$.

Case 1: $\alpha_i(s) = \alpha_i(t)$ for all $i \in \{1, \dots, n\}$.

# Orthogonality and Additivity: Proof (2)

## Proof (continued).

Consistency: Let $s \xrightarrow{o} t \in T$. We must prove $h(s) \leq c(o) + h(t)$.

Because the abstractions are orthogonal, $\alpha_i(s) \neq \alpha_i(t)$ for at most one $i \in \{1, \ldots, n\}$.

Case 1: $\alpha_i(s) = \alpha_i(t)$ for all $i \in \{1, \ldots, n\}$.

Then $h(s) = \sum_{i=1}^{n} h^{\alpha_i}(s)$

$$
\begin{aligned}
&= \sum_{i=1}^{n} h^*_{\mathcal{T}^{\alpha_i}}(\alpha_i(s)) \\
&= \sum_{i=1}^{n} h^*_{\mathcal{T}^{\alpha_i}}(\alpha_i(t)) \\
&= \sum_{i=1}^{n} h^{\alpha_i}(t) \\
&= h(t) \leq c(o) + h(t).
\end{aligned}
$$

...

# Orthogonality and Additivity: Proof (3)

**Proof (continued).**

Case 2: $\alpha_i(s) \neq \alpha_i(t)$ for exactly one $i \in \{1, \dots, n\}$.

Let $k \in \{1, \dots, n\}$ such that $\alpha_k(s) \neq \alpha_k(t)$.

# Orthogonality and Additivity: Proof (3)

**Proof (continued).**

Case 2: $\alpha_i(s) \neq \alpha_i(t)$ for exactly one $i \in \{1, \dots, n\}$.

Let $k \in \{1, \dots, n\}$ such that $\alpha_k(s) \neq \alpha_k(t)$.

Then
$$\begin{aligned} h(s) &= \textstyle\sum_{i=1}^{n} h^{\alpha_i}(s) \\ &= \textstyle\sum_{i \in \{1,\dots,n\} \setminus \{k\}} h^*_{\mathcal{T}^{\alpha_i}}(\alpha_i(s)) + h^{\alpha_k}(s) \\ &\leq \textstyle\sum_{i \in \{1,\dots,n\} \setminus \{k\}} h^*_{\mathcal{T}^{\alpha_i}}(\alpha_i(t)) + c(o) + h^{\alpha_k}(t) \\ &= c(o) + \textstyle\sum_{i=1}^{n} h^{\alpha_i}(t) \\ &= c(o) + h(t), \end{aligned}$$
where the inequality holds because $\alpha_i(s) = \alpha_i(t)$ for all $i \neq k$ and $h^{\alpha_k}$ is consistent. □

# Outlook

# Using Abstraction Heuristics in Practice

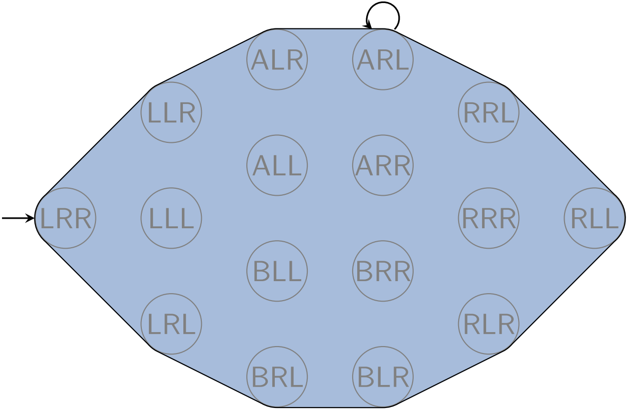
In practice, there are conflicting goals for abstractions:

- we want to obtain an informative heuristic, but
- want to keep its representation small.

Abstractions have small representations if

- there are few abstract states and
- there is a succinct encoding for $\alpha$.

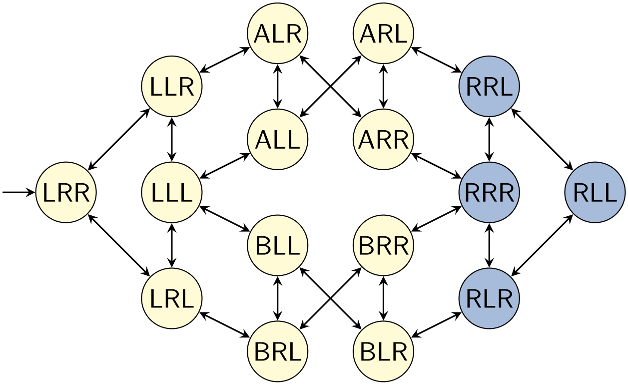
# Counterexample: One-State Abstraction




One-state abstraction: $\alpha(s) :=$ const.

- $+$ very few abstract states and succinct encoding for $\alpha$
- $-$ completely uninformative heuristic

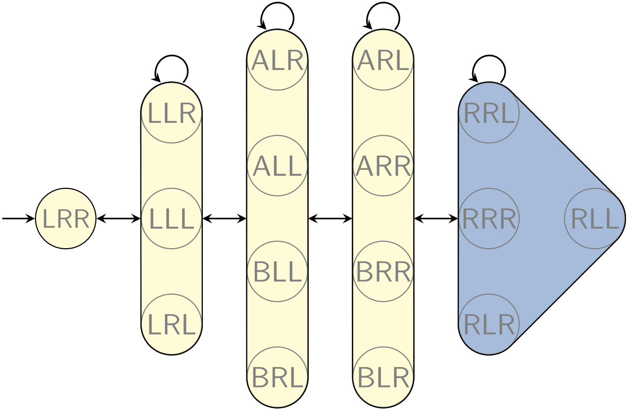
# Counterexample: Identity Abstraction




Identity abstraction: $\alpha(s) := s$.

- $+$ perfect heuristic and succinct encoding for $\alpha$
- $-$ too many abstract states

# Counterexample: Perfect Abstraction




Perfect abstraction: $\alpha(s) := h^*(s)$.

- $+$ perfect heuristic and usually few abstract states
- $-$ usually no succinct encoding for $\alpha$

# Automatically Deriving Good Abstraction Heuristics

**Abstraction Heuristics for Planning: Main Research Problem**

Automatically derive effective abstraction heuristics for planning tasks.

⇝ we will study two state-of-the-art approaches in Chapters D4–D8

# Summary

# Summary

- Abstraction heuristics from orthogonal abstractions can be added without losing admissibility or consistency.
- One sufficient condition for orthogonality is that all abstractions are affected by disjoint sets of labels.
- Practically useful abstractions are those which give informative heuristics, yet have a small representation.
- Coming up with good abstractions automatically is the main research challenge when applying abstraction heuristics in planning.