Mechanically Proving Guarantees of Generalized Heuristics: First Results and Ongoing Work

Mohammad Abdulaziz$^{1, 2}$
Florian Pommerening$^{3}$    Augusto B. Corrêa$^{3}$

$^1$King’s College London, United Kingdom,
$^2$Technische Universität München, Germany,
$^3$University of Basel, Switzerland

GenPlan’22
Primitive Concepts & Roles

- $ontable = \{\text{blue}, \text{green}\}$
- $on = \{(\text{red}, \text{green})\}$
- $holding = \emptyset$
- $clear = \{\text{blue}, \text{red}\}$
- $clear_G = \{\text{green}\}$
Example: Clearing a Block

Definition (Generalized Potential Heuristic)
Linear combination of features well-defined over all tasks:

\[ h(s) = \sum_{f \in F} w(f) \cdot f(s) \]

Generalized Potential Heuristic for Blocksworld

Blocksworld tasks where the goal is to clear a set of blocks

\[ h(s) = 2 \cdot |C_1| + |C_2| \]

- \( C_1 \equiv \exists on^+.clear_G \):
  "Set of blocks above some block that needs to be cleared"

- \( C_2 \equiv holding \):
  "Set of blocks being held"
generalized potential heuristics with **performance guarantees**:
- heuristics that lead search directly to goal state (i.e., DDA)
- exist for many domains
- can be learned from given examples
- *pen-and-paper proofs* of generalization
generalized potential heuristics with performance guarantees:

- heuristics that lead search directly to goal state (i.e., DDA)
- exist for many domains
- can be learned from given examples
- pen-and-paper proofs of generalization

our goal: automate these proofs as much as possible
Interactive Proof using \textit{Isabelle/HOL}

Is $h$ DDA in any possible task of this domain?
heuristics representing *tiered-measures of progress*:

- order between concepts used, from “best” to “worst”
- object can be at (max.) one concept
- move object to better concept = make search progress

invariants are given:

- assume they are provided
- prove that they are invariants
- related: Bonet et al. (IJCAI 2019)
heuristics representing tiered-measures of progress:
- order between concepts used, from “best” to “worst”
- object can be at (max.) one concept
- move object to better concept = make search progress

invariants are given:
- assume they are provided
- prove that they are invariants
- related: Bonet et al. (IJCAI 2019)

most importantly: so far we only implemented Miconic
Input

Background and domain theories
Input

Background and domain theories

Find and prove invariants

- No features occurs twice in \( h \)
- Concepts are mutually exclusive
- Case distinction over which concepts are empty
- Find a witness action
- No new object appears during transition
- No object moves to a worse concept
- One object (at least) moves to a better concept for each case

Steps above imply \( h \) is descending
Input

Background and domain theories

Find and prove invariants

- #features is finite
- No features occurs twice in $h$

Steps above imply $h$ is descending
Background and domain theories

Find and prove invariants

#features is finite

No features occurs twice in $h$

Concepts are mutually exclusive

Input
Background and domain theories

Find and prove invariants

- \#features is finite
- No features occurs twice in \( h \)

Concepts are mutually exclusive

Case distinction over which concepts are empty

For each case:

- Find a witness action

  - No new object appears during transition
  - No object moves to a worse concept
  - One object (at least) moves to a better concept
Input

Background and domain theories

Find and prove invariants

#features is finite  No features occurs twice in $h$

Concepts are mutually exclusive

Case distinction over which concepts are empty

for each case

Find a witness action

No new object appears during transition

No object moves to a worse concept

One object (at least) moves to a better concept

Steps above imply $h$ is descending
what we have:

- interactive proof for Miconic domain
- parts of the proof are already fully automated
- thousands of lines of Isabelle/HOL theory
  - great part of it is generated automatically by scripts
  - in a new domain, we expect the user to do very little

next steps:

- finish automation of the proof as much as possible
- use for other domains
- other types of generalized potential heuristics