On the Complexity of Heuristic Synthesis for Satisficing Classical Planning: Potential Heuristics and Beyond

Malte Helmert Silvan Sievers
Alexander Rovner Augusto B. Corrêa

University of Basel, Switzerland

ICAPS 2022

Classical Planning Tasks

SAS$^+$ representation:

- state variables:
  $x \in \{0, 1, 2\}$, $y \in \{0, 1\}$, $z \in \{0, 1, 2\}$
- initial state:
  $\{x \mapsto 0, y \mapsto 0, z \mapsto 0\}$
- goal:
  $\{x \mapsto 2, z \mapsto 1\}$
- actions:
  $a_1 : \quad x \mapsto 0,\ y \mapsto 0 \quad \rightarrow \quad y := 1,\ z := 2$
  $a_2 : \quad x \mapsto 0,\ z \mapsto 1 \quad \rightarrow \quad z := 0$

Potential Heuristics

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$

features and weights

Potential Heuristics

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$

features and weights

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$

features and weights

Potential Heuristics

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$

features and weights

$h_1(\{x \mapsto 1, y \mapsto 1, z \mapsto 1\}) = 3 + 2 = 5$

Potential Heuristics

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$ dimension 1

features and weights

$h_1(\{x \mapsto 1, y \mapsto 1, z \mapsto 1\}) = 3 + 2 = 5$

Potential Heuristics

- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$ dimension 1
- h_2: $3\,[x = 0 \wedge y = 0] + 3\,[z = 1] - 2$ dimension 2

features and weights

$h_1(\{x \mapsto 1, y \mapsto 1, z \mapsto 1\}) = 3 + 2 = 5$

Potential Heuristics

- h_0: 7 — dimension 0
- h_1: $3\,[x = 1] - 3\,[y = 0] + 2\,[z = 1]$ — dimension 1
- h_2: $3\,[x = 0 \wedge y = 0] + 3\,[z = 1] - 2$ — dimension 2

features and weights

$$h_1(\{x \mapsto 1, y \mapsto 1, z \mapsto 1\}) = 3 + 2 = 5$$

Verification and Synthesis

study computational complexity of:

Is a given potential heuristic nice? (verification)

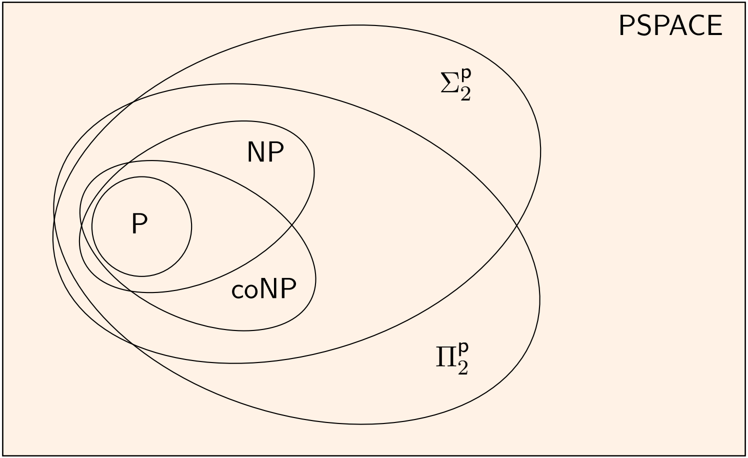
Does a nice potential heuristic exist? (synthesis)

Related Work

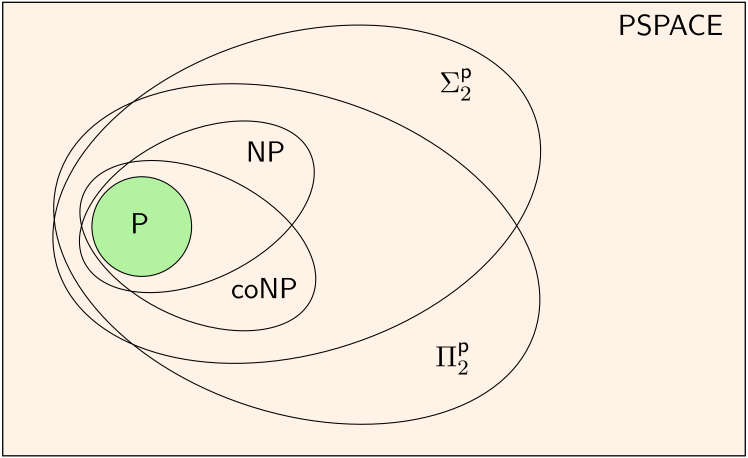
most closely related:

- Pommerening, Helmert, Röger and Seipp (AAAI 2015):
  polynomial algorithm for verification and synthesis of admissible and consistent potential heuristics of dimension 1
- Pommerening, Helmert and Bonet (AAAI 2017):
  extend this to dimension 2;
  verification of consistency coNP-complete for dimension 3

Complexity Classes

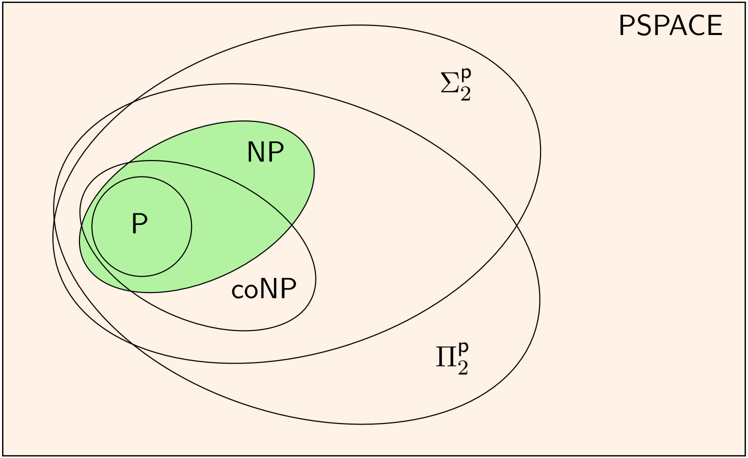


Complexity Classes



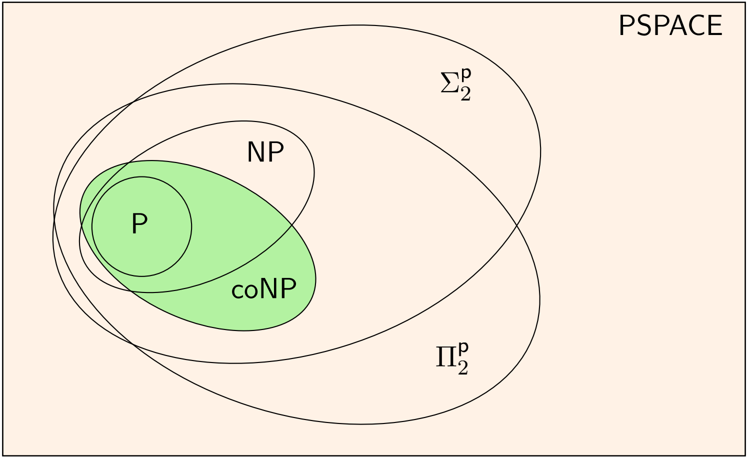
P: solvable in polynomial time

Complexity Classes



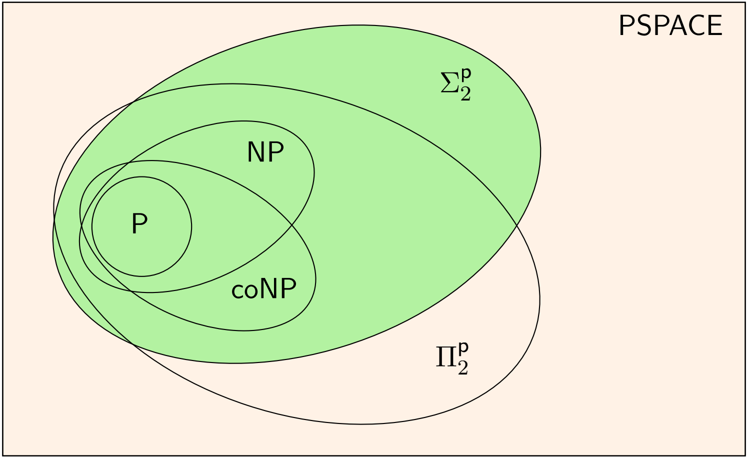
NP: “yes” answers have simple proofs

Complexity Classes



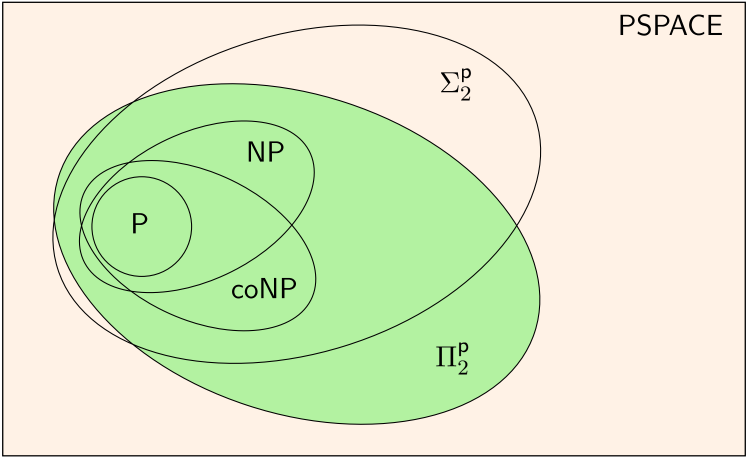
coNP: "no" answers have simple proofs

Complexity Classes



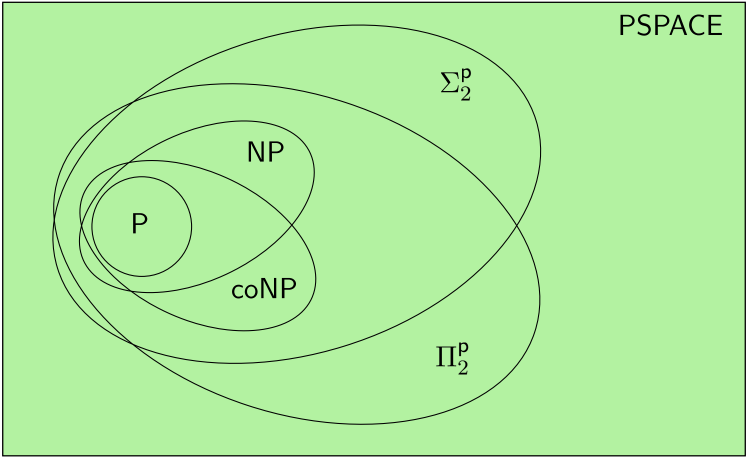
Σ_2^p: "yes" answers have simple proofs given an NP oracle

Complexity Classes



Π_2^p: "no" answers have simple proofs given an NP oracle

Complexity Classes



PSPACE: solvable in polynomial space

Part 1: DDA

The DDA Property

DDA property of a heuristic (Seipp et al. 2016)

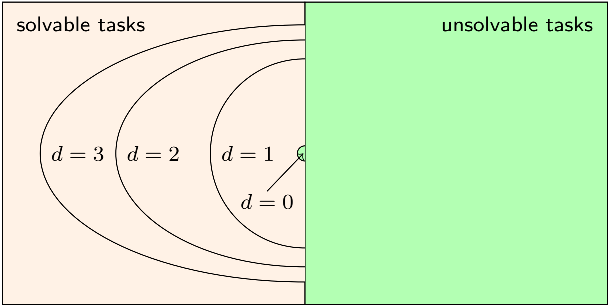
for all alive states:

- descending: there is a successor $h(s') < h(s)$
- dead-end-avoiding: dead successors have $h(s') \geq h(s)$

Seipp et al. 2016:

- h is DDA $\rightsquigarrow$ greedy BFS and hill-climbing backtrack-free
- low-dimension DDA potential heuristics are common

DDA Verification



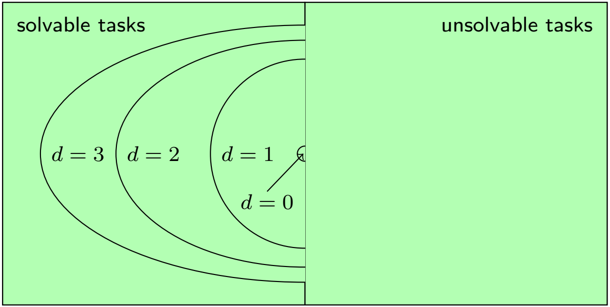
DDA verification for dimension 0

Theorem:

DDA verification for potential heuristics is PSPACE-complete.

This already holds for dimension 0.

DDA Synthesis



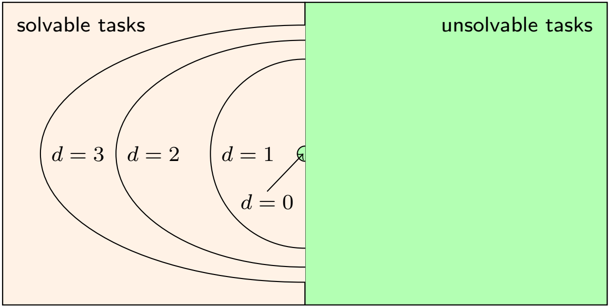
DDA synthesis for unrestricted potential heuristics

Theorem:

DDA synthesis for unrestricted potential heuristics is in P.

It is PSPACE-complete for restricted cases, such as dimension 0.

DDA Synthesis



DDA synthesis for dimension 0

Theorem:

DDA synthesis for unrestricted potential heuristics is in P.

It is PSPACE-complete for restricted cases, such as dimension 0.

Are We Supposed to Find This Interesting?

Did we learn anything about potential heuristic synthesis?

DDA combines "solvable without backtracking"
with "unsolvable for any reason"

$\rightsquigarrow$ need a better property

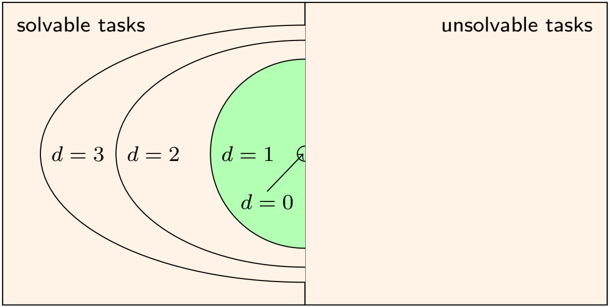
Part 2: SDDA

The SDDA Property

SDDA (solvable DDA) property of a heuristic:

- heuristic is DDA, and
- initial state is alive

SDDA Verification and Synthesis



SDDA verification/synthesis for dimension 1

Theorem:

SDDA verification for potential heuristics is in P for dimension 0 and PSPACE-complete for dimension 1 or higher.

The same is true for SDDA synthesis.

Is This More Interesting?

nice:

- no more trivial results; actual insight in reductions
- dimension-1 heuristics can solve PSPACE-complete problems

not so nice:

- everything interesting is hard

reason:

- SDDA property essentially requires perfect dead-end detection

$\rightsquigarrow$ relax this requirement

Part 3: ∞DDA

The ∞DDA Property

∞DDA property:
replace "alive state" with "state with finite h" in SDDA definition

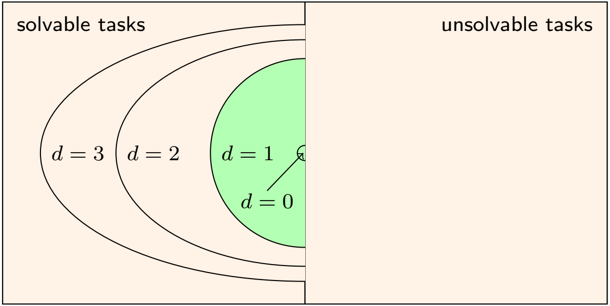
- states with finite h value have improving successor
- initial state has finite h value

also in paper:

- unrestricted DDA (UDDA)
- predicate-based pruning DDA (PDDA)

$\rightsquigarrow$ same properties as ∞DDA

∞DDA Verification and Synthesis



∞DDA verification/synthesis for dimension 1

Theorem:

∞DDA verification for potential heuristics is coNP-complete for dimension 1 or higher.

∞DDA synthesis for potential heuristics is Σ_2^{p}-complete for dimension 1 or higher.

This is What We Were Really Looking For

good news:

- this is what we hoped for:
  well below PSPACE, yet still very expressive
- interesting connection to $\exists\forall$QBF

bad news:

- no tractability for low dimension
  (unlike admissibility and consistency)

Beyond & Discussion

OK, Time to 'Fess Up!

I've been lying to you: we did not actually prove these results.

- proved something less satisfying:
  some results need "compact" heuristics
  (polynomial number of bits for feature weights)
- required for "guess-and-check" to be in NP

working on it $\rightsquigarrow$ stay tuned!

Beyond

beyond potential heuristics:

- membership results generalize to any polynomial-space representable, polynomial-time computable heuristic
- hardness results generalize to anything at least as powerful as potential heuristics of dimension 1

$\rightsquigarrow$ link to Kolmogorov complexity and universal search algorithms

What Next?

possible future directions:

- get rid of technical restrictions
- synthesize ∞DDA heuristics
- relationship to generalized representations

Thank You for Your Attention!