Learning Inadmissible Heuristics During Search

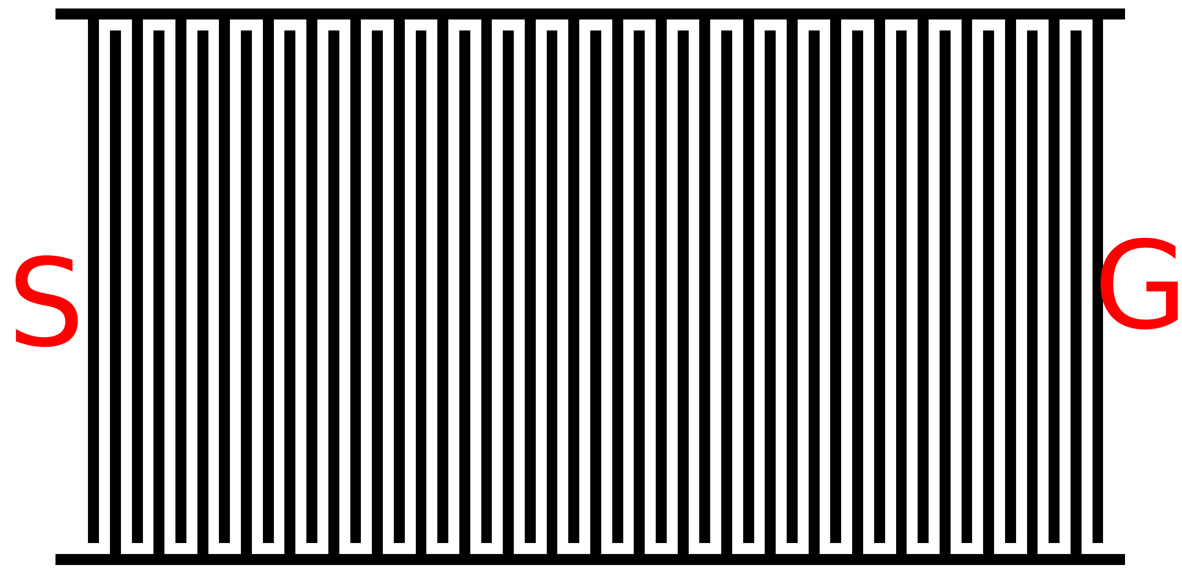
Jordan Thayer, Austin Dionne, and Wheeler Ruml

University of New Hampshire

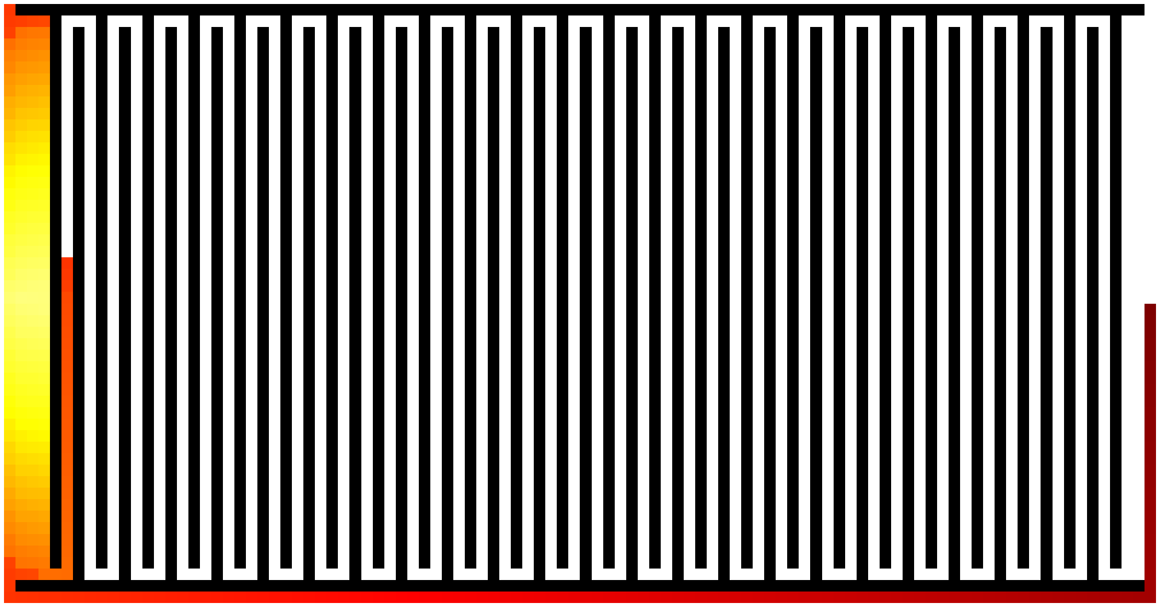
jtd7, ajv2, ruml at cs.unh.edu

With thanks to NSF Grant IIS-0812141
and DARPA CSSG Grant N10AP20029

Heuristics Guide Search

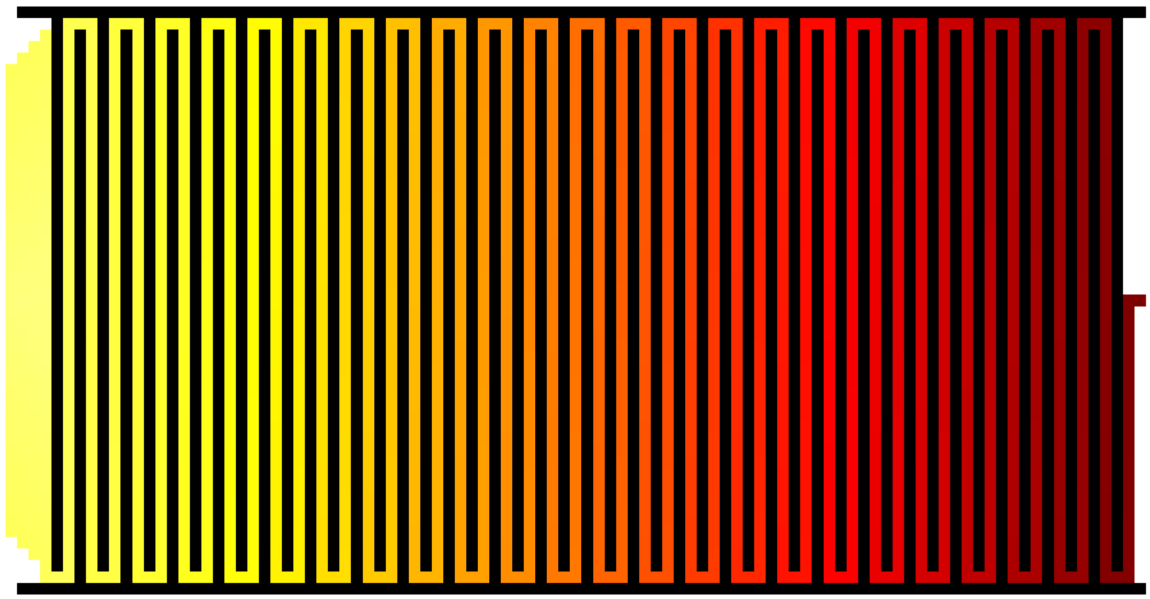


A*



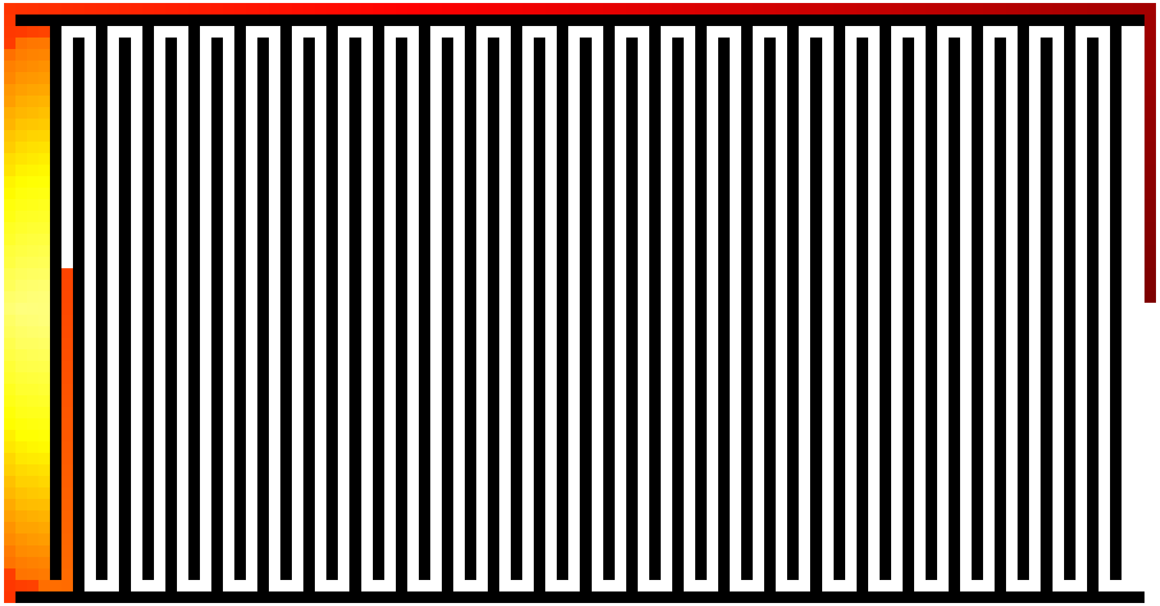
Heuristics Guide Search

greedy best-first search



Heuristics Guide Search

greedy best-first search with learning



Motivation For Our Approach

goal: work out of the box on single instances

- avoid offline training
- avoid domain specific features
- rely on data easily available in any best-first search

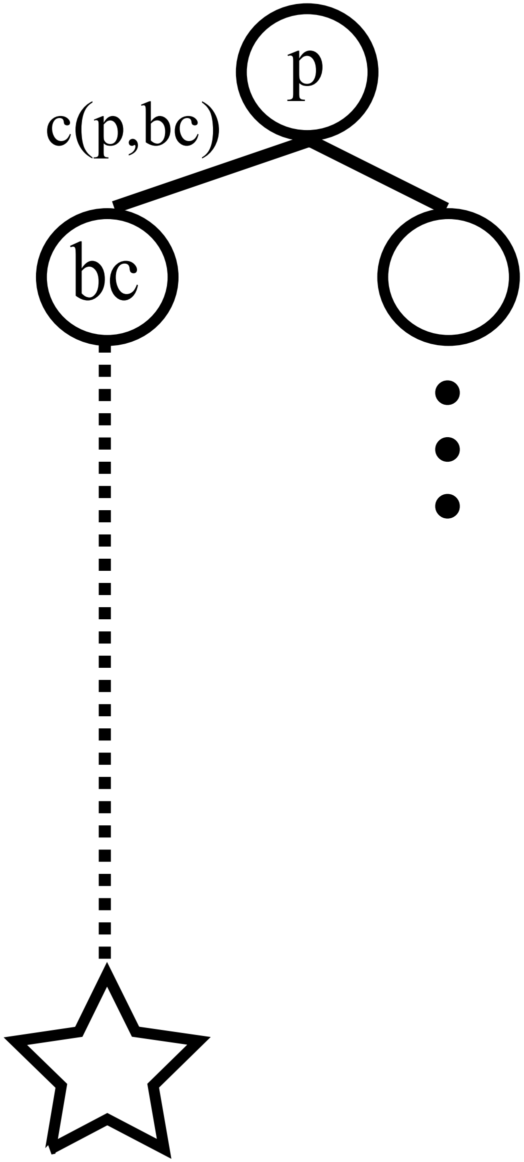
boost any suboptimal search

Outline

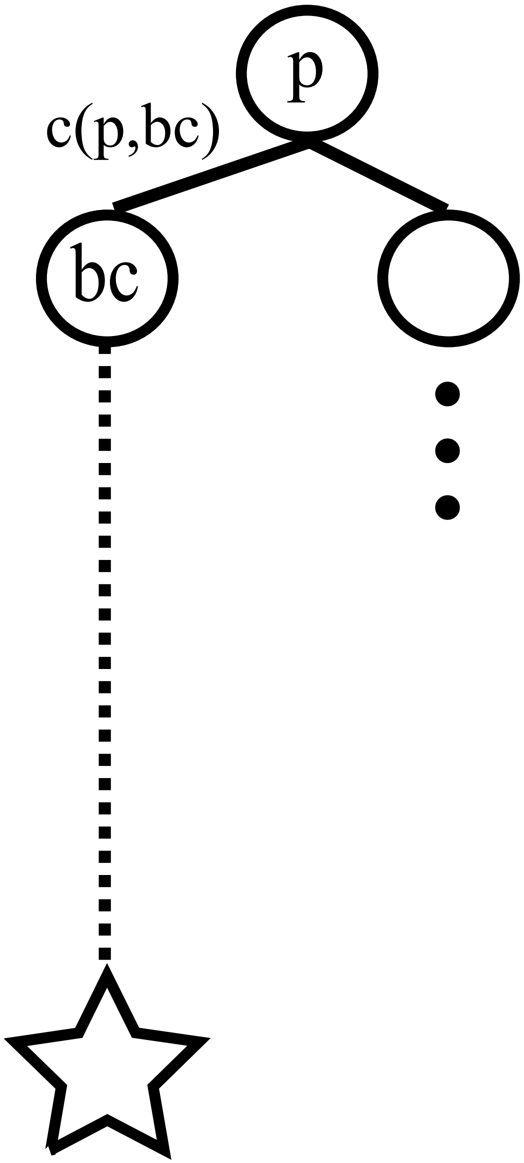
- motivation
- building inadmissible heuristics during search
  - observing error
  - correcting for error
- performance of learned heuristics
  - suboptimal search - greedy best-first search
  - bounded suboptimal search - skeptical search

Learning

Observing Error Between Parent and Best Child

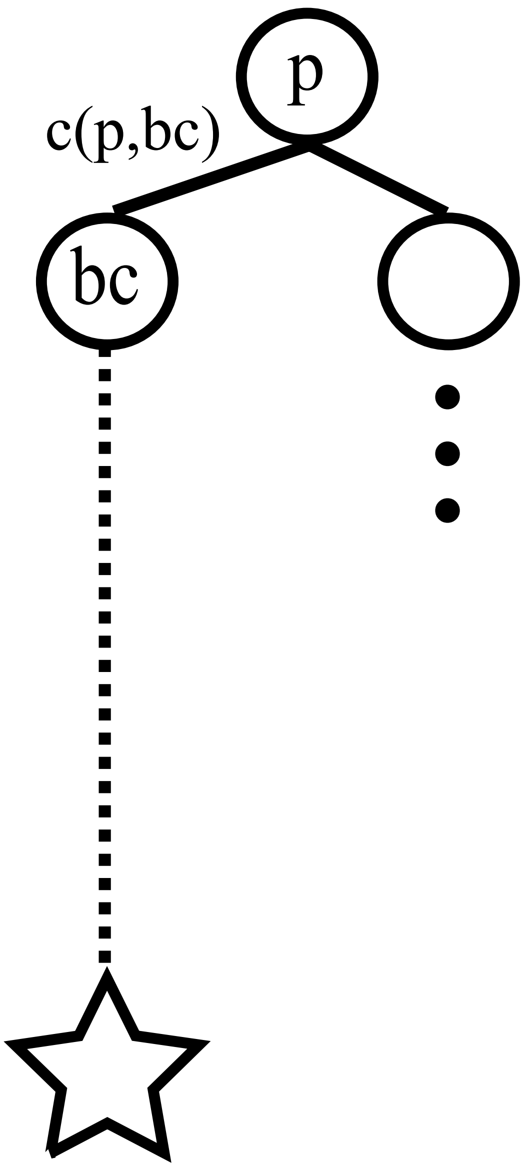


Observing Error Between Parent and Best Child



$f(p)$ should equal $f(bc)$

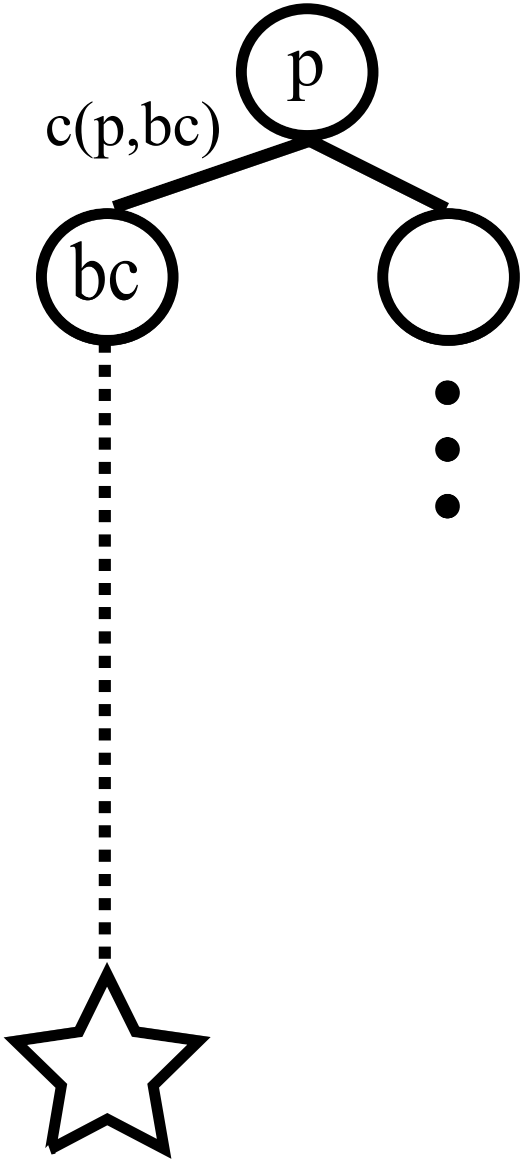
Observing Error Between Parent and Best Child



$f(p)$ should equal $f(bc)$

$$
\begin{aligned}
f^*(p) &= f^*(bc) \\
g(p) + h^*(p) &= g(bc) + h^*(bc) \\
h^*(p) &= h^*(bc) + c(p, bc)
\end{aligned}
$$

Observing Error Between Parent and Best Child

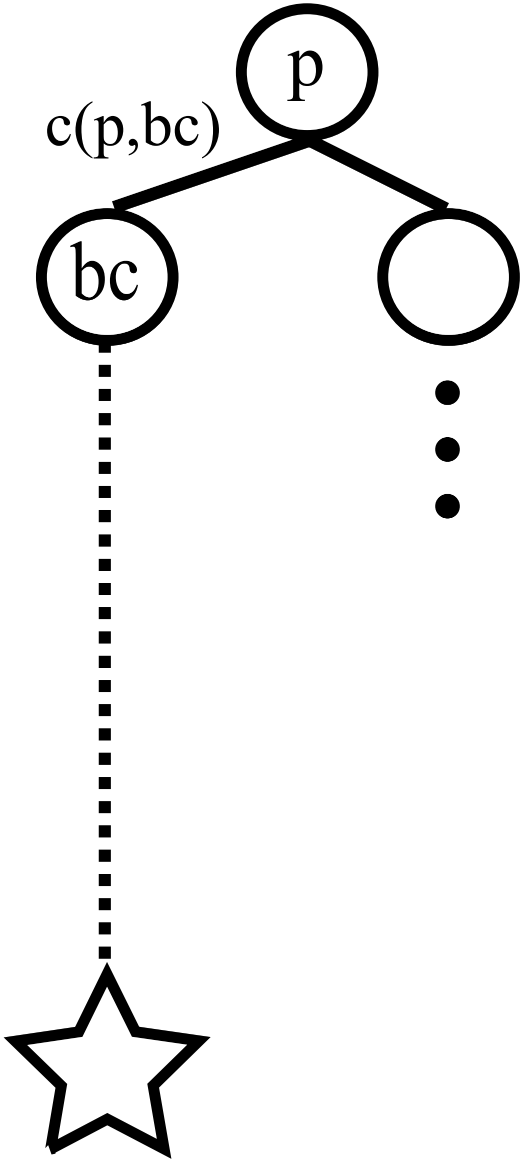


$f(p)$ should equal $f(bc)$

$$\begin{aligned} f^*(p) &= f^*(bc) \\ g(p) + h^*(p) &= g(bc) + h^*(bc) \\ h^*(p) &= h^*(bc) + c(p, bc) \end{aligned}$$

$$\begin{aligned} h(p) &= h(bc) + c(p, bc) - \epsilon_h \\ \epsilon_h &= h(bc) + c(p, bc) - h(p) \end{aligned}$$

Observing Error Between Parent and Best Child



$f(p)$ should equal $f(bc)$

$$
\begin{aligned}
f^*(p) &= f^*(bc) \\
g(p) + h^*(p) &= g(bc) + h^*(bc) \\
h^*(p) &= h^*(bc) + c(p, bc)
\end{aligned}
$$

$$
\begin{aligned}
h(p) &= h(bc) + c(p, bc) - \epsilon_h \\
\epsilon_h &= h(bc) + c(p, bc) - h(p)
\end{aligned}
$$

$$
\begin{aligned}
\widehat{h}(n) &= h(n) + \bar{\epsilon_h} \cdot d(n) \\
\widehat{h}(n) &= h(n) + \bar{\epsilon_h} \cdot \widehat{d}(n)
\end{aligned}
$$

Path Based Corrections

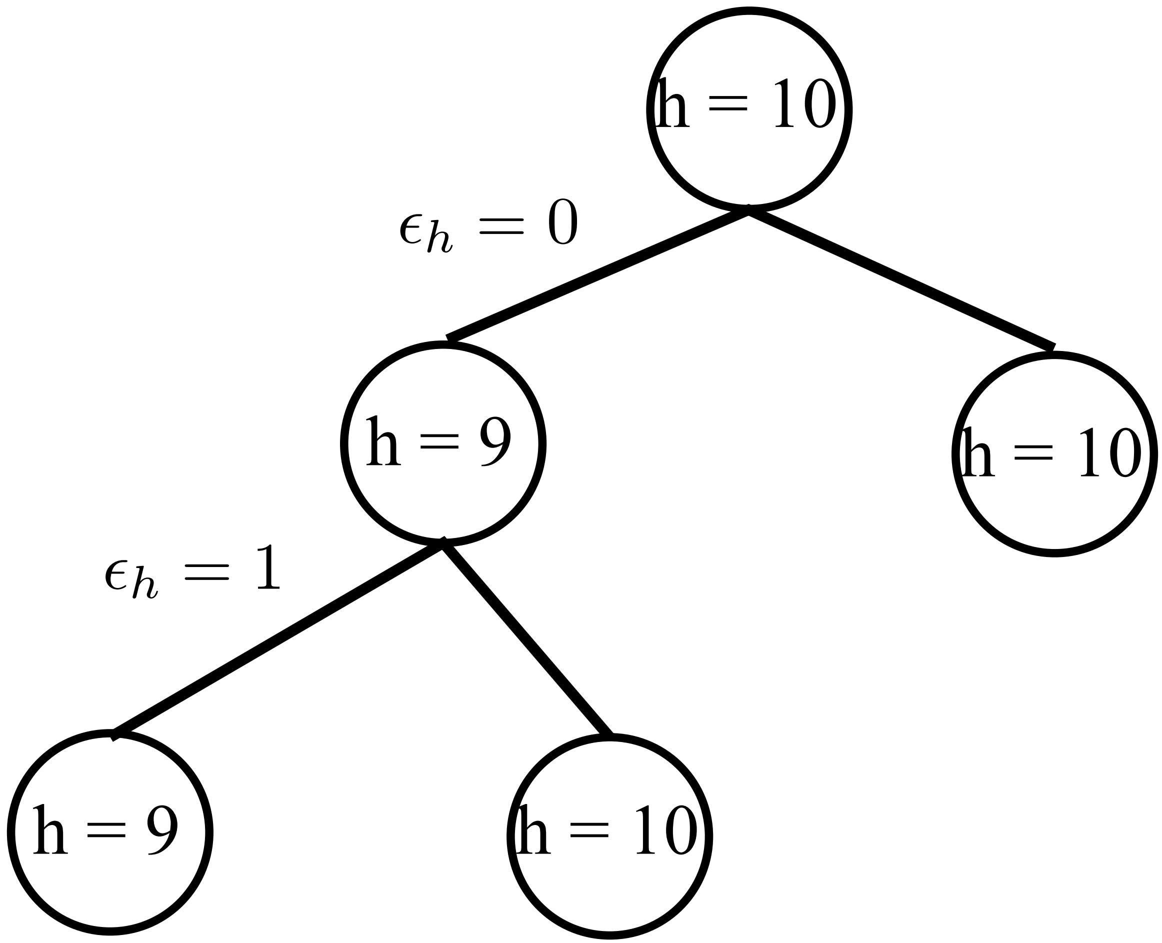
$$\widehat{h}(n) = h(n) + \bar{\epsilon_h} \cdot \widehat{d}(n)$$

how do we estimate $\bar{\epsilon_h}$ from ϵ_h? simple global average

Path Based Corrections

$\widehat{h}(n) = h(n) + \bar{\epsilon_h} \cdot \widehat{d}(n)$

how do we estimate $\bar{\epsilon_h}$ from ϵ_h? simple global average or ...

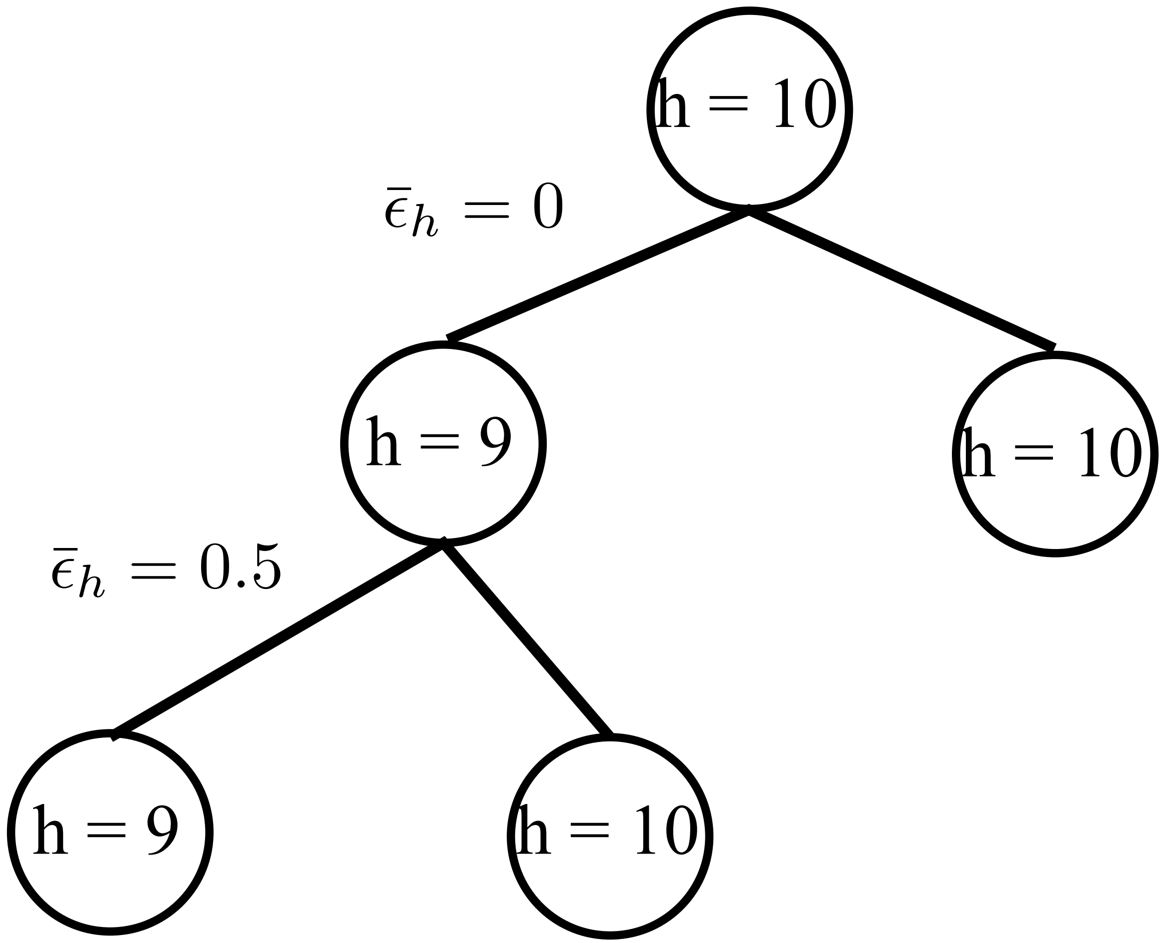


Path Based Corrections

$$\widehat{h}(n) = h(n) + \bar{\epsilon_h} \cdot \widehat{d}(n)$$

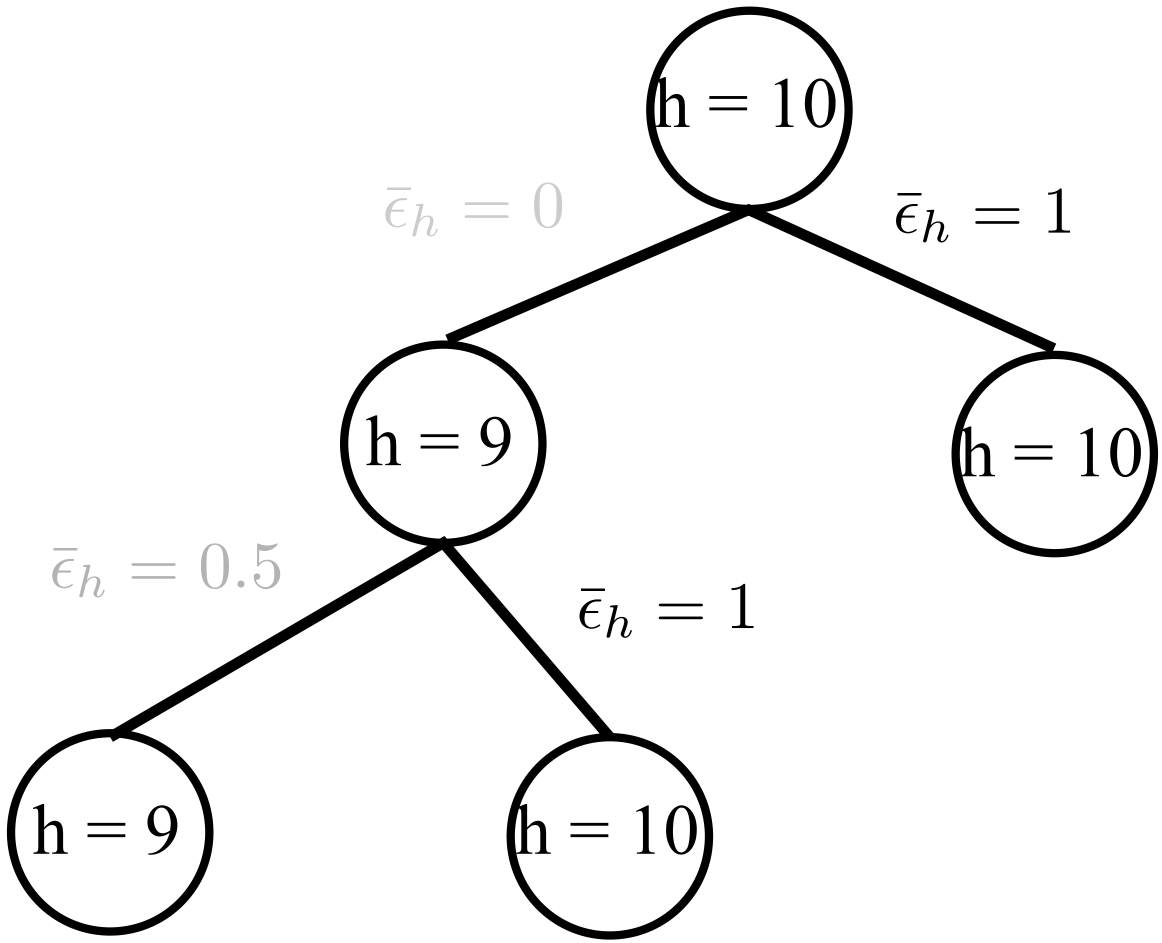
how do we estimate $\bar{\epsilon_h}$ from ϵ_h?

simple global average or ...



$$\widehat{h}(n) = h(n) + \bar{\epsilon_h} \cdot \widehat{d}(n)$$

how do we estimate $\bar{\epsilon_h}$ from ϵ_h? simple global average or ...



Summary

- a parent and its best child should have same f
- every expansion provides information – use it!
- single step error can be measured during search

  and we can use those corrections during that search

Performance

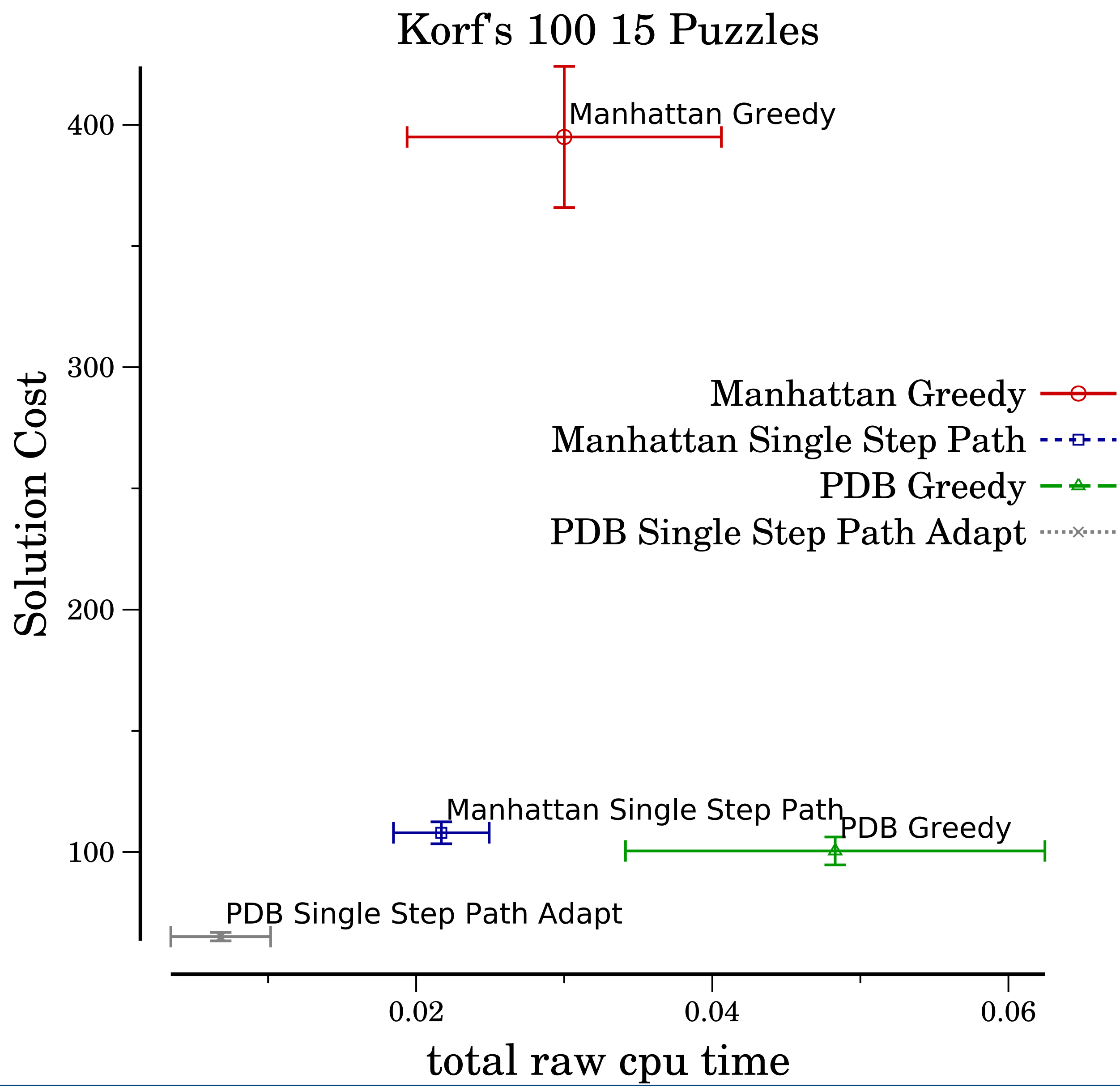
Outline

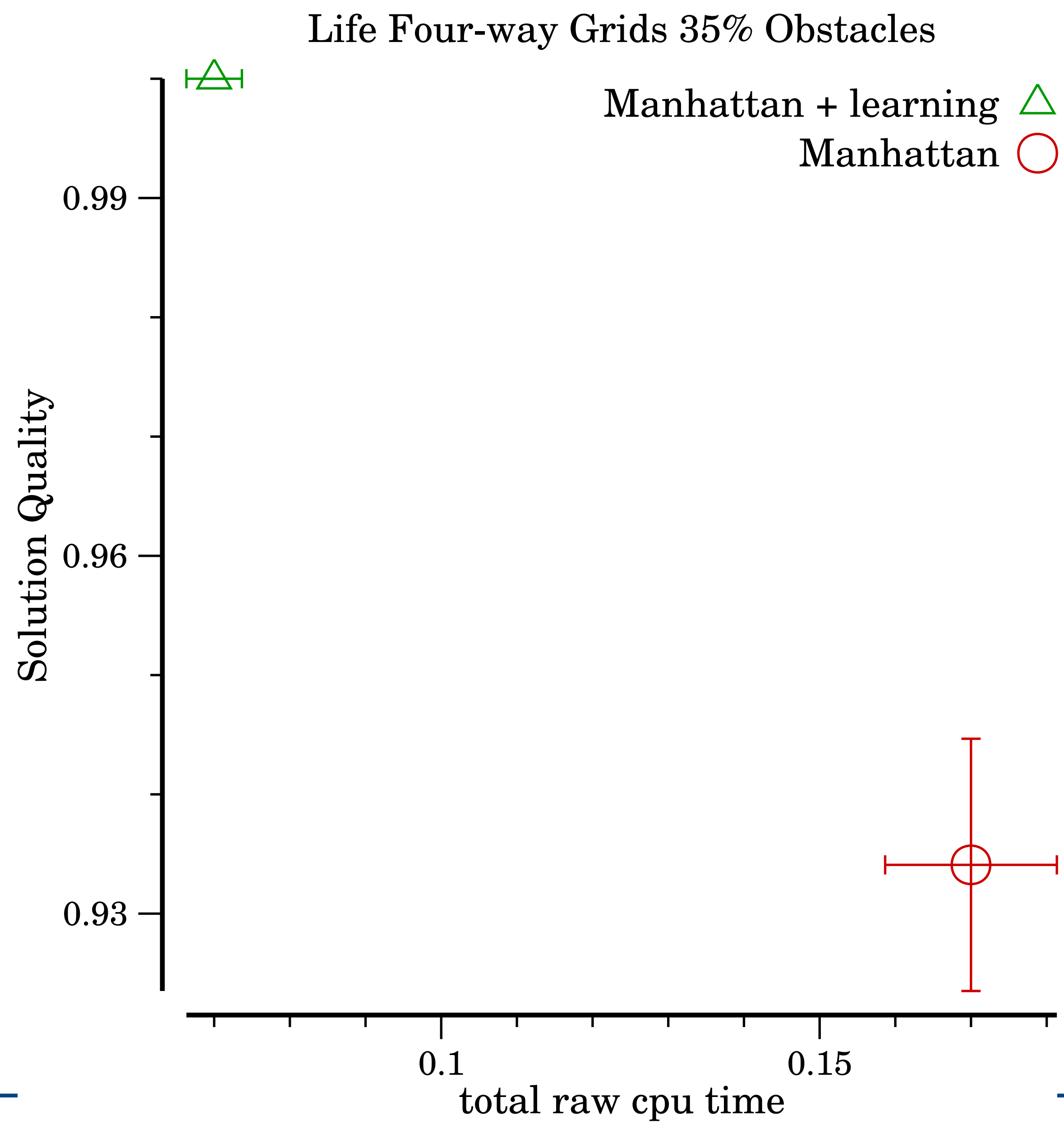
- motivation
- building inadmissible heuristics during search
- performance of learned heuristics

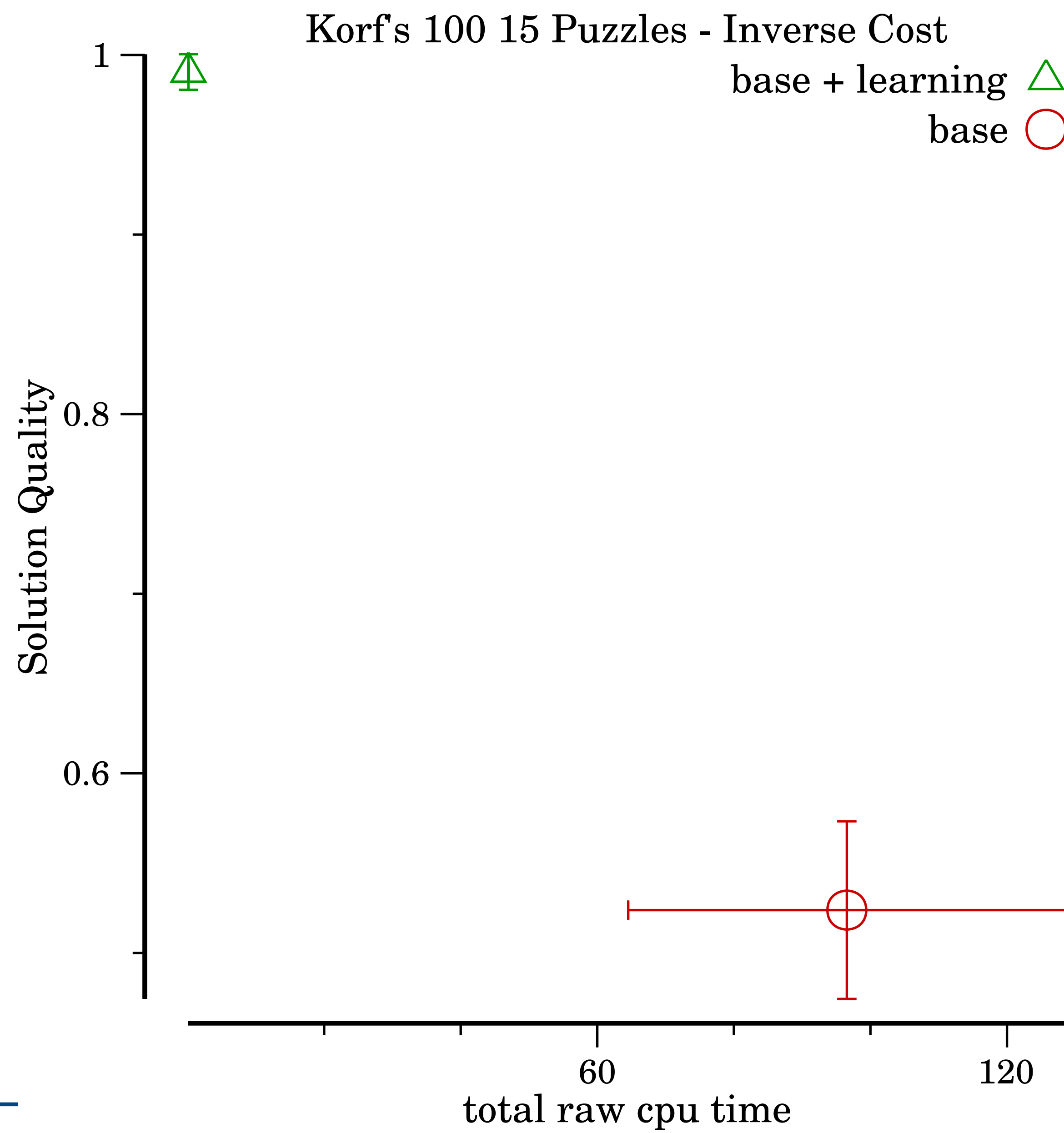
    suboptimal – greedy best-first search

    bounded suboptimal – skeptical search

Greedy Best First Search




Life Four-way Grids 35% Obstacles
Manhattan + learning
Manhattan
0.99
0.96
0.93
Solution Quality
0.1
0.15
total raw cpu time


Korf's 100 15 Puzzles - Inverse Cost
base + learning
base
Solution Quality
1
0.8
0.6
60
120
total raw cpu time

Outline

- motivation
- building inadmissible heuristics during search
- performance of learned heuristics

    suboptimal – greedy best-first search

    bounded suboptimal – skeptical search

Bounded Suboptimal Search: Skeptical Search

given a suboptimality bound w,
find a solution within the bound as quickly as possible

Bounded Suboptimal Search: Skeptical Search

given a suboptimality bound w,
find a solution within the bound as quickly as possible

use optimistic framework (Thayer and Ruml, ICAPS-08):

1. run weighted A^* with an inadmissible heuristic

$$f'(n) = g(n) + w \cdot \widehat{h}(n)$$

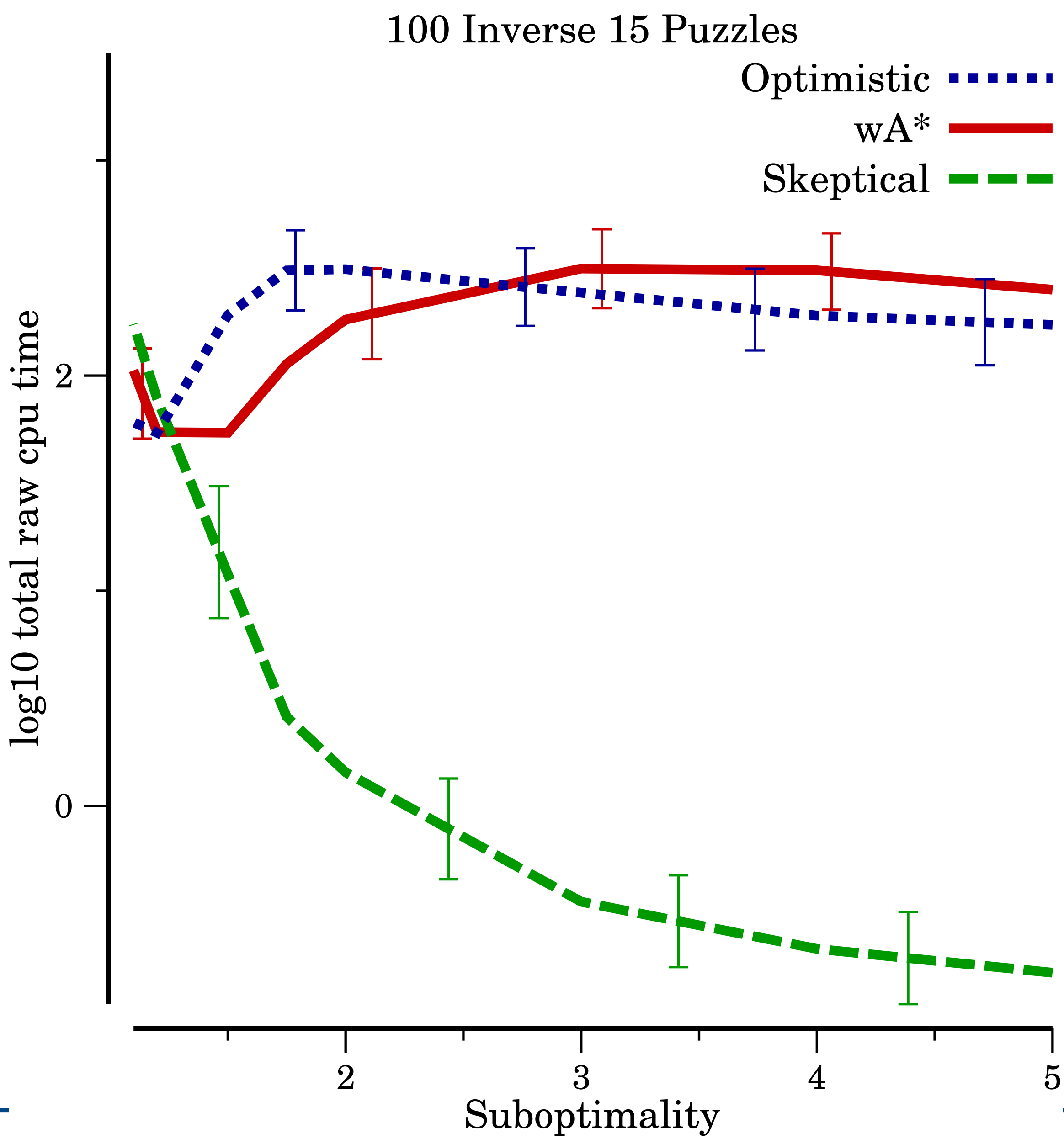
2. after a solution is found expand node with lowest f value

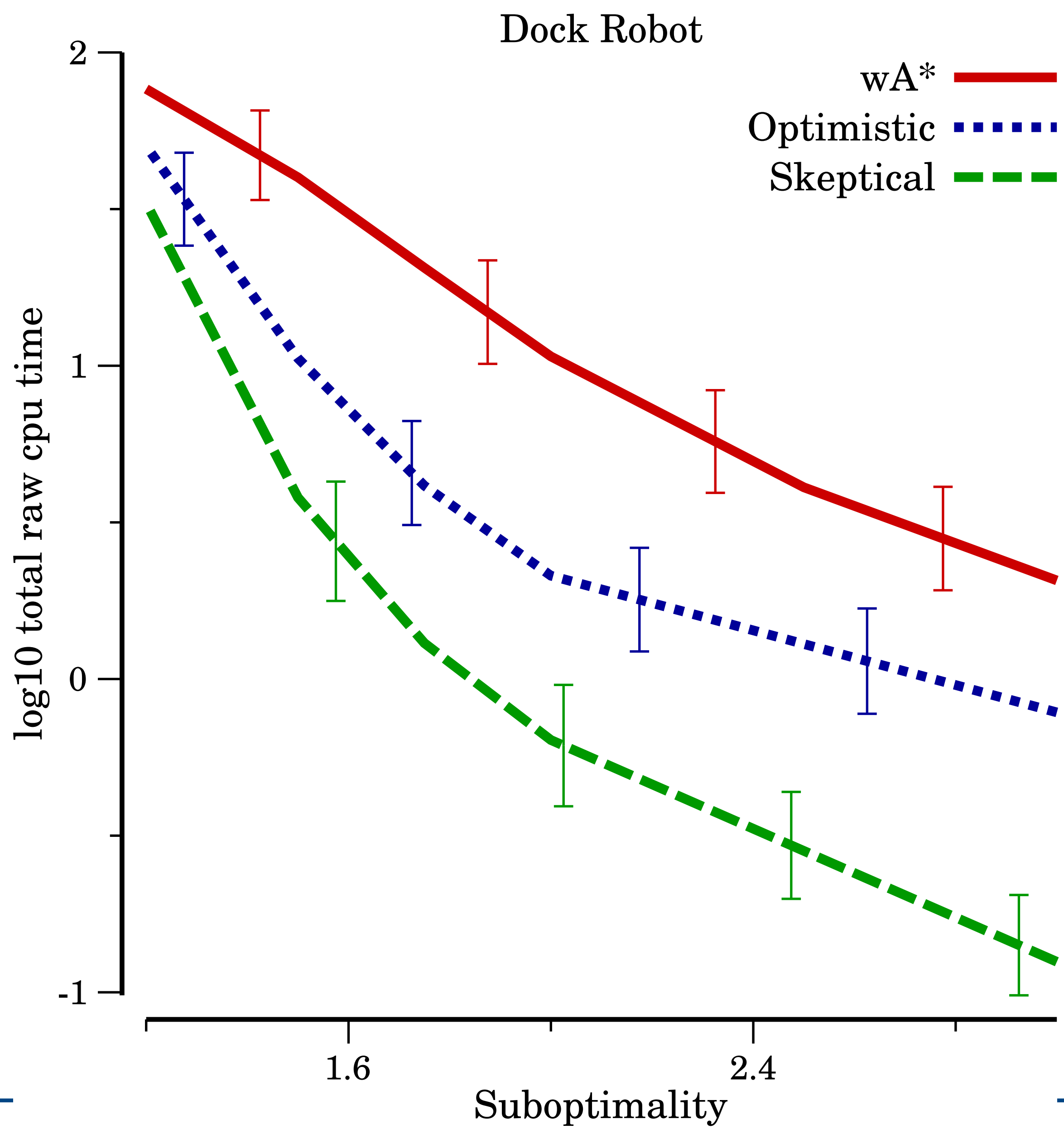
continue until $w \cdot f(best_f) \geq f(sol)$

this 'clean up' guarantees solution quality

(no ad hoc optimism parameter!)

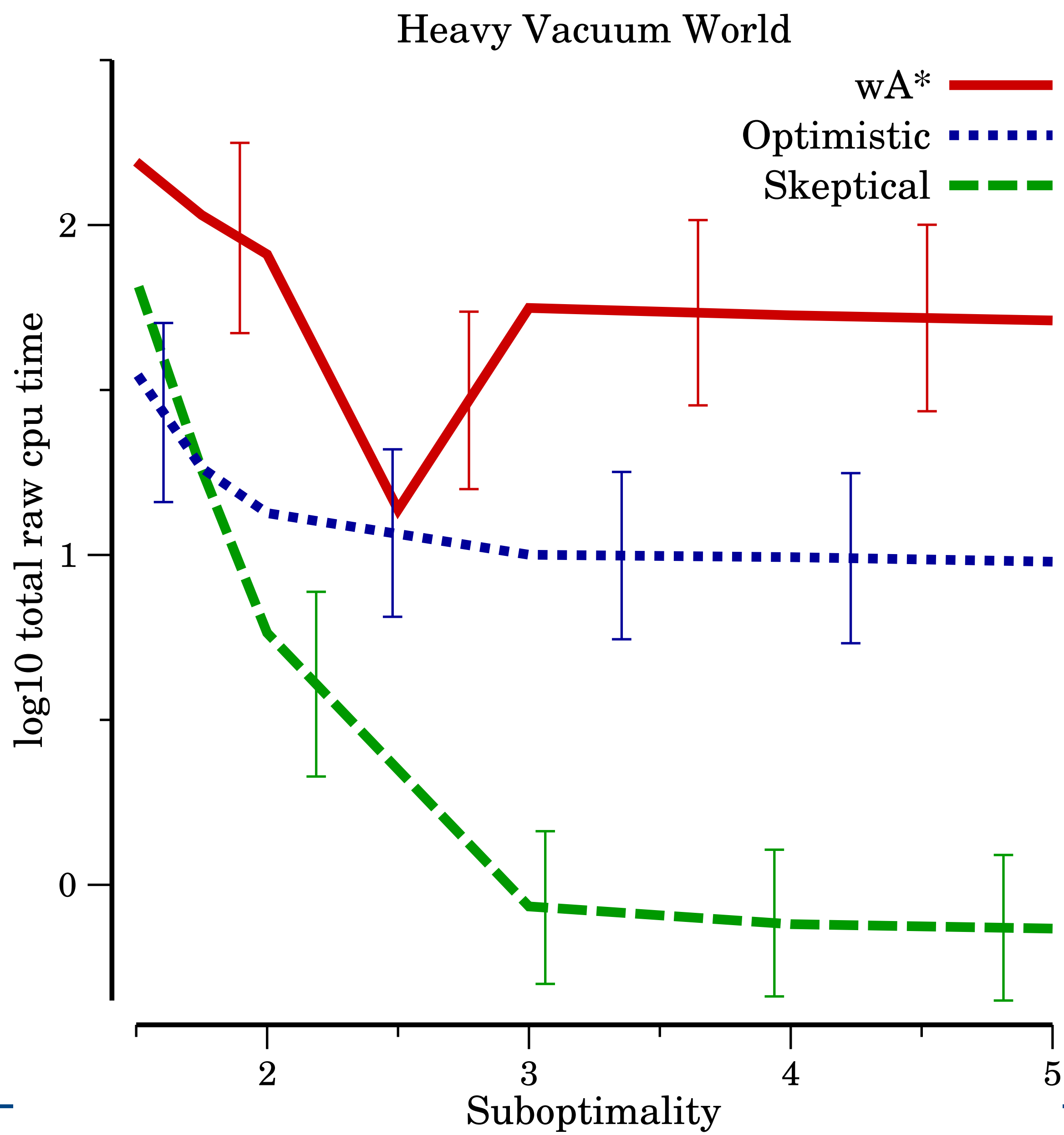
Performance In Bounded Suboptimal Search





Dock Robot
wA*
Optimistic
Skeptical
log10 total raw cpu time
2
1
0
-1
1.6
2.4
Suboptimality

Performance In Bounded Suboptimal Search



In the Paper

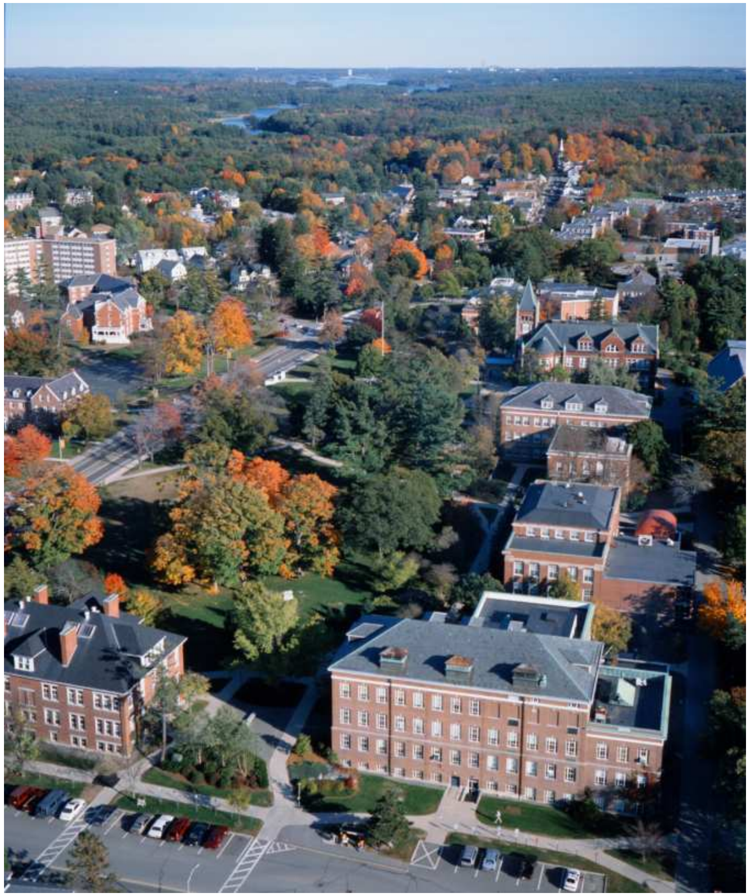
- accuracy less important than relative ordering
- instance specific learning truly beneficial
- distance estimates very helpful
  for non-unit cost problems
- skeptical proof of bounded suboptimality

Summary

- we can learn inadmissible heuristics
  - these improve search guidance, make search go fast
- we can learn them online, during search
  - no dependence on domain specific information
  - no offline training
  - can learn instance specific correction
- skeptical search
  - removes parameter of optimistic search
  - state of the art performance

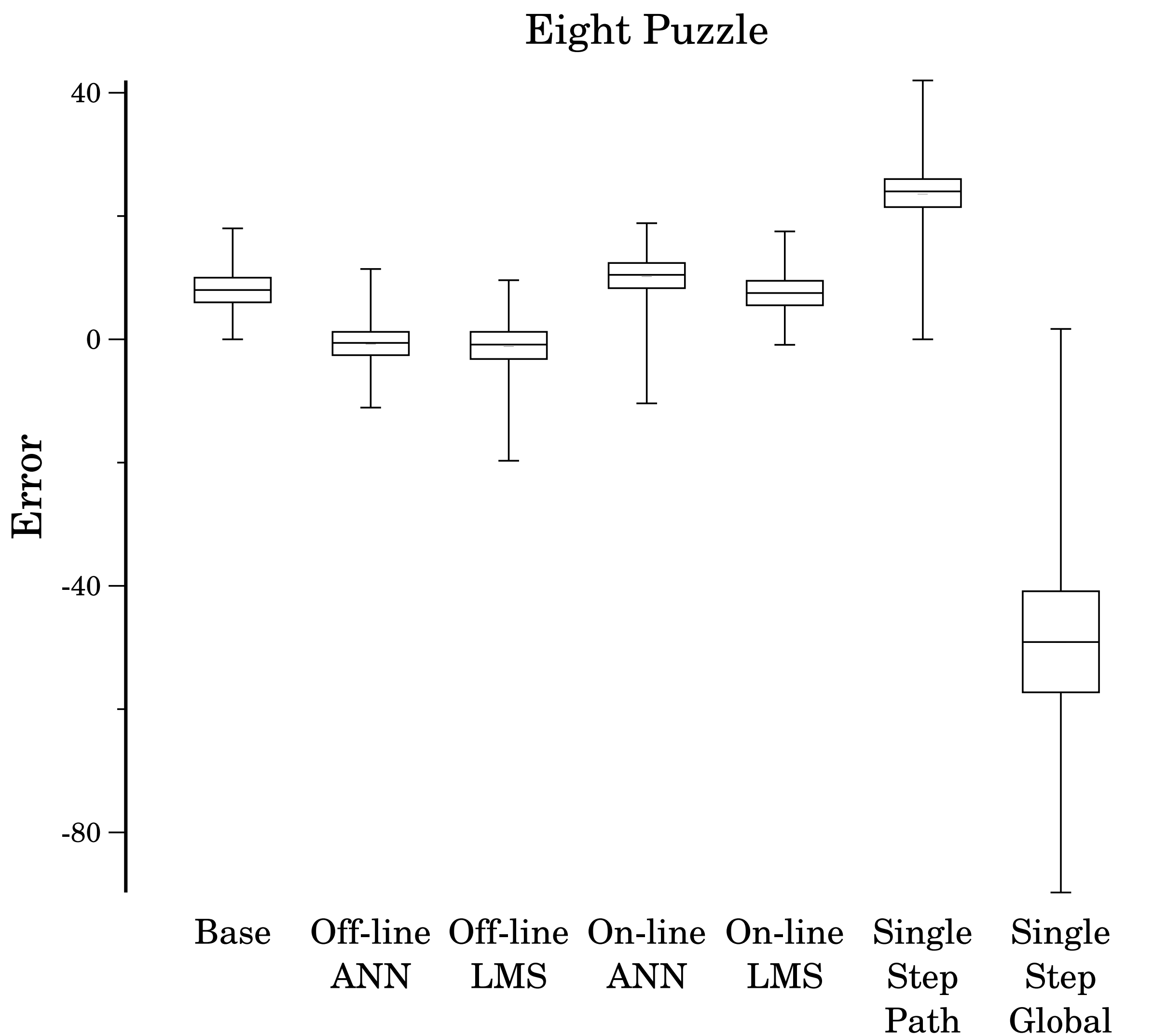
The University of New Hampshire

Tell your students to apply to grad school in CS at UNH!

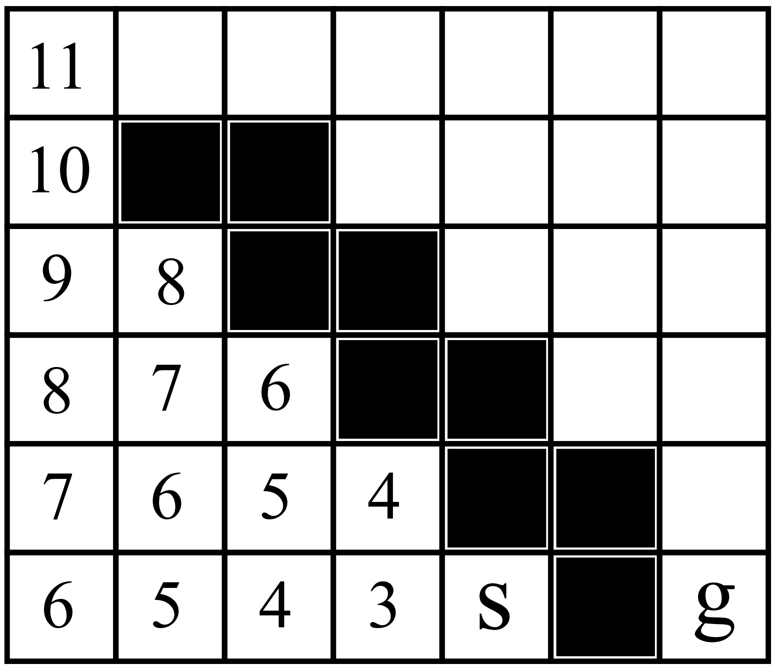


- friendly faculty
- funding
- individual attention
- beautiful campus
- low cost of living
- easy access to Boston, White Mountains
- strong in AI, infoviz, networking, systems, bioinformatics

Heuristic Accuracy



It Doesn't Always Work



d Is Important!