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
greedy best-first search
greedy best-first search with learning
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**
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
Observing Error Between Parent and Best Child

Learning
- Observing Error
- Path
- Summary

Performance

Conclusions

Backup Slides

Jordan Thayer (UNH)
Observing Error Between Parent and Best Child

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

\[ f^*(p) = f^*(bc) \]
\[ g(p) + h^*(p) = g(bc) + h^*(bc) \]
\[ h^*(p) = h^*(bc) + c(p, bc) \]
Observing Error Between Parent and Best Child

Introduction
Learning
- Observing Error
- Path
- Summary
Performance
Conclusions
Backup Slides

\[ f(p) \text{ should equal } f(bc) \]

\[ f^*(p) = f^*(bc) \]
\[ g(p) + h^*(p) = g(bc) + h^*(bc) \]
\[ h^*(p) = h^*(bc) + c(p, bc) \]

\[ h(p) = h(bc) + c(p, bc) - \epsilon_h \]
\[ \epsilon_h = h(bc) + c(p, bc) - h(p) \]
Observing Error Between Parent and Best Child

Introduction

Learning
- Observing Error
- Path
- Summary

Performance

Conclusions

Backup Slides

---

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

\[
\begin{align*}
  f^*(p) &= f^*(bc) \\
  g(p) + h^*(p) &= g(bc) + h^*(bc) \\
  h^*(p) &= h^*(bc) + c(p, bc) \\
  h(p) &= h(bc) + c(p, bc) - \epsilon_h \\
  \epsilon_h &= h(bc) + c(p, bc) - h(p)
\end{align*}
\]

\[
\begin{align*}
  \hat{h}(n) &= h(n) + \bar{\epsilon}_h \cdot d(n) \\
  \hat{h}(n) &= h(n) + \bar{\epsilon}_h \cdot \hat{d}(n)
\end{align*}
\]
\[ \hat{h}(n) = h(n) + \bar{\epsilon}_h \cdot \hat{d}(n) \]

how do we estimate \( \bar{\epsilon}_h \) from \( \epsilon_h \)? simple global average
\[ \hat{h}(n) = h(n) + \bar{\epsilon}_h \cdot \tilde{d}(n) \]

how do we estimate \( \bar{\epsilon}_h \) from \( \epsilon_h \)?

simple global average or ...
\[ \hat{h}(n) = h(n) + \bar{\epsilon}_h \cdot \hat{d}(n) \]

how do we estimate \( \bar{\epsilon}_h \) from \( \epsilon_h \)?

simple global average or ...

\[ \bar{\epsilon}_h = 0 \]

\[ \bar{\epsilon}_h = 0.5 \]
\[ \hat{h}(n) = h(n) + \bar{\epsilon}_h \cdot \hat{d}(n) \]

how do we estimate \( \bar{\epsilon}_h \) from \( \epsilon_h \)?

simple global average or ...

\[ \bar{\epsilon}_h = \frac{1}{4} \sum_{n=1}^{4} \epsilon_h(n) \]

\[ \bar{\epsilon}_h = \frac{2}{4} \sum_{n=1}^{4} \epsilon_h(n) \]

\[ \bar{\epsilon}_h = \frac{3}{4} \sum_{n=1}^{4} \epsilon_h(n) \]
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
■ motivation

■ building inadmissible heuristics during search

■ performance of learned heuristics

  suboptimal – greedy best-first search

  bounded suboptimal – skeptical search
Greedy Best First Search

Korf’s 100 15 Puzzles

- Manhattan Greedy
- Manhattan Single Step Path
- PDB Greedy
- PDB Single Step Path Adapt

Solution Cost vs. total raw cpu time
Greedy Best First Search

Life Four-way Grids 35% Obstacles

Solution Quality vs. total raw cpu time

- Manhattan + learning (△)
- Manhattan (○)

Jordan Thayer (UNH)
Greedy Best First Search

Korf's 100 15 Puzzles - Inverse Cost

Solution Quality vs. total raw cpu time

- Green triangle: base + learning
- Red circle: base
motivation

building inadmissible heuristics during search

performance of learned heuristics

- suboptimal – greedy best-first search
- bounded suboptimal – skeptical search
given a suboptimality bound $w$, find a solution within the bound as quickly as possible
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
   \[ f'(n) = g(n) + w \cdot \hat{h}(n) \]

2. after a solution is found expand node with lowest $f$ value
   continue until $w \cdot f(\text{best}_f) \geq f(\text{sol})$
   this 'clean up' guarantees solution quality

(no ad hoc optimism parameter!)
Performance In Bounded Suboptimal Search

100 Inverse 15 Puzzles

Optimistic

wA*

Skeptical

Suboptimality

log10 total raw cpu time

Jordan Thayer (UNH)
Dock Robot

Comparison of wA*, Optimistic, and Skeptical

Suboptimality

log10 total raw cpu time
Heavy Vacuum World

Suboptimality

log10 total raw cpu time

wA*

Optimistic

Skeptical

Introduction
Learning
Performance
Greedy Search
Bounded Quality
Conclusions
Backup Slides
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
- 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
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
Eight Puzzle

Error

-80 -40 0 40

ANN LMS ANN LMS Step Step
     Path     Global

Jordan Thayer (UNH)
It Doesn’t Always Work
Life Four-way Grids 35% Obstacles

Solution Cost vs. total raw cpu time

- Single Step Path
- Just h Path
- Just h Global
- h Path

- 3e+06
- 2.9e+06
- 2.8e+06
- 0.1
- 0.15
- 0.2

Jordan Thayer (UNH)