Fast and Loose in Bounded Suboptimal Heuristic Search

Jordan Thayer and Wheeler Ruml

University of New Hampshire

{jtd7, ruml} at cs.unh.edu

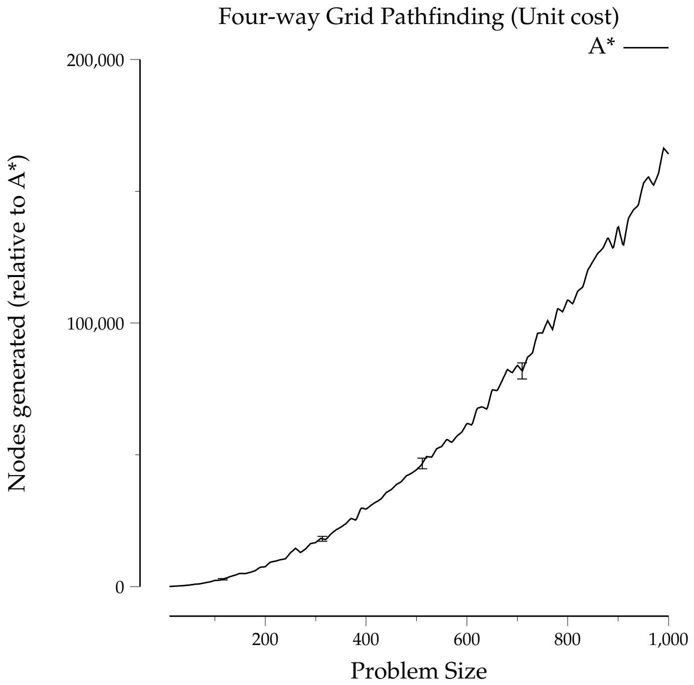
Ephrat Bitton

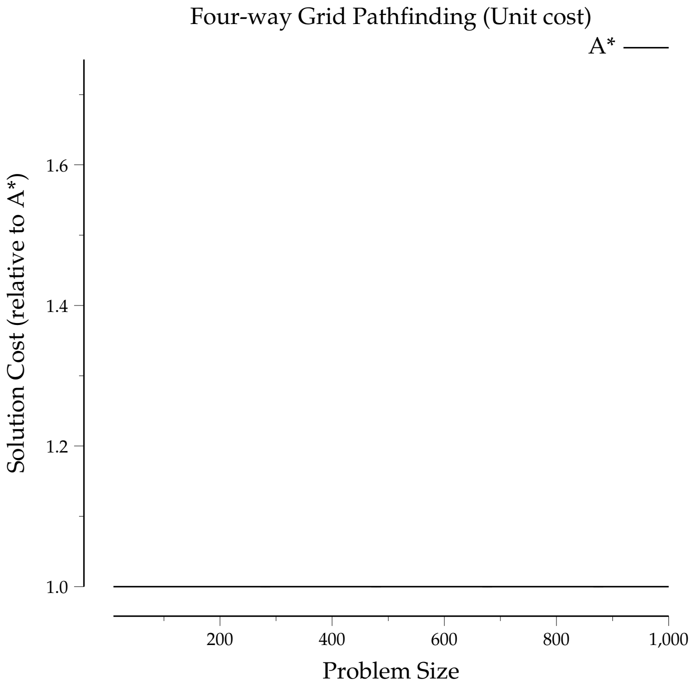
University of California Berkeley

ebitton at berkeley.edu

Motivation

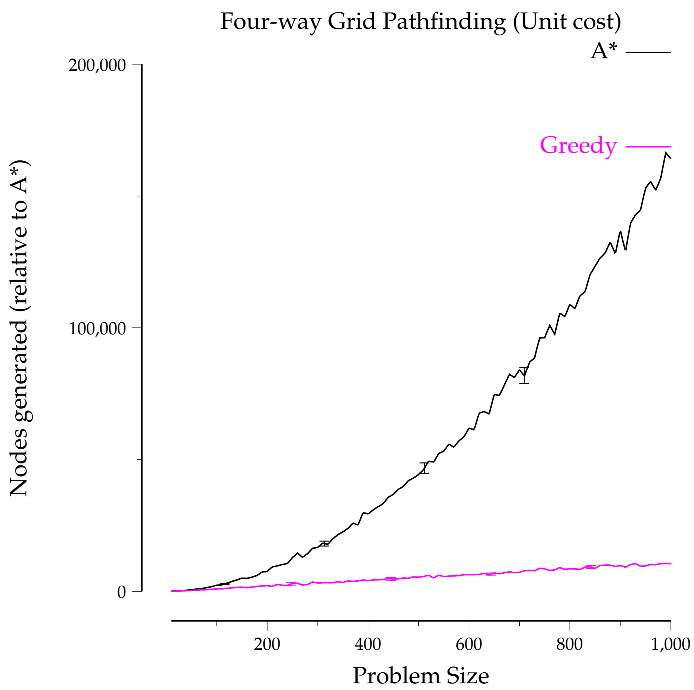
- Finding optimal solutions is prohibitively expensive.

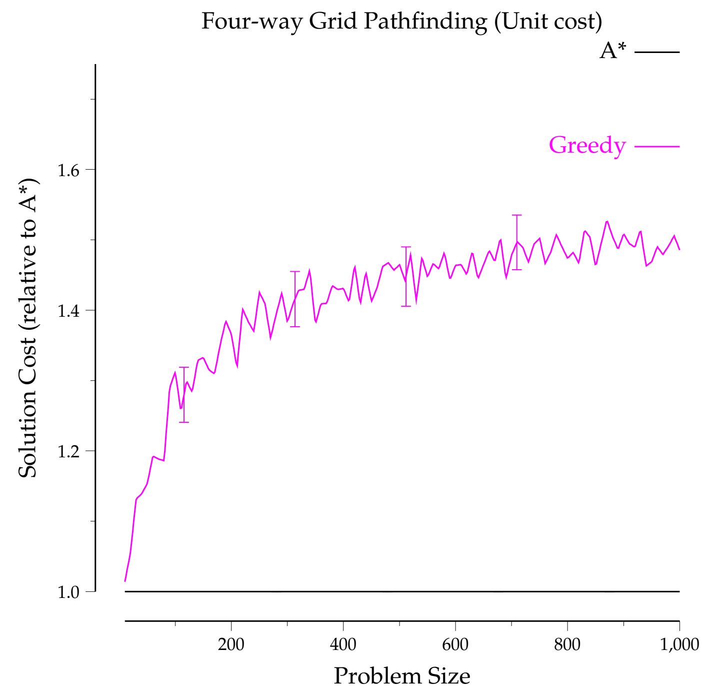




Motivation

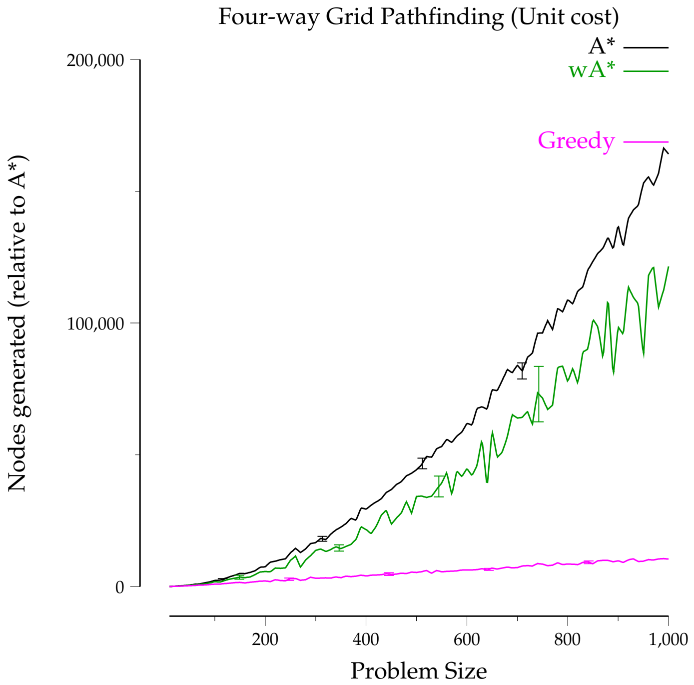
- Finding optimal solutions is prohibitively expensive.
- Its nice to limit suboptimality.

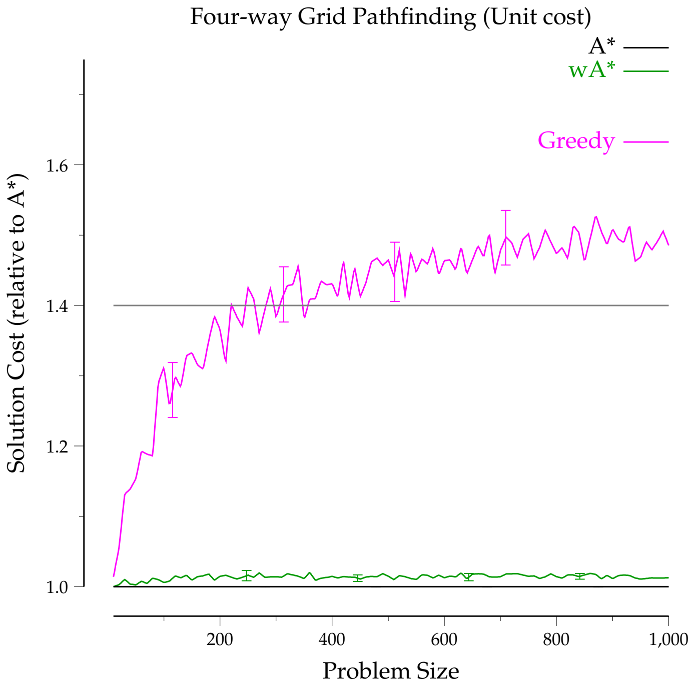




Motivation

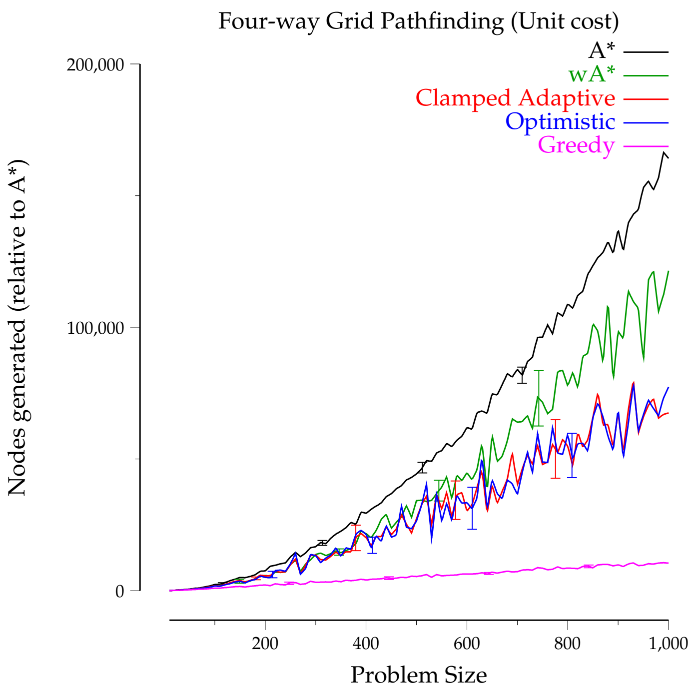
- Finding optimal solutions is prohibitively expensive.
- Its nice to limit suboptimality.
- Weighted A* is a popular method for doing that.

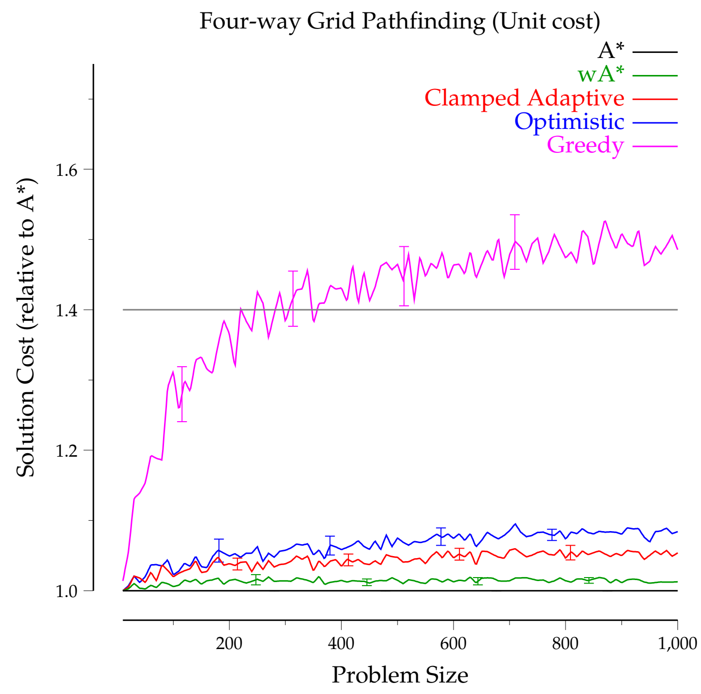




Motivation

- Finding optimal solutions is prohibitively expensive.
- Its nice to limit suboptimality.
- Weighted A* is a popular method for doing that.
- This talk: two algorithms which are often better.



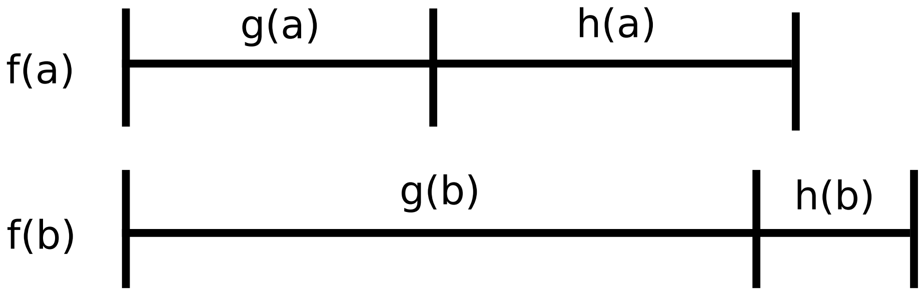


Talk Outline

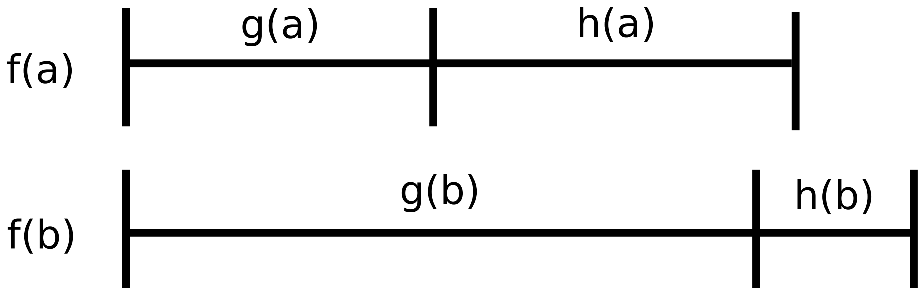
- Background
  - Weighted A*
- Strict Approach: Clamped Adaptive
  - Correct for underestimating $h(n)$
  - Bound correction to ensure w-admissibility
- Loose Approach: Optimistic Search
  - Greedily search for a solution
  - Enforce suboptimality bound afterwards

Weighted A^* (Pohl, 1970)

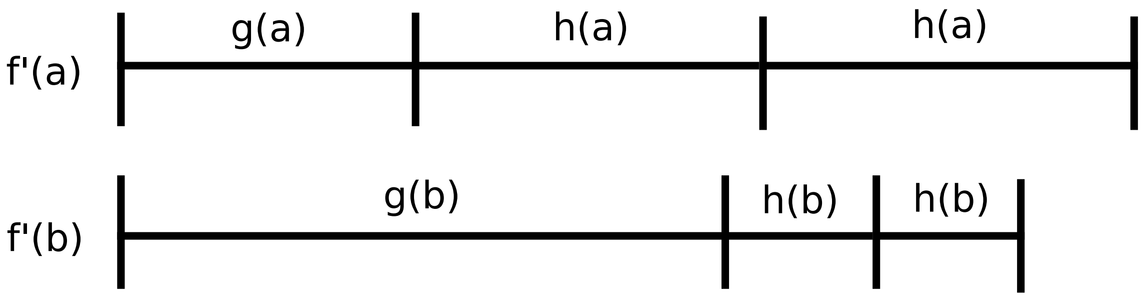
A^* is a best first search ordered on $f(n) = g(n) + h(n)$



A^* is a best first search ordered on $f(n) = g(n) + h(n)$

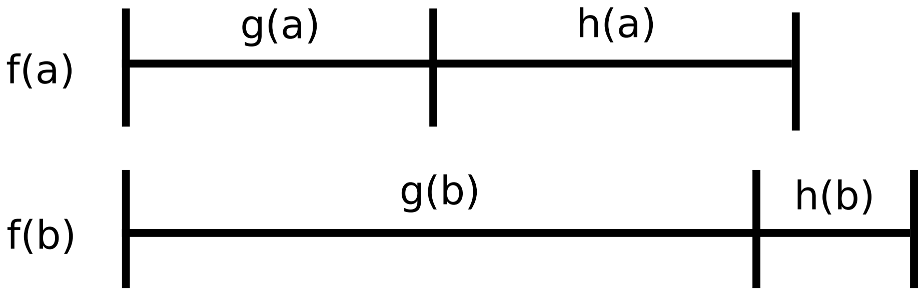


Weighted A^*: $f'(n) = g(n) + w \cdot h(n)$

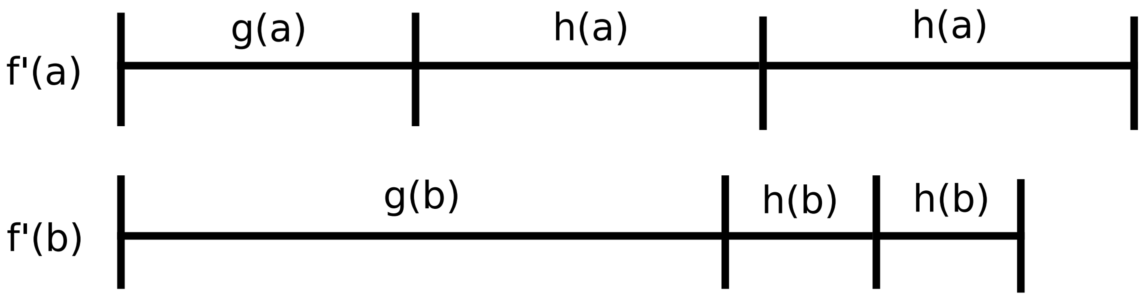


Weighted A^* (Pohl, 1970)

A^* is a best first search ordered on $f(n) = g(n) + h(n)$



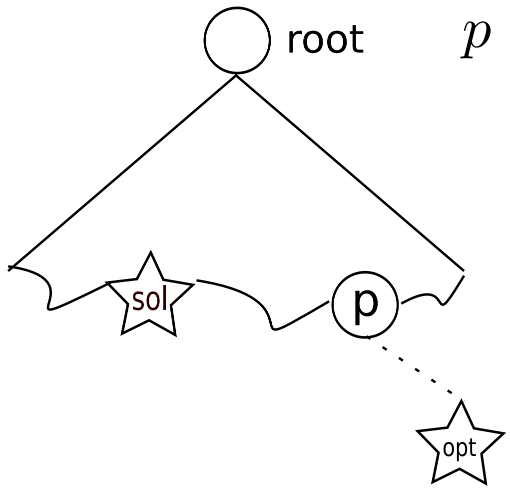
Weighted A^*: $f'(n) = g(n) + w \cdot h(n)$



What does w do?

- breaks ties on $f(n)$ in favor of high $g(n)$
- corrects for underestimating $h(n)$
- deepens search / emphasises greed

Weighted A^* Respects a Bound



p is a node in open on an optimal path to opt

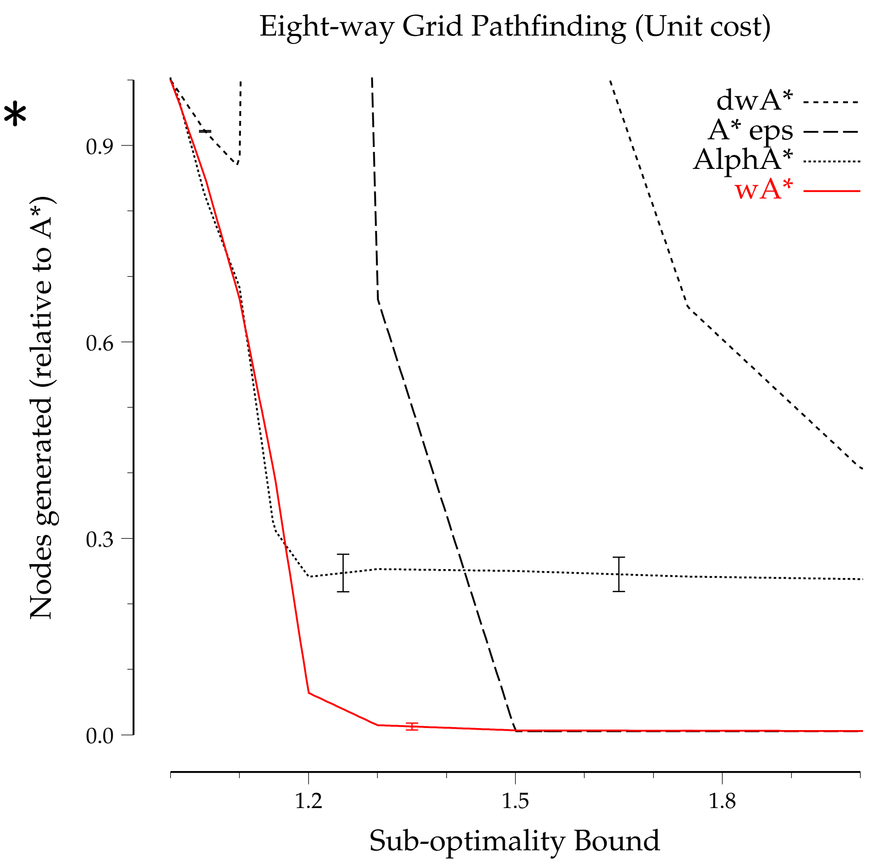
$$f(n) = g(n) + h(n)$$
$$f'(n) = g(n) + w \cdot h(n)$$

$$\begin{aligned}
& g(sol) \\
& f'(sol) \leq f'(p) \\
& \quad g(p) + w \cdot h(p) \leq w \cdot (g(p) + h(p)) \\
& \quad\quad w \cdot f(p) \leq w \cdot f(opt) \\
& \quad\quad\quad w \cdot g(opt)
\end{aligned}$$

Therefore, $g(sol) \leq w \cdot g(opt)$

Weighted A^* is a Popular Choice

- Weighted A*
  Pohl (1970)
- Dynamically Weighted A*
  Pohl (1973)
- A_ϵ
  Ghallab & Allard (1983)
- A^*_ϵ
  Pearl (1984)
- AlphA*
  Reese & Frichs (unpublished)



Talk Outline

- Background
  - Weighted A*
- Strict Approach: Clamped Adaptive
  - Correct for underestimating $h(n)$
  - Bound correction to ensure w-admissibility
- Loose Approach: Optimistic Search
  - Greedily search for a solution
  - Enforce suboptimality bound afterwards

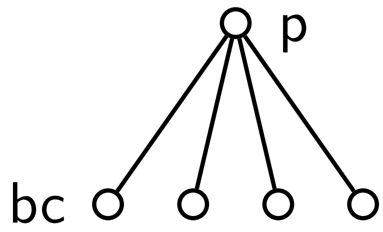
Improving Weighted A^*

- If h were perfect, solutions would be found in linear time.
- How do we improve $h(n)$?

  By correcting for the error in $h(n)$
- We'll ensure w-admissibility shortly.

Correcting $h(n)$ with one step error

Consider the single expansion:

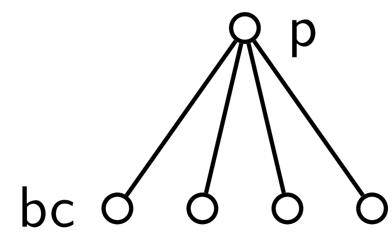


Recall that $f(n) = g(n) + h(n)$

- $f(n)$ should remain constant across parent and child.
  - if $f(n) = g(n) + h^*(n)$ this would be true.
  - $g(n)$ is exact.
  - All the error in $f(n)$ comes from $h(n)$.
- $err_h = f(bc) - f(p)$

Track a running average of err_h.
$\hat{f}(n) = g(n) + \hat{h}(n)$
$\hat{h}(n) = h(n) \cdot (1 + err_h)$

Consider the single expansion:



Recall that $f(n) = g(n) + h(n)$

- $f(n)$ should remain constant across parent and child.
  - if $f(n) = g(n) + h^*(n)$ this would be true.
  - $g(n)$ is exact.
  - All the error in $f(n)$ comes from $h(n)$.
- $err_h = f(bc) - f(p)$

Track a running average of err_h.

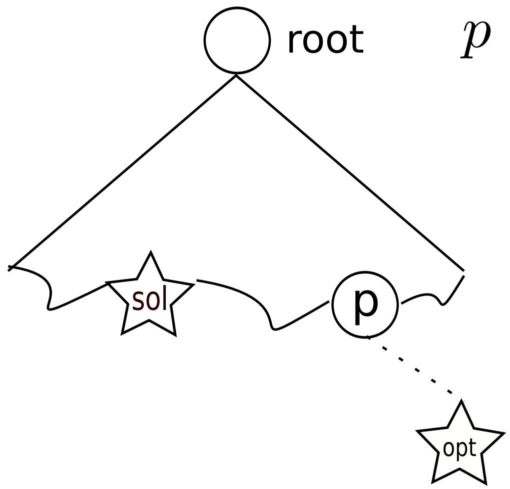
$\hat{f}(n) = g(n) + \hat{h}(n)$

$\hat{h}(n) = h(n) \cdot (1 + err_h)$

$\hat{h}(n)$ is inadmissible.

Clamping enforces w-admissibility.

Admissibility of Clamping: Weighted A*



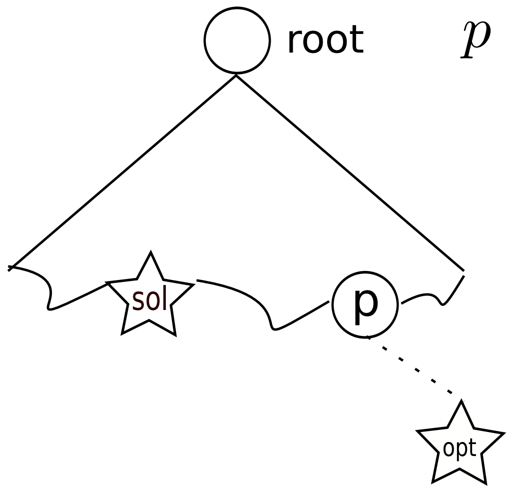
p is a node in open on an optimal path to opt

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

$$\begin{aligned}
&g(sol)\\
&f'(sol) \leq f'(p)\\
&g(p) + w \cdot h(p) \leq w \cdot (g(p) + h(p))\\
&w \cdot f(p) \leq w \cdot f(opt)\\
&w \cdot g(opt)
\end{aligned}$$

Therefore, $g(sol) \leq w \cdot g(opt)$

Admissibility of Clamping: Clamped Adaptive



p is a node in open on an optimal path to opt

$$f(n) = g(n) + h(n)$$
$$\widetilde{f}(n) = \min(\widehat{f}(n), w \cdot f(n))$$

$$\begin{aligned}
g(sol) = \widetilde{f}(sol) \\
\widetilde{f}(sol) &\leq \widetilde{f}(p) \\
\widetilde{f}(p) &\leq w \cdot f(p) \\
w \cdot f(p) &\leq w \cdot f(opt)
\end{aligned}$$

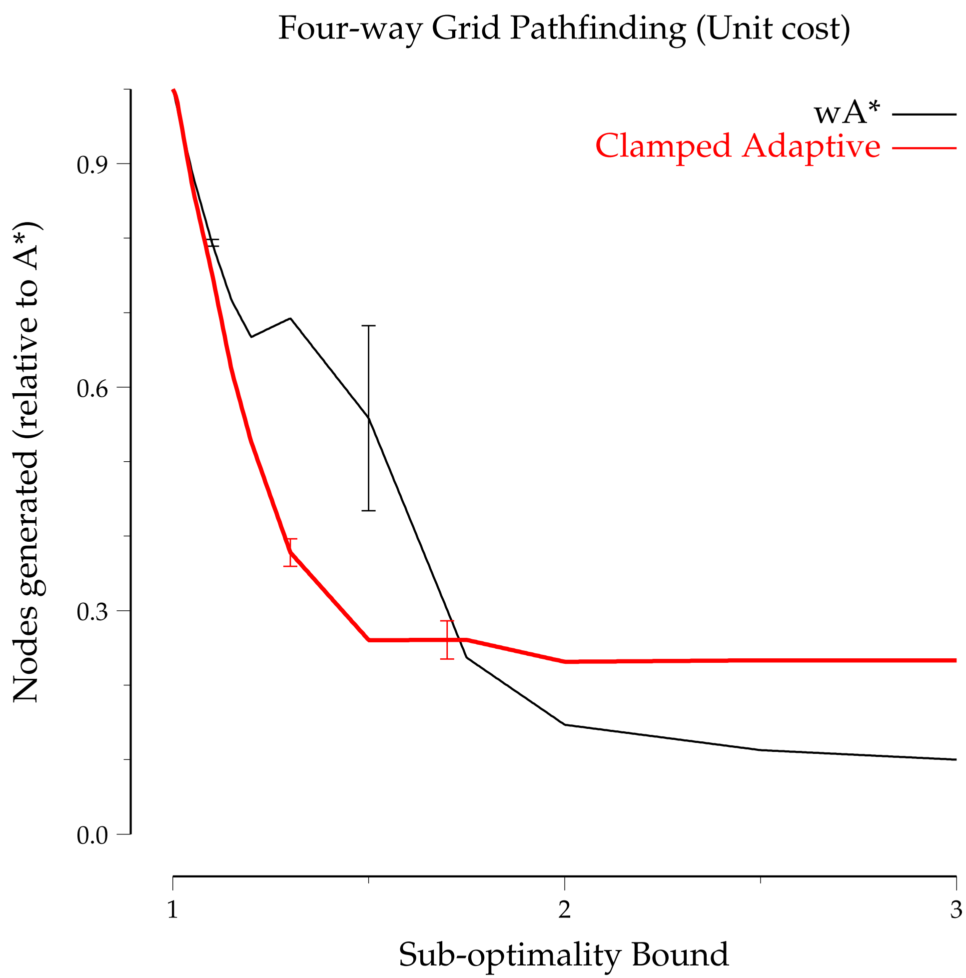
And $g(s) \leq w \cdot g(opt)$ is still true.

Empirical Evaluation

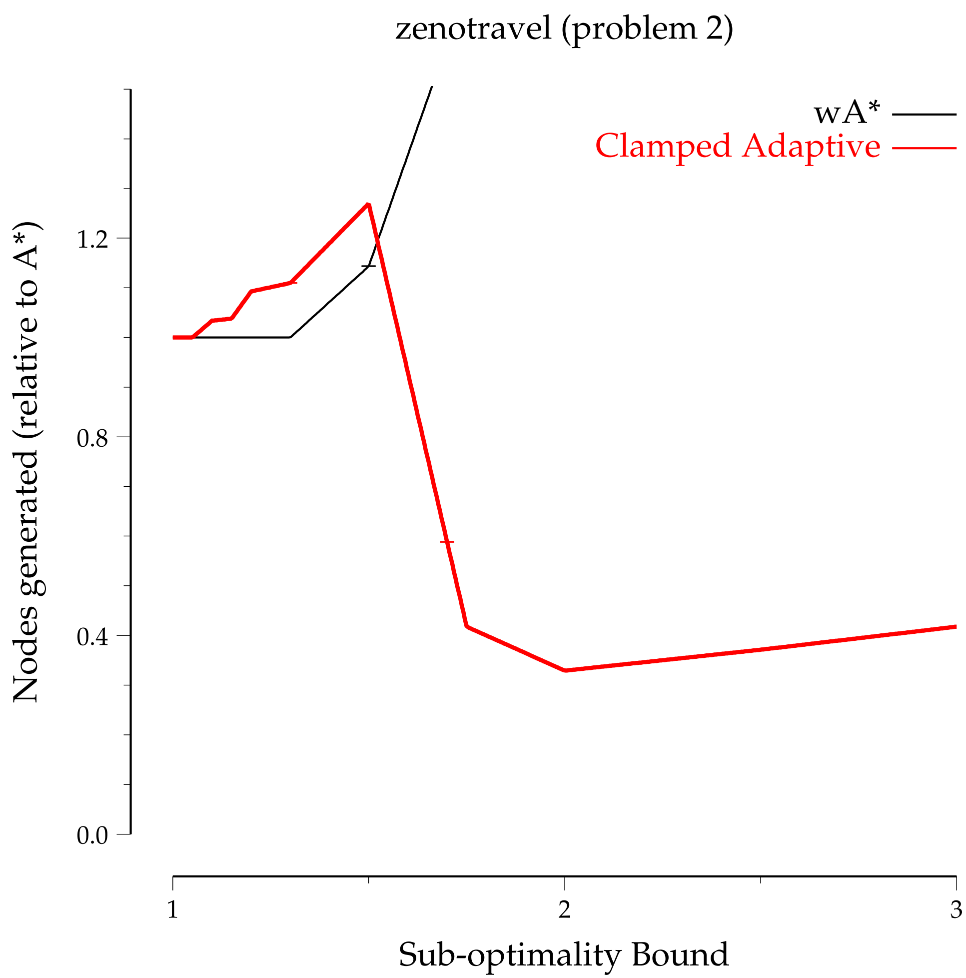
- Grid world path finding
  - Four-way and Eight-way Movement
  - Unit and Life Cost Models
  - 25%, 30%, 35%, 40%, 45% obstacles
- Temporal Planning
  - Blocksworld, Logistics, Rover, Satellite, Zenotravel

See the paper for details.

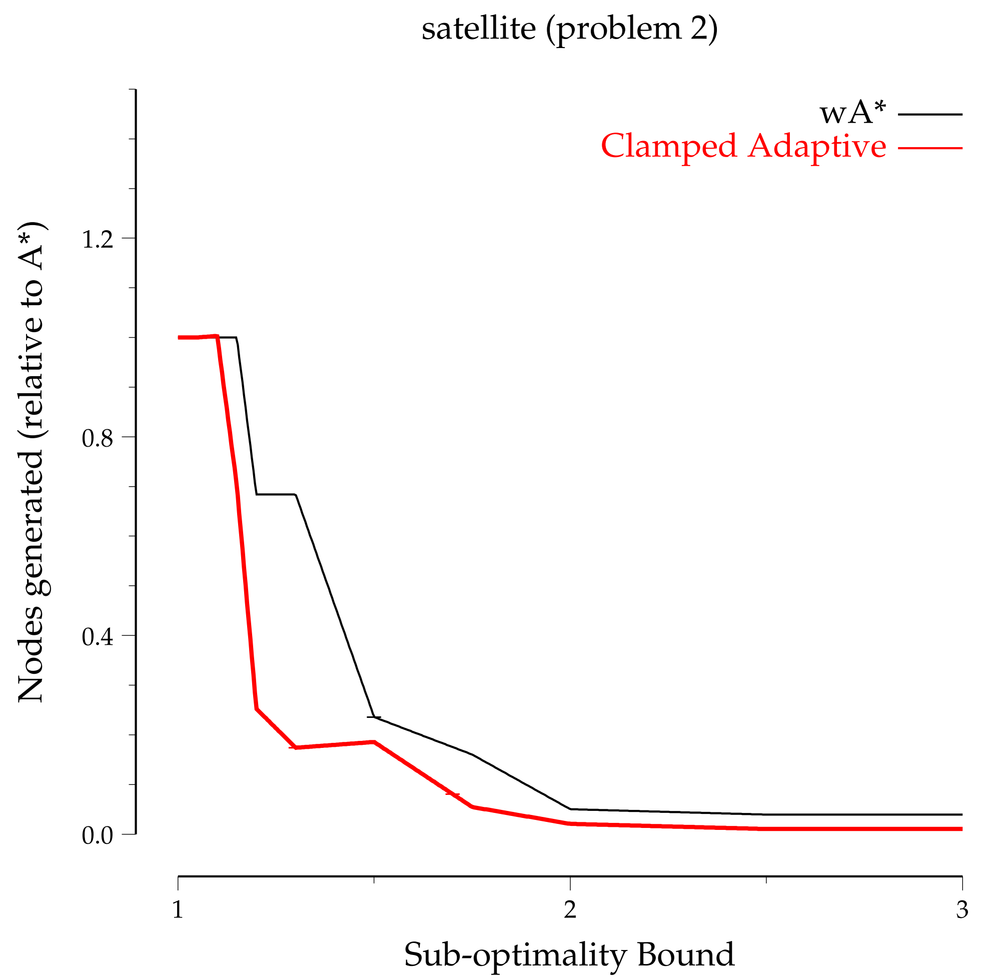
Performance of Clamped Adaptive



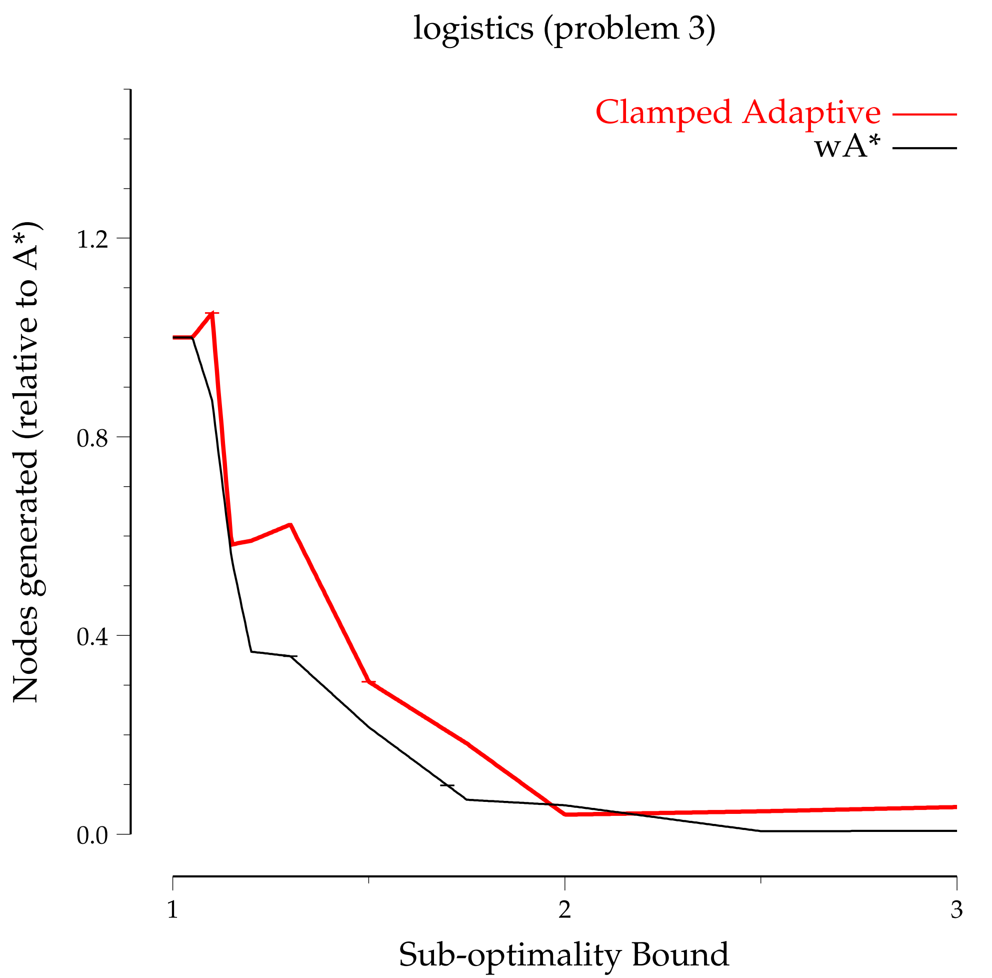
Performance of Clamped Adaptive



Performance of Clamped Adaptive



Performance of Clamped Adaptive



Clamped Adaptive: Summary

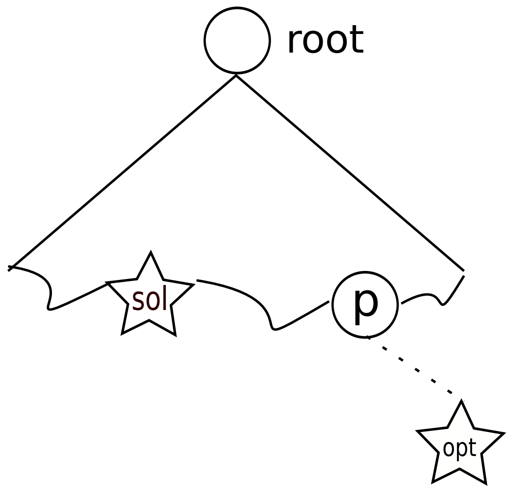
Clamped Adaptive:

- On-line heuristic correction seems promising
  - Performance varies
    - Does well for small bounds
    - Fails to become greedy
- No parameter tuning needed
- Clamping for admissibility of inadmissible heuristics

Talk Outline

- Background
  - Weighted A*
- Strict Approach: Clamped Adaptive
  - Correct for underestimating $h(n)$
  - Bound correction to ensure w-admissibility
- Loose Approach: Optimistic Search
  - Greedily search for a solution
  - Enforce suboptimality bound afterwards

Weighted A^* Respects a Bound

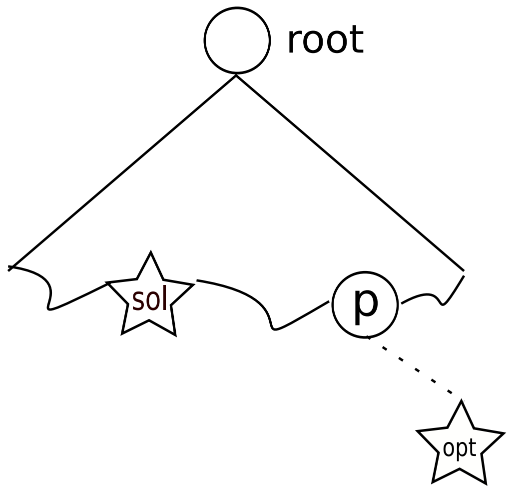


$$f(n) = g(n) + h(n)$$
$$f'(n) = g(n) + w \cdot h(n)$$

$$\begin{aligned} & g(sol) \\ & f'(sol) \leq f'(p) \\ & \quad g(p) + w \cdot h(p) \leq w \cdot (g(p) + h(p)) \\ & \quad\quad w \cdot f(p) \leq w \cdot f(opt) \\ & \quad\quad\quad w \cdot g(opt) \end{aligned}$$

Therefore, $g(sol) \leq w \cdot g(opt)$

Weighted A^* Respects the Bound and Then Some

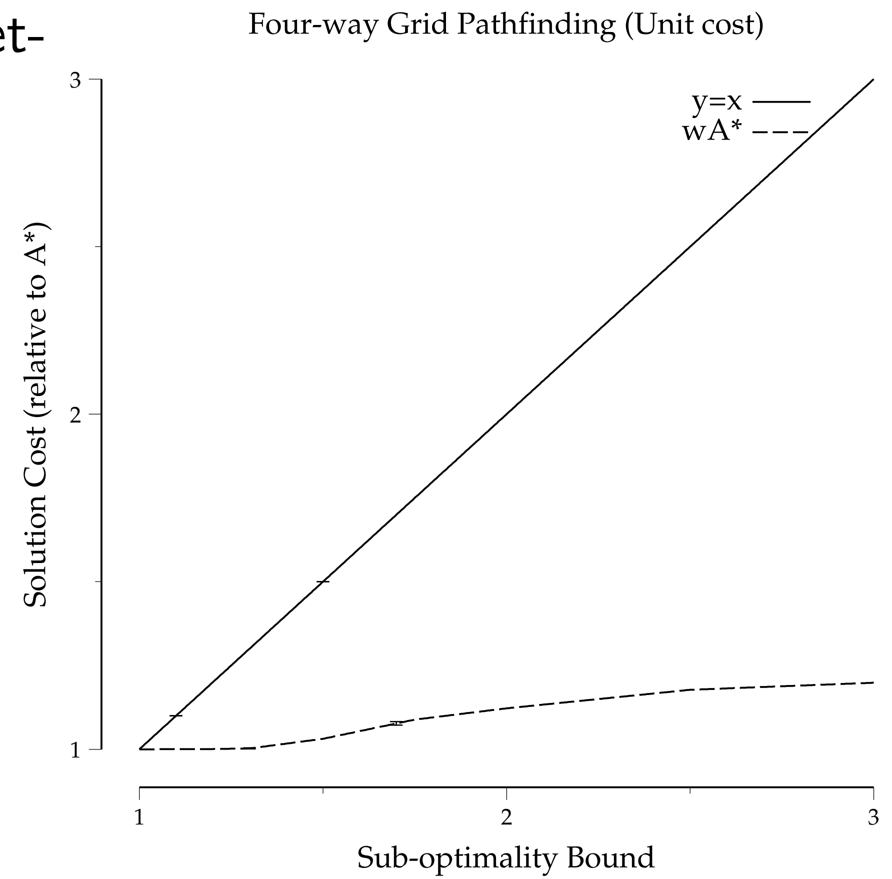


$$f(n) = g(n) + h(n)$$
$$f'(n) = g(n) + w \cdot h(n)$$

$$\begin{array}{lllll}
g(sol) & & & & \\
f'(sol) & \leq & f'(p) & & \\
 & & g(p) + w \cdot h(p) \leq & w \cdot (g(p) + h(p)) & \\
 & & & w \cdot f(p) & \leq w \cdot f(opt) \\
 & & & & w \cdot g(opt)
\end{array}$$

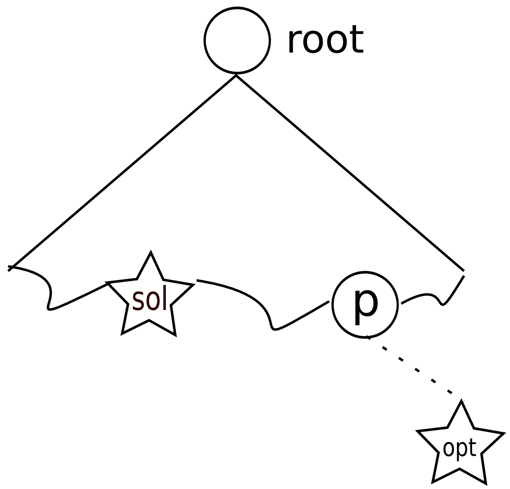
$$g(p) + w \cdot h(p) \leq w \cdot g(p) + w \cdot h(p)$$

- wA^* returns solutions better than the bound.
- Be optimistic
- Run with higher weight



How do we guarantee a suboptimality bound?

Enforcing the Bound



- p is the deepest node on an optimal path to opt

$f(p) \leq f(opt)$
$f(f_{min}) \leq f(p)$
f_{min} provides a lower bound on solution cost.
Determine f_{min} by priority queue sorted on f

Optimistic Search: Run a greedy search
Expand f_{min} until $w \cdot f_{min} \geq f(sol)$

Empirical Evaluation

This Paper:

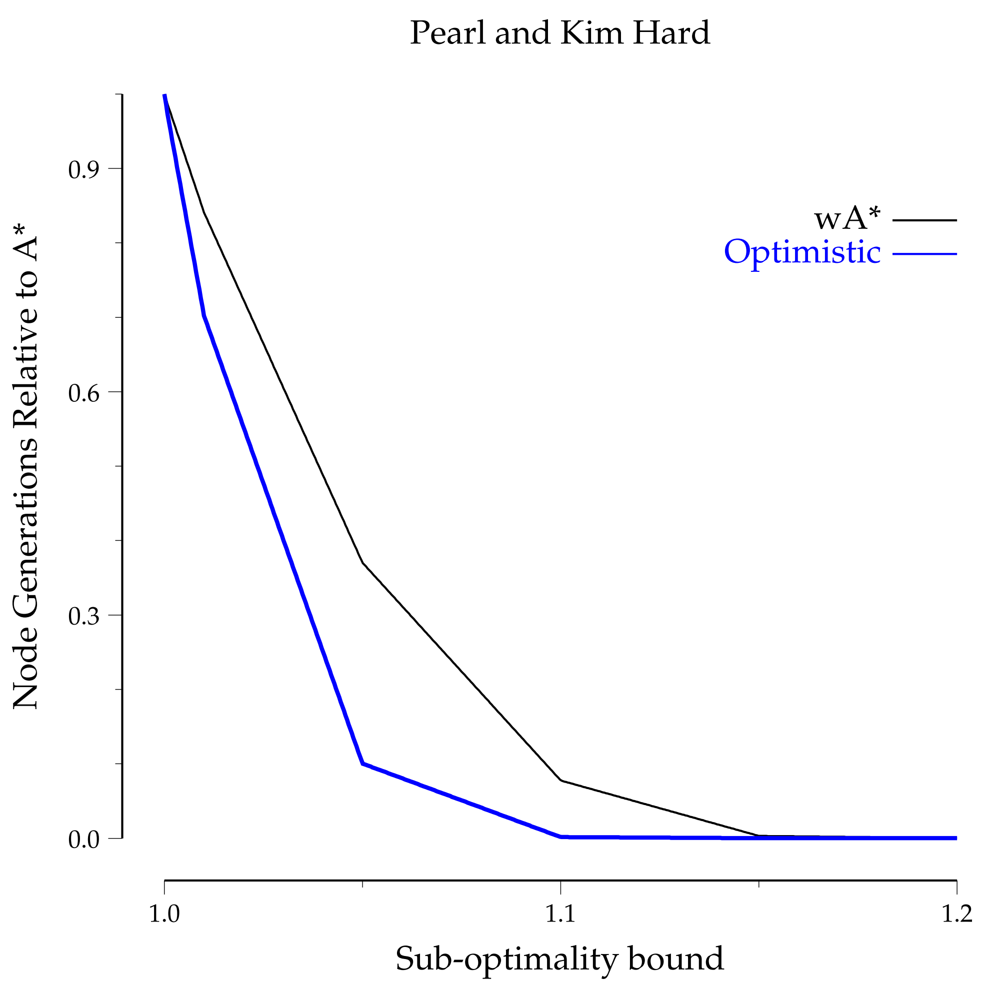
- Grid world path finding
  - Four-way and Eight-way Movement
  - Unit and Life Cost Models
  - 25
- Temporal Planning
  - Blocksworld, Logistics, Rover, Satellite, Zenotravel

To Appear in ICAPS:

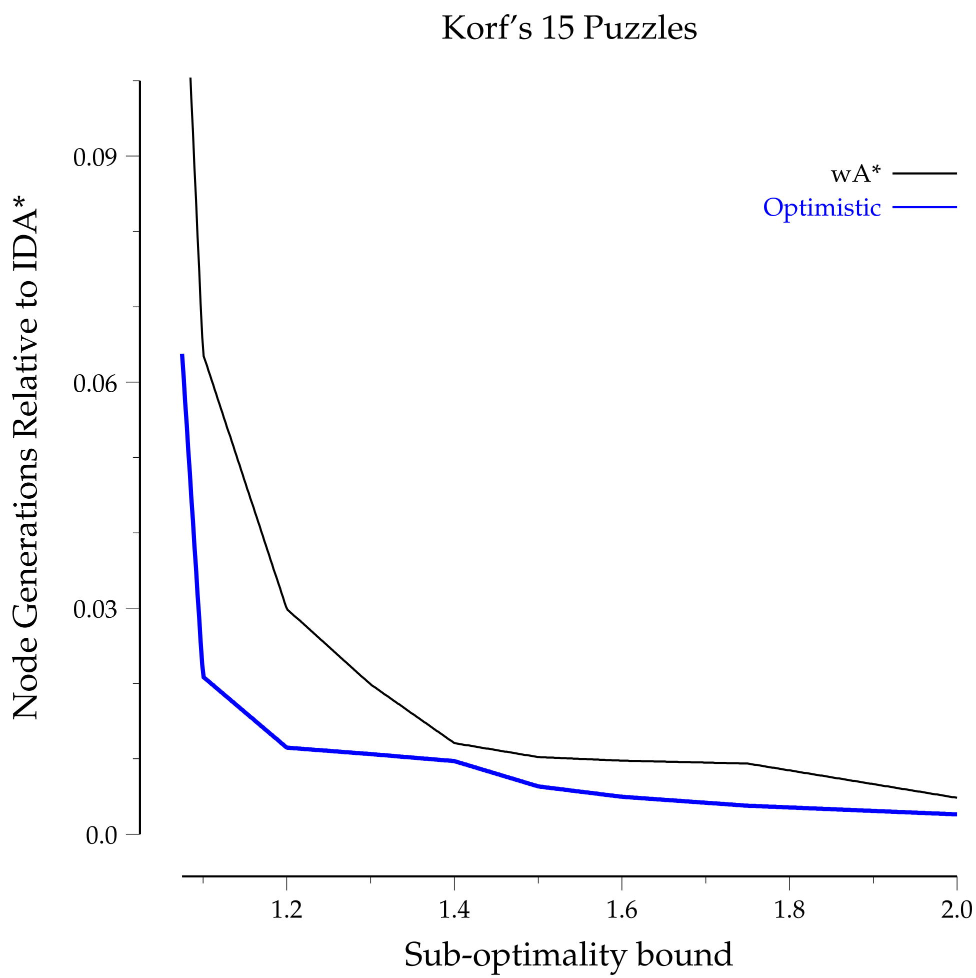
- Traveling Salesman
  - Unit Square
  - Pearl and Kim Hard
- Sliding Tile Puzzles
  - Korf's 100 15-puzzle instances

See papers for details.

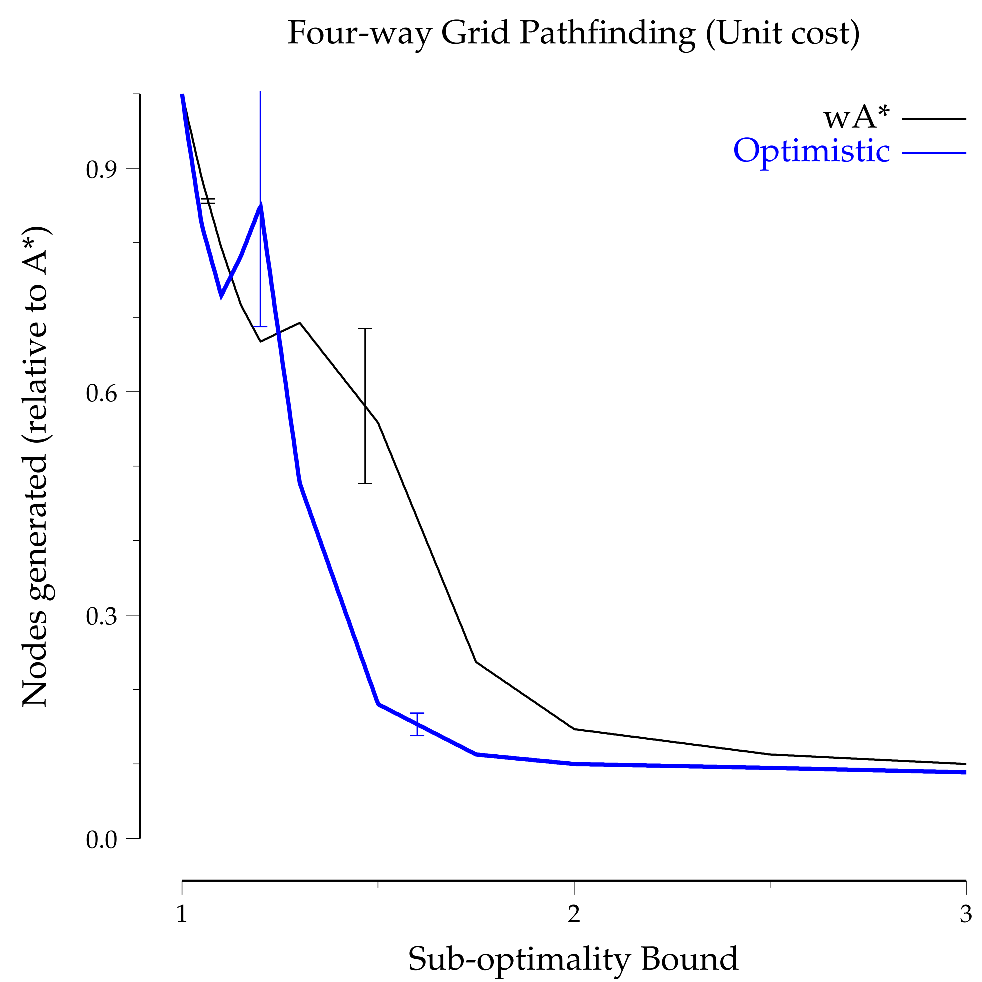
Performance of Optimistic Search



Performance of Optimistic Search



Performance of Optimistic Search



Conclusion

Clamped Adaptive:

- On-line heuristic correction seems promising.
- No parameter tuning needed.

Optimistic Search:

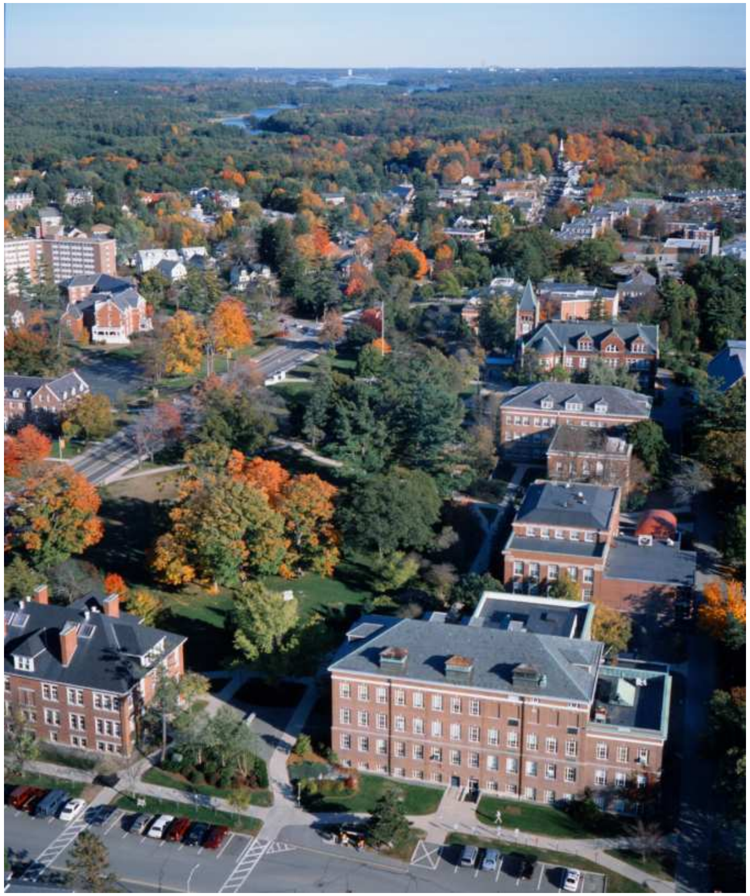
- Performance is predictable.
- Current results are good, could be improved.

We have two algorithms that can outperform weighted A^*

We can use arbitrary heuristics for w-admissible search.

The University of New Hampshire

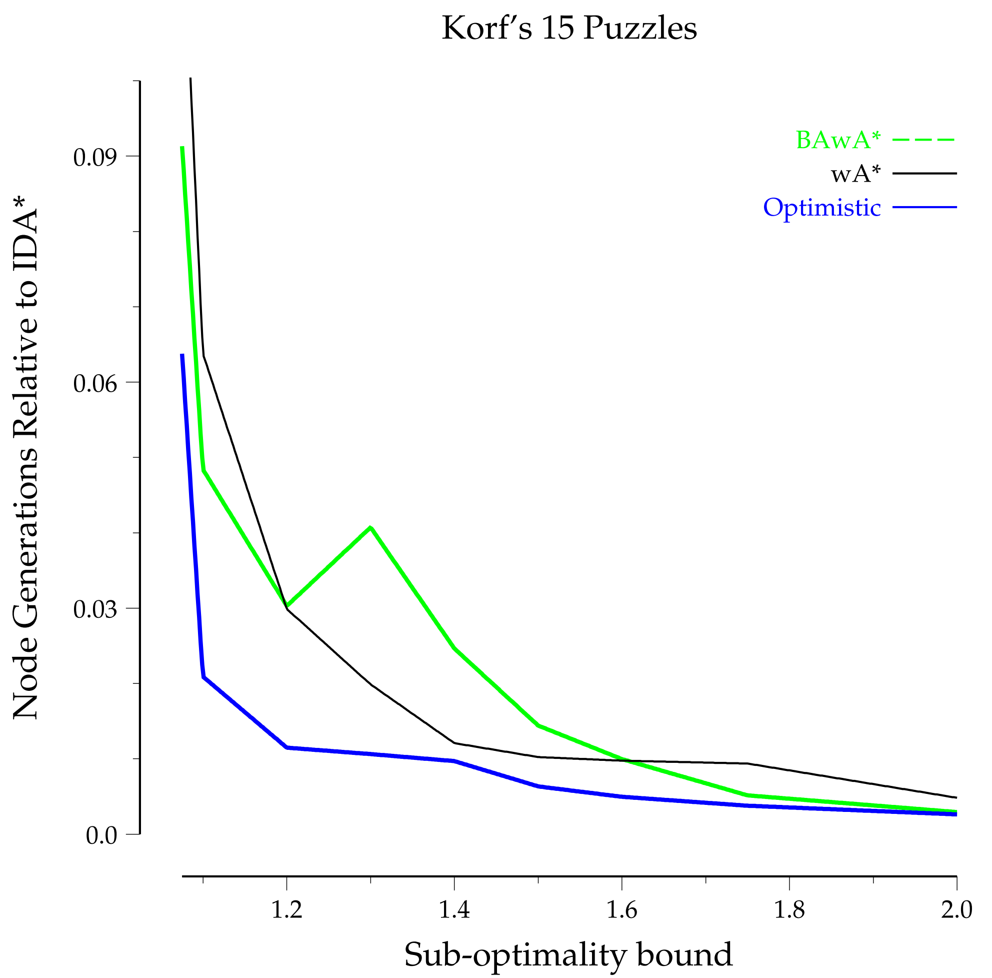
Tell your students to apply to grad school in CS at UNH!



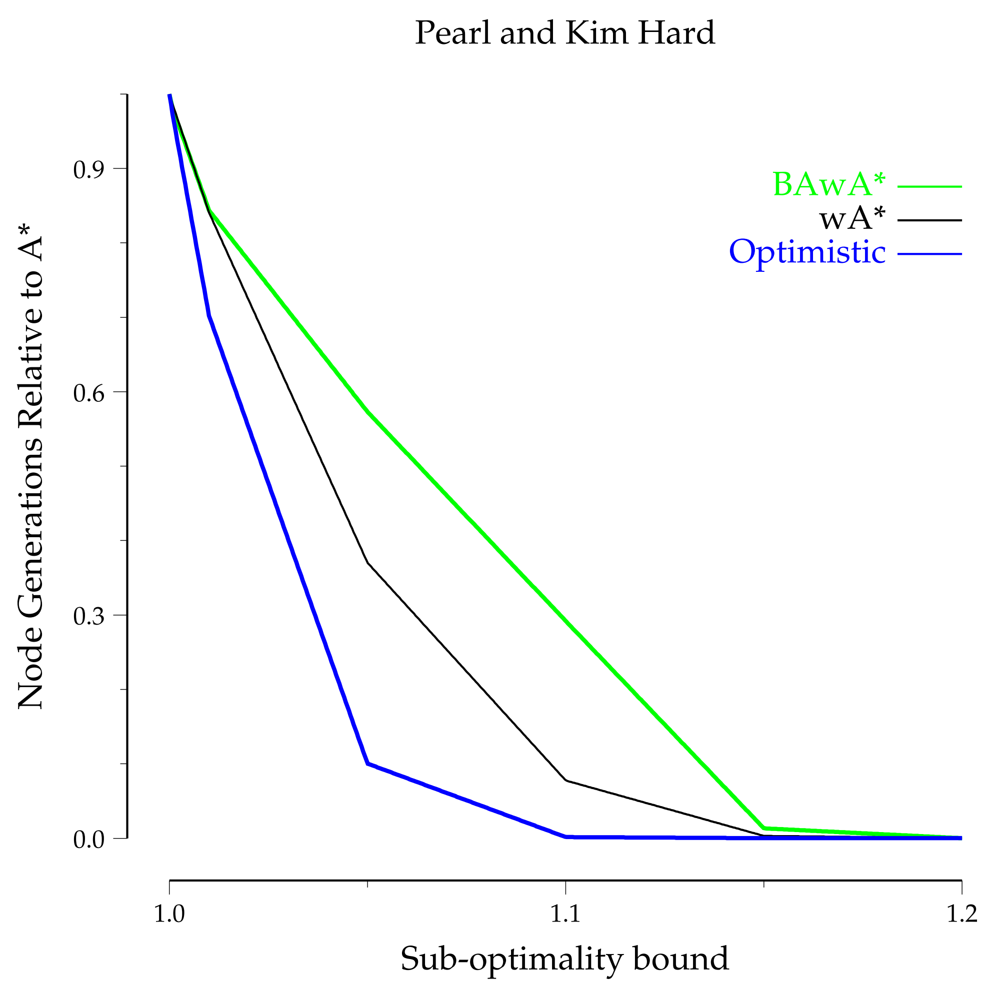
- friendly faculty
- funding
- individual attention
- beautiful campus
- low cost of living
- easy access to Boston, White Mountains
- strong in AI, infoviz, networking, systems, bioinformatics

Bounded Anytime Weighted A*

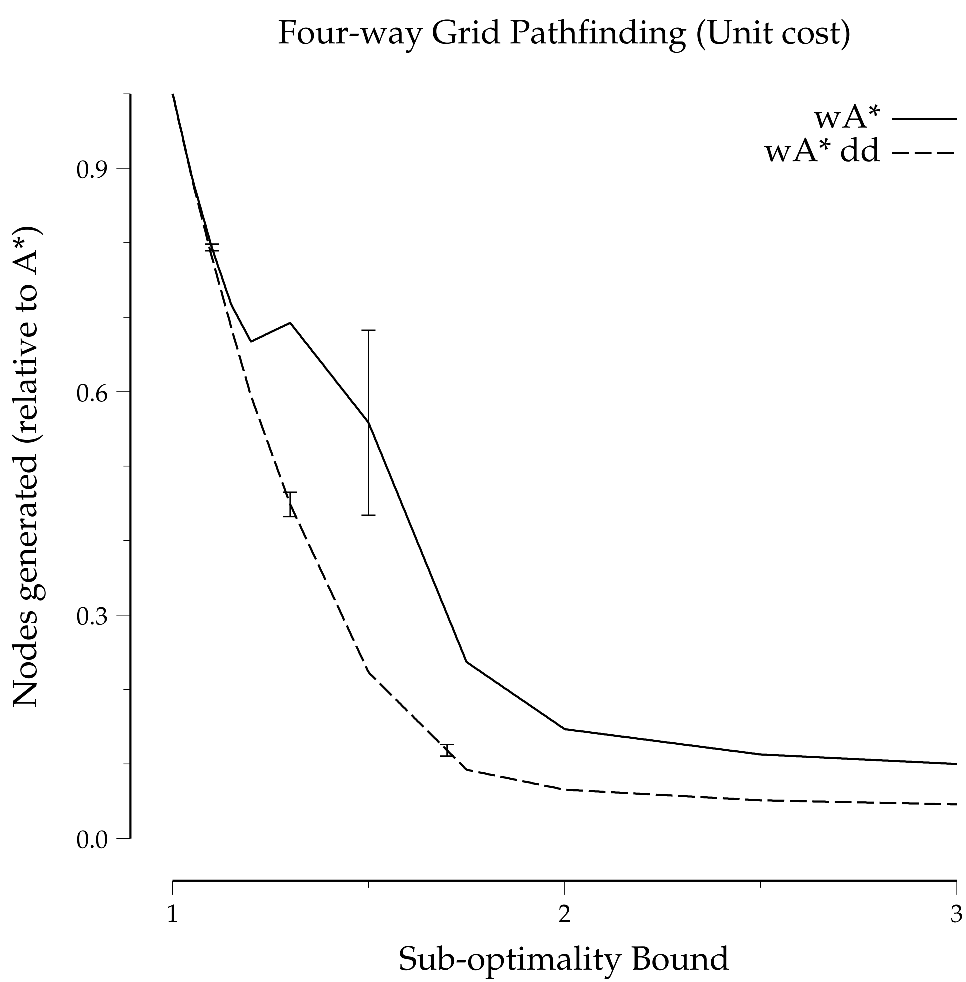
- Introduction
- Weighted A^*
- Clamped Adaptive
- Optimistic Search
- Conclusion
- **Bounded Anytime Weighted A***



Bounded Anytime Weighted A*



Duplicate Dropping can be Important



Sometimes it isn't