Bounded Suboptimal Search: A Direct Approach Using Inadmissible Estimates

Jordan T. Thayer and Wheeler Ruml

University of New Hampshire

jtd7, ruml at cs.unh.edu

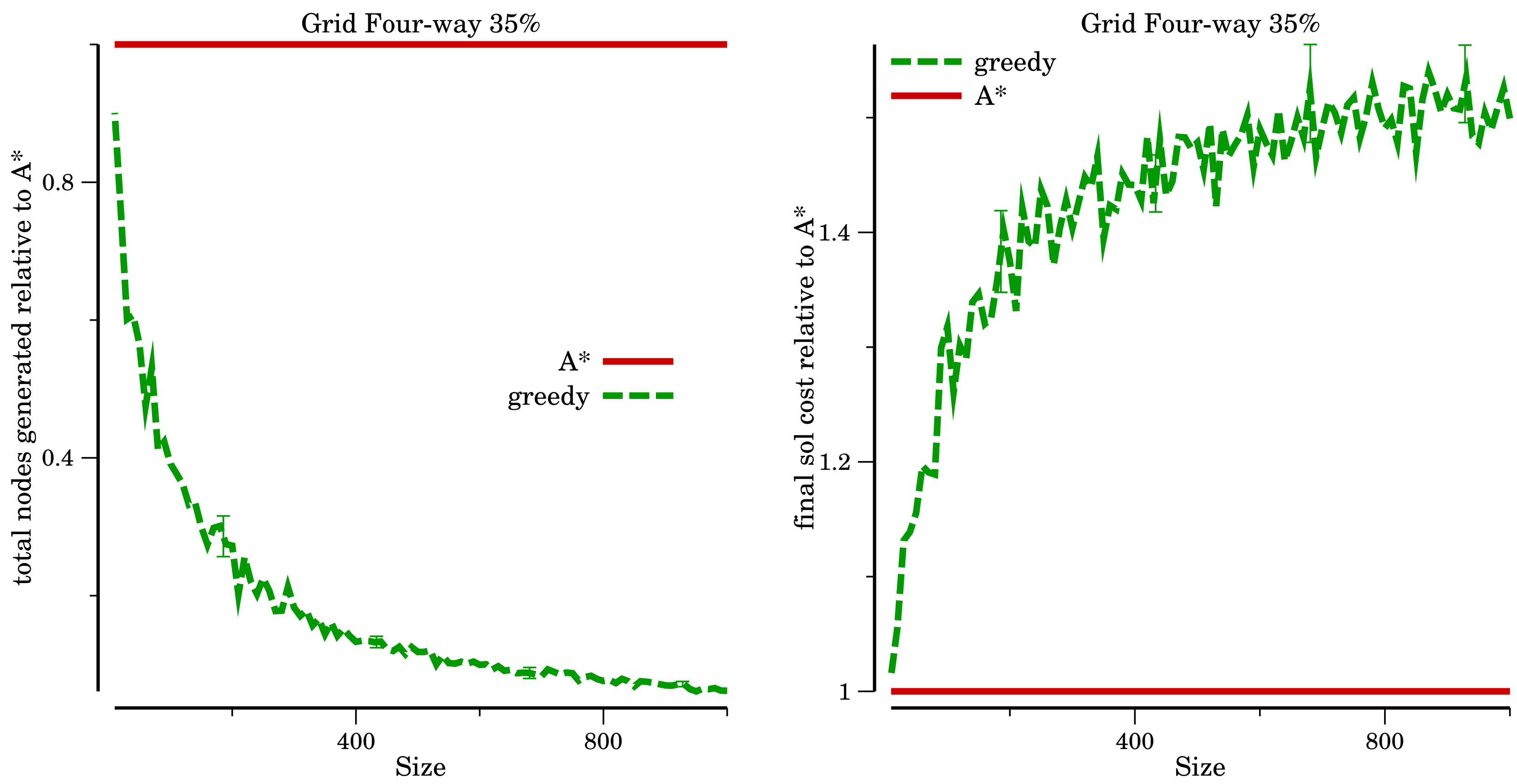
With Thanks To NSF-IIS0812141 and DARPA CSSG N10AP20029

Greedy Search

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion



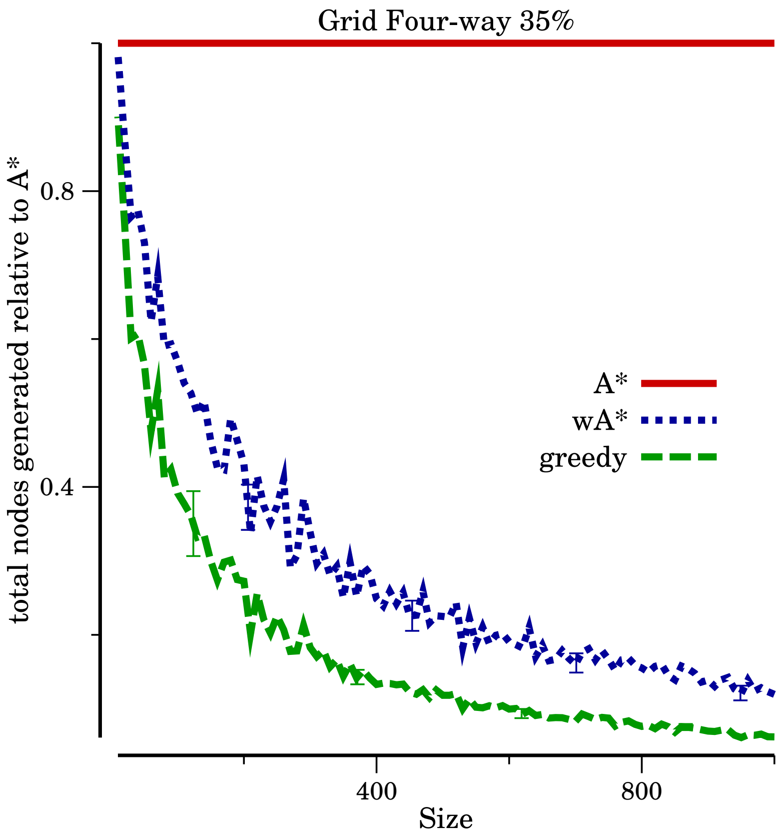
- sacrificing optimality can speed search
- solutions could be arbitrarily bad

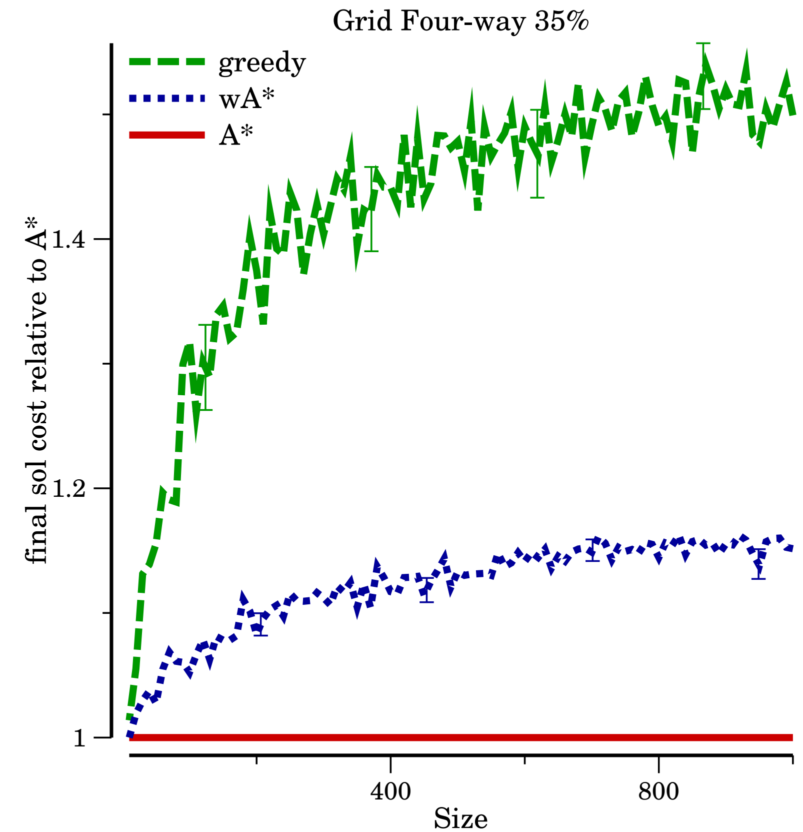
Bounded Suboptimal Search: A Middle Ground

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion





given a suboptimality bound w,
find a solution with cost within a factor w of optimal
as quickly as possible.

Three Useful Ideas

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion

- finding solutions and proving bounds are separate tasks
- inadmissible cost estimates can be more informed
- searching on distance is faster than cost

Three Useful Ideas

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion

- finding solutions and proving bounds are separate tasks
- inadmissible cost estimates can be more informed
- searching on distance is faster than cost

- A^*_ϵ Pearl and Kim, 1982
- Optimistic Search Thayer and Ruml, 2008
- Skeptical Search Thayer and Ruml, 2011

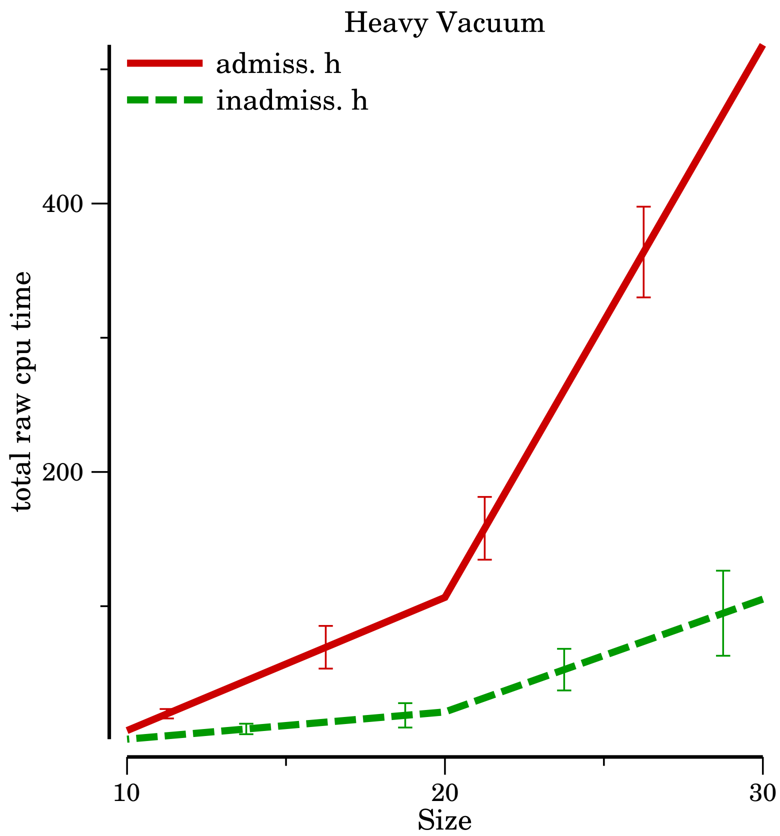
Three Useful Ideas

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion

- finding solutions and proving bounds are separate tasks
- inadmissible cost estimates can be more informed
- searching on distance is faster than cost



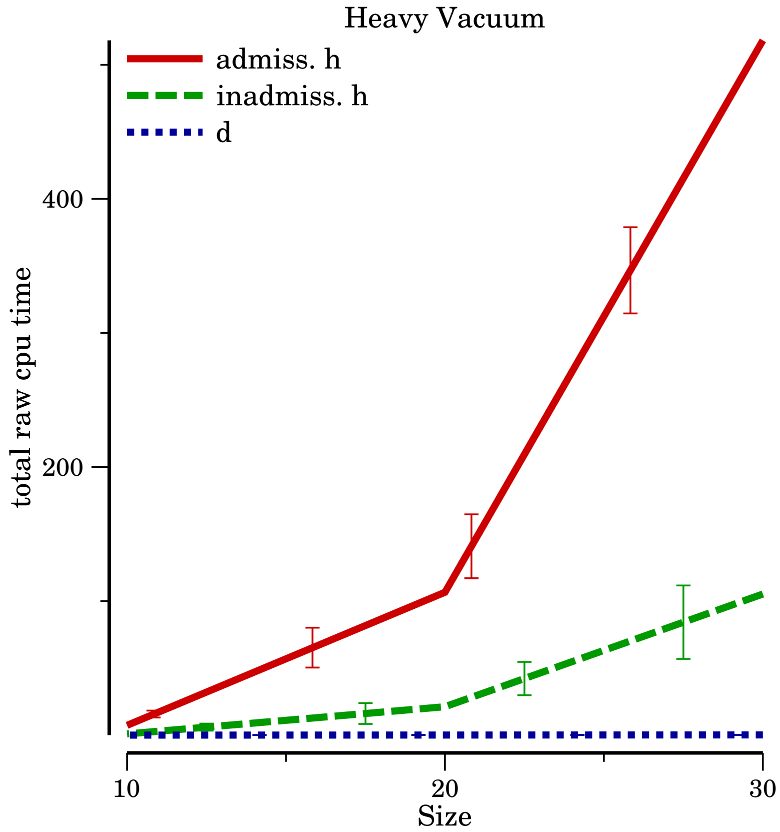
Three Useful Ideas

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion

- finding solutions and proving bounds are separate tasks
- inadmissible cost estimates can be more informed
- searching on distance is faster than cost



Three Useful Ideas

Outline
- Greedy Search
- Bounded Search
- Three Ideas

EES

Conclusion

- finding solutions and proving bounds are separate tasks
- inadmissible cost estimates can be more informed
- searching on distance is faster than cost

Explicit Estimation Search (EES) combines these three ideas.

Outline

EES
- Direct Approach
- Three Heuristics
- EES

Conclusion

Explicit Estimation Search

A Direct Approach

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

minimize solving time subject to suboptimality bound w

A Direct Approach

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

minimize solving time subject to suboptimality bound w

weighted A* ($f'(n) = g(n) + w \cdot h(n)$) is simple but ad hoc

(Pohl, AIJ vol 1, 1970)

A Direct Approach

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

minimize solving time subject to suboptimality bound w

weighted A* ($f'(n) = g(n) + w \cdot h(n)$) is simple but ad hoc

(Pohl, AIJ vol 1, 1970)

expand the node closest to a solution within the bound

$best_{\widehat{d}}$: node estimated within bound closest to a goal

Three Heuristic Sources Of Information

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

1. h: an admissible estimate of cost-to-go

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

finding solutions and proving bounds are separate tasks

Three Heuristic Sources Of Information

1. h: an admissible estimate of cost-to-go

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

   finding solutions and proving bounds are separate tasks

2. $\widehat{h}$: a potentially inadmissible estimate of cost-to-go

   inadmissible cost estimates can be more informed

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

   (Thayer and Ruml, ICAPS-11)

Three Heuristic Sources Of Information

Outline
EES
- Direct Approach
- Three Heuristics
- EES
Conclusion

1. h: an admissible estimate of cost-to-go

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

   finding solutions and proving bounds are separate tasks

2. $\widehat{h}$: a potentially inadmissible estimate of cost-to-go

   inadmissible cost estimates can be more informed

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

   (Thayer and Ruml, ICAPS-11)

3. $\widehat{d}$: a potentially inadmissible estimate of distance-to-go

   searching on distance is faster than cost

   (Pearl and Kim, IEEE PAMI 1982,
   Thayer et al, ICAPS-09)

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

$best_f$: open node with minimum f

$$\operatorname*{argmin}_{n \in open} f(n)$$

Finding $best_{\widehat{d}}$

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

$best_f$: open node with minimum f

$$\operatorname*{argmin}_{n \in open} f(n)$$

$best_{\widehat{f}}$: open node with minimum $\widehat{f}$

$$\operatorname*{argmin}_{n \in open} \widehat{f}(n)$$

Finding $best_{\widehat{d}}$

$best_f$: open node with minimum f

$$\operatorname*{argmin}_{n \in open} f(n)$$

$best_{\widehat{f}}$: open node with minimum $\widehat{f}$

$$\operatorname*{argmin}_{n \in open} \widehat{f}(n)$$

pursuing the shortest solution within the bound should be fast

$best_{\widehat{d}}$: estimated w-admissible node with minimum $\widehat{d}$

$$\operatorname*{argmin}_{n \in open \wedge \widehat{f}(n) \leq w \cdot \widehat{f}(best_{\widehat{f}})} \widehat{d}(n)$$

Outline
EES
- Direct Approach
- Three Heuristics
- EES
Conclusion

$best_f$: open node with minimum f
$best_{\widehat{f}}$: open node with minimum $\widehat{f}$
$best_{\widehat{d}}$: estimated w-admissible node with minimum $\widehat{d}$

node to expand next:

1. pursue the shortest solution that is within the bound.
2.
3.

in other words:

1. $best_{\widehat{d}}$
2.
3.

EES Expansion Order

Outline

EES
- Direct Approach
- Three Heuristics
- EES

Conclusion

$best_f$: open node with minimum f
$best_{\widehat{f}}$: open node with minimum $\widehat{f}$
$best_{\widehat{d}}$: estimated w-admissible node with minimum $\widehat{d}$

node to expand next:

1. pursue the shortest solution that is within the bound.
2.
3.

in other words:

1. **if** $\widehat{f}(best_{\widehat{d}}) \leq w \cdot f(best_f)$ **then** $best_{\widehat{d}}$
2.
3.

note that $f(best_f) \leq f(opt)$ and $f(n) \leq \widehat{f}(n)$

EES Expansion Order

Outline
EES
Direct Approach
Three Heuristics
EES
Conclusion

$best_f$: open node with minimum f
$best_{\widehat{f}}$: open node with minimum $\widehat{f}$
$best_{\widehat{d}}$: estimated w-admissible node with minimum $\widehat{d}$

node to expand next:

1. pursue the shortest solution that is within the bound.
2. pursue the optimal solution.
3.

in other words:

1. **if** $\widehat{f}(best_{\widehat{d}}) \leq w \cdot f(best_f)$ **then** $best_{\widehat{d}}$
2. **else if** $\widehat{f}(best_{\widehat{f}}) \leq w \cdot f(best_f)$ **then** $best_{\widehat{f}}$
3.

EES Expansion Order

Outline

EES

- Direct Approach
- Three Heuristics
- EES

Conclusion

$best_f$: open node with minimum f

$best_{\widehat{f}}$: open node with minimum $\widehat{f}$

$best_{\widehat{d}}$: estimated w-admissible node with minimum $\widehat{d}$

node to expand next:

1. pursue the shortest solution that is within the bound.
2. pursue the optimal solution.
3. raise the lower bound on optimal solution cost.

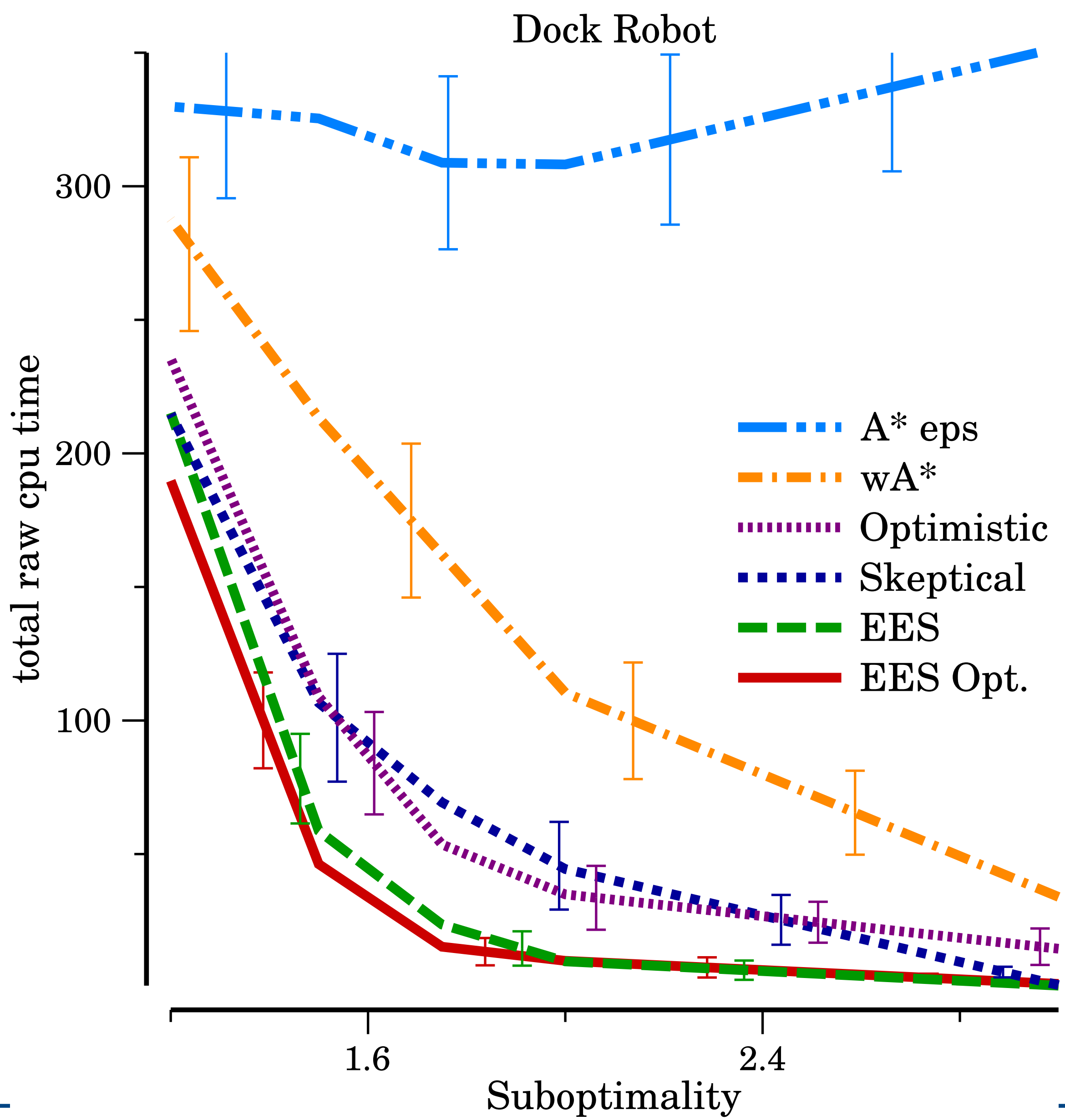
in other words:

1. **if** $\widehat{f}(best_{\widehat{d}}) \leq w \cdot f(best_f)$ **then** $best_{\widehat{d}}$
2. **else if** $\widehat{f}(best_{\widehat{f}}) \leq w \cdot f(best_f)$ **then** $best_{\widehat{f}}$
3. **else** $best_f$

see paper for further justification

EES Results

Outline
EES
Direct Approach
Three Heuristics
EES
Conclusion



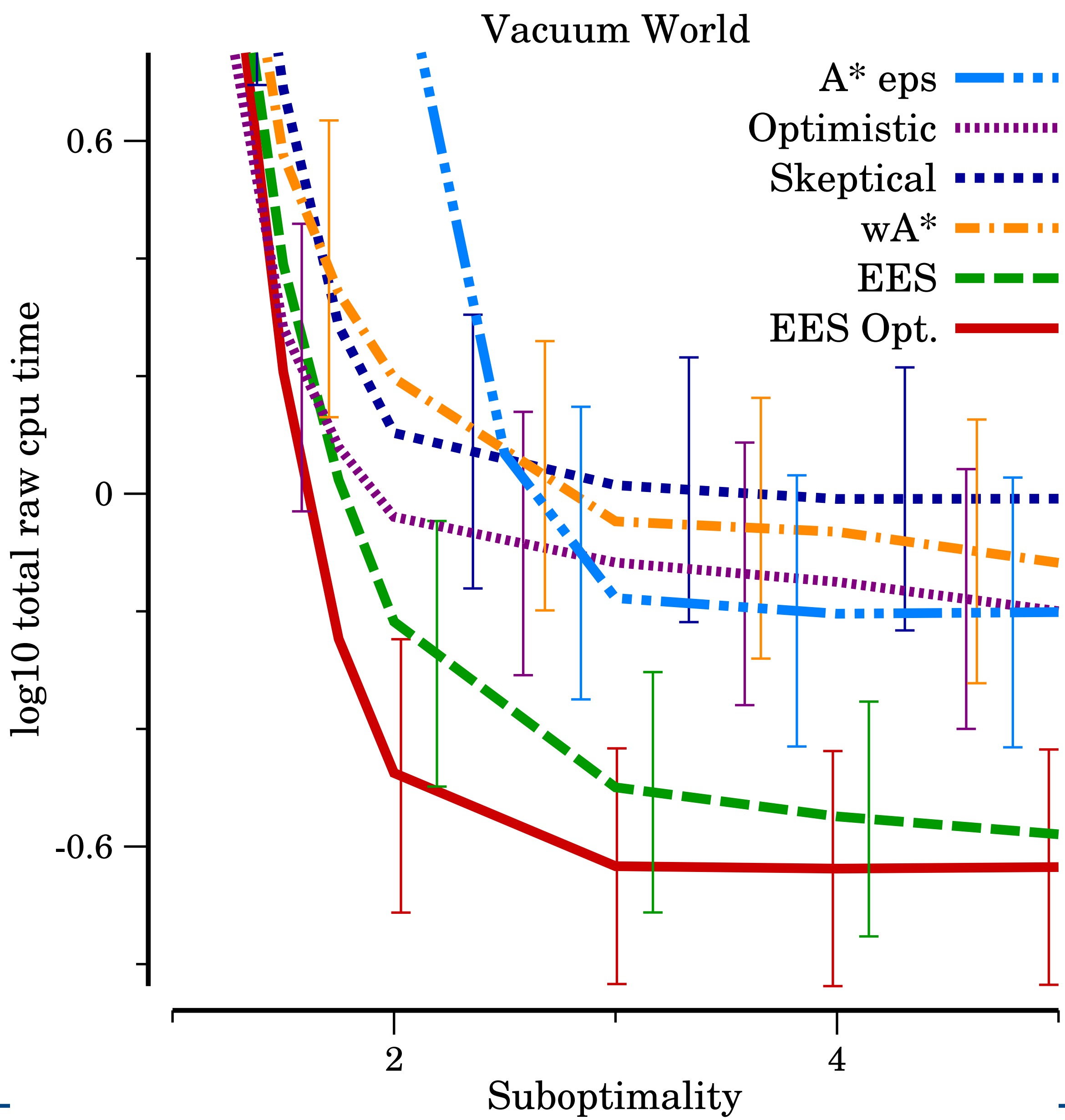
EES Results

Outline

EES

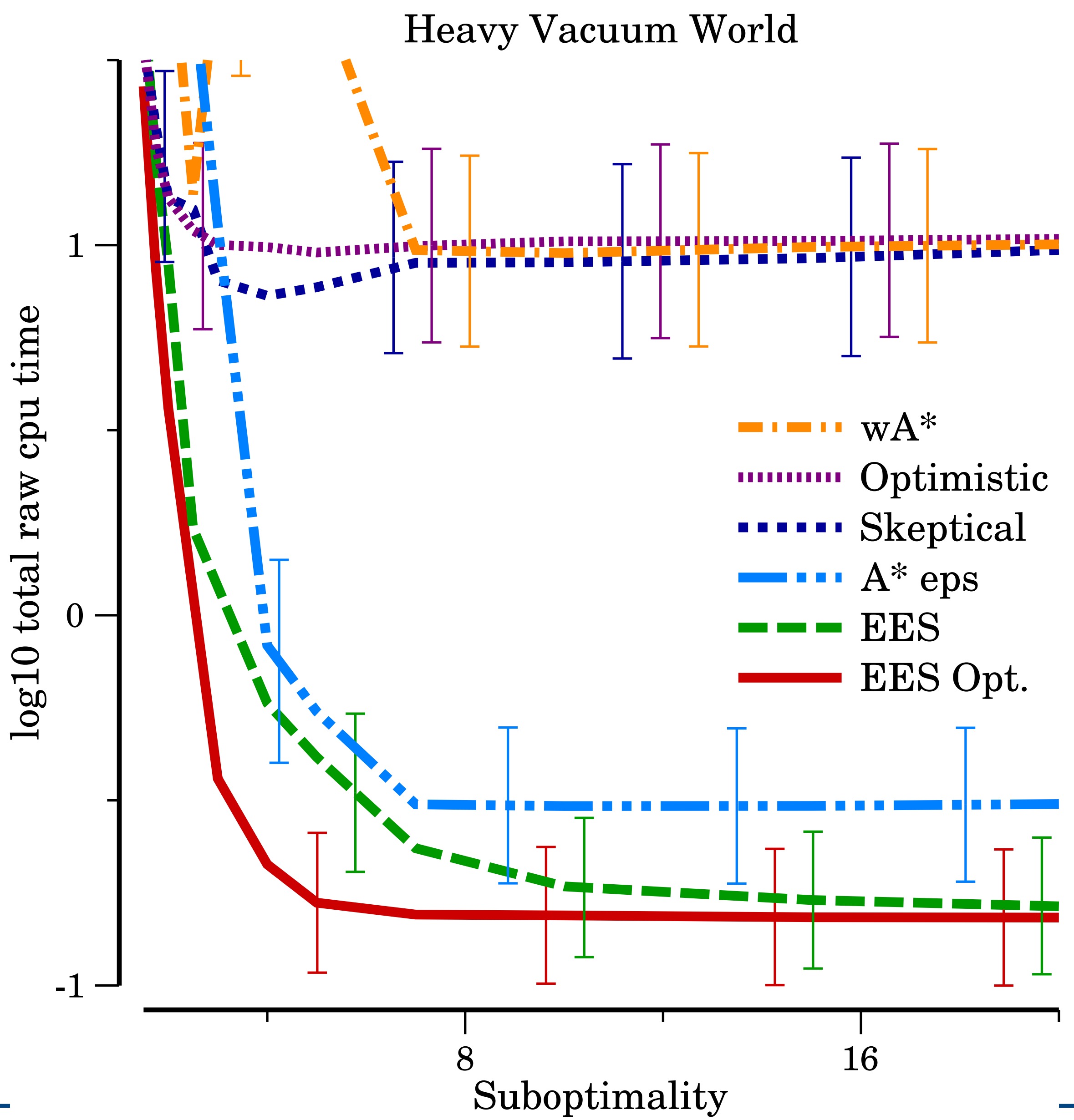
- Direct Approach
- Three Heuristics
- EES

Conclusion



EES Results

Outline
EES
- Direct Approach
- Three Heuristics
- EES
Conclusion



Contributions

Outline
EES
Conclusion

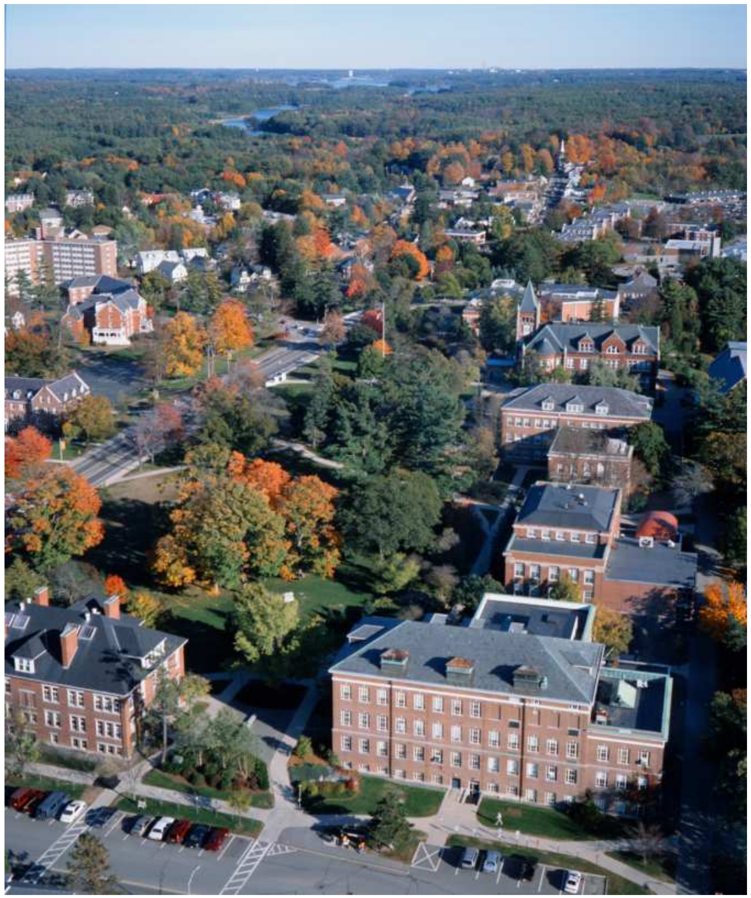
Explicit Estimation Search (EES)

- follows directly from the objectives of bounded suboptimal search
- state of the art search bounded suboptimal search
- use inadmissible heuristics without losing bounds
- robust, works best in domains with action costs

The University of New Hampshire

Outline
EES
Conclusion

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

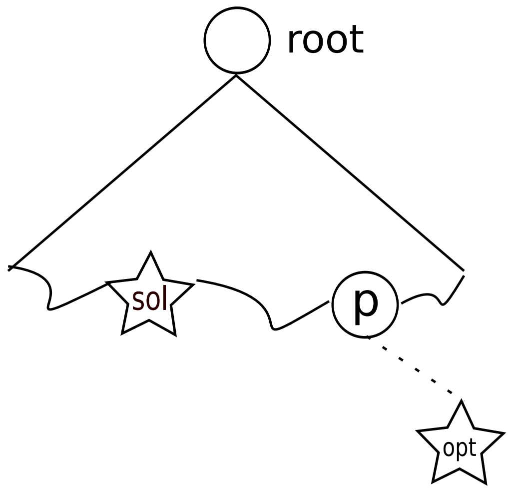
EES Nodes

Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure

$$
\begin{aligned}
best_{\widehat{f}} &= \operatorname*{argmin}_{n \in open} \widehat{f}(n) \\
best_{\widehat{d}} &= \operatorname*{argmin}_{n \in open \wedge \widehat{f}(n) \leq w \cdot \widehat{f}(best_{\widehat{f}})} \widehat{d}(n) \\
best_{f} &= \operatorname*{argmin}_{n \in open} f(n)
\end{aligned}
$$

EES Respects a Bound

Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure



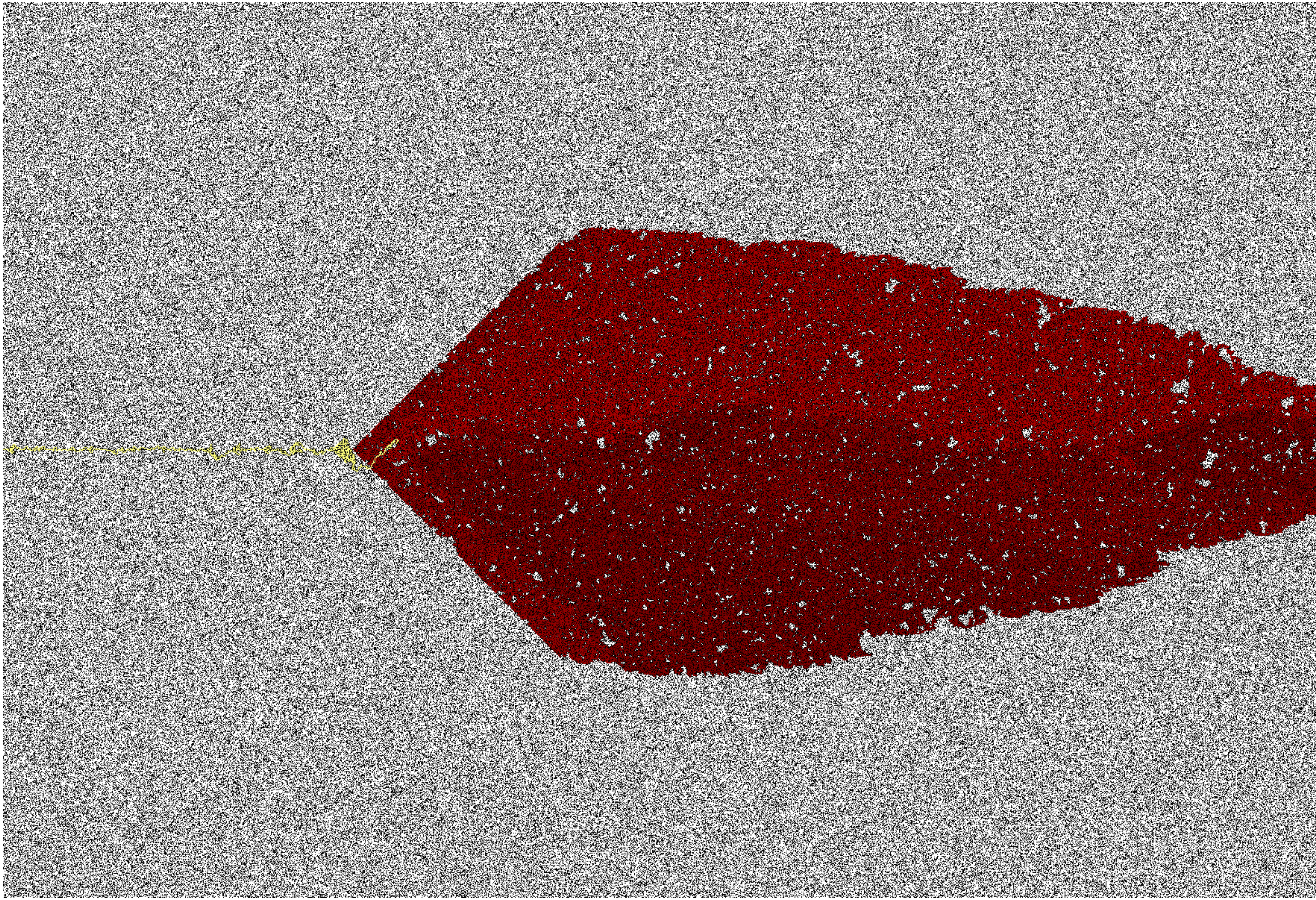
- p is the deepest node on an optimal path to opt.
- $best_f$ is the node with the smallest f value.

$$\begin{aligned} f(p) &\leq f(opt) \\ f(best_f) &\leq f(p) \end{aligned}$$

$best_f$ provides a lower bound on solution cost.

determine $best_f$ by priority queue sorted on f

Why Doesn't A^*_ϵ Work Well?



EES Overhead

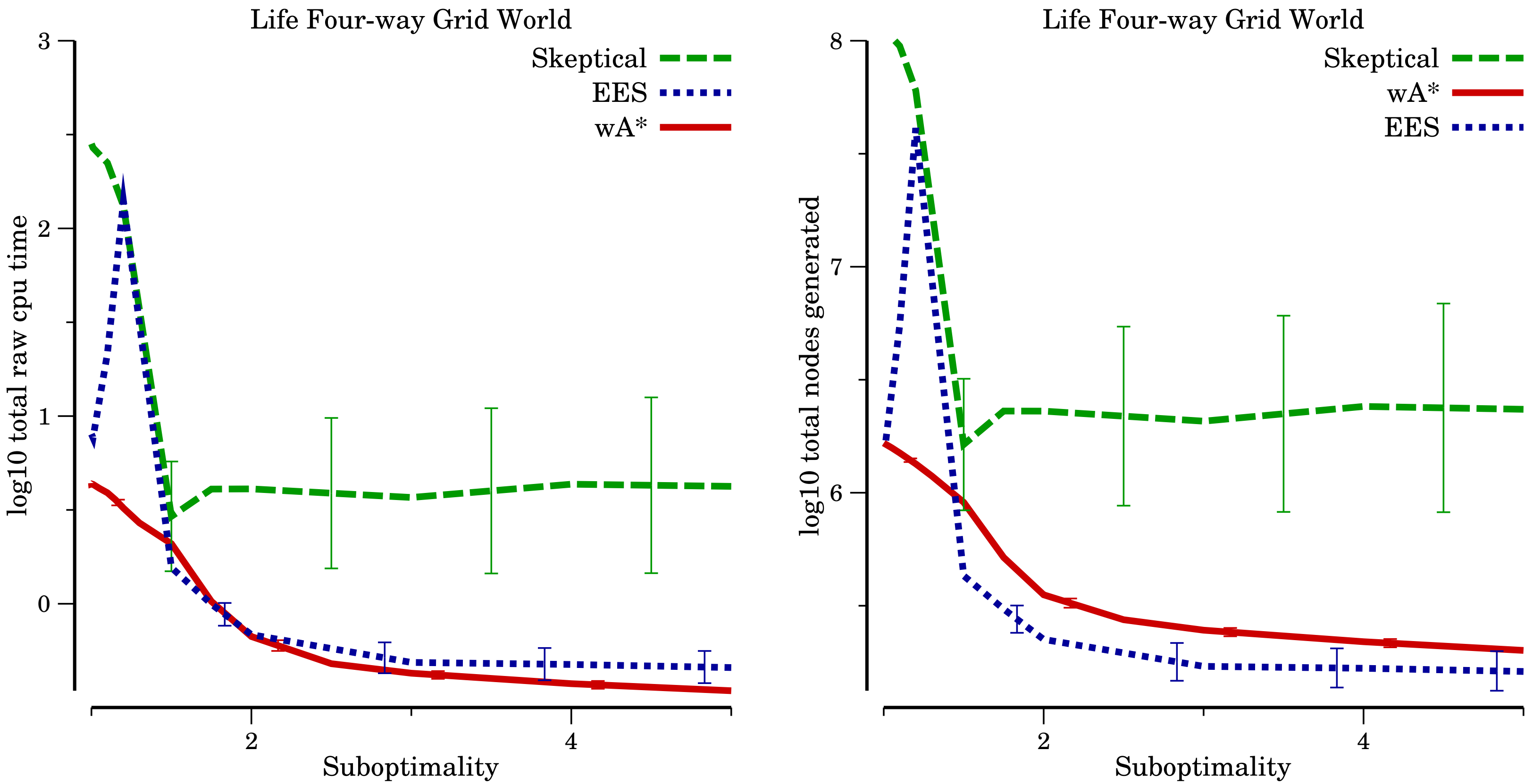
Outline

EES

Conclusion

Backup Slides

- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure



Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure

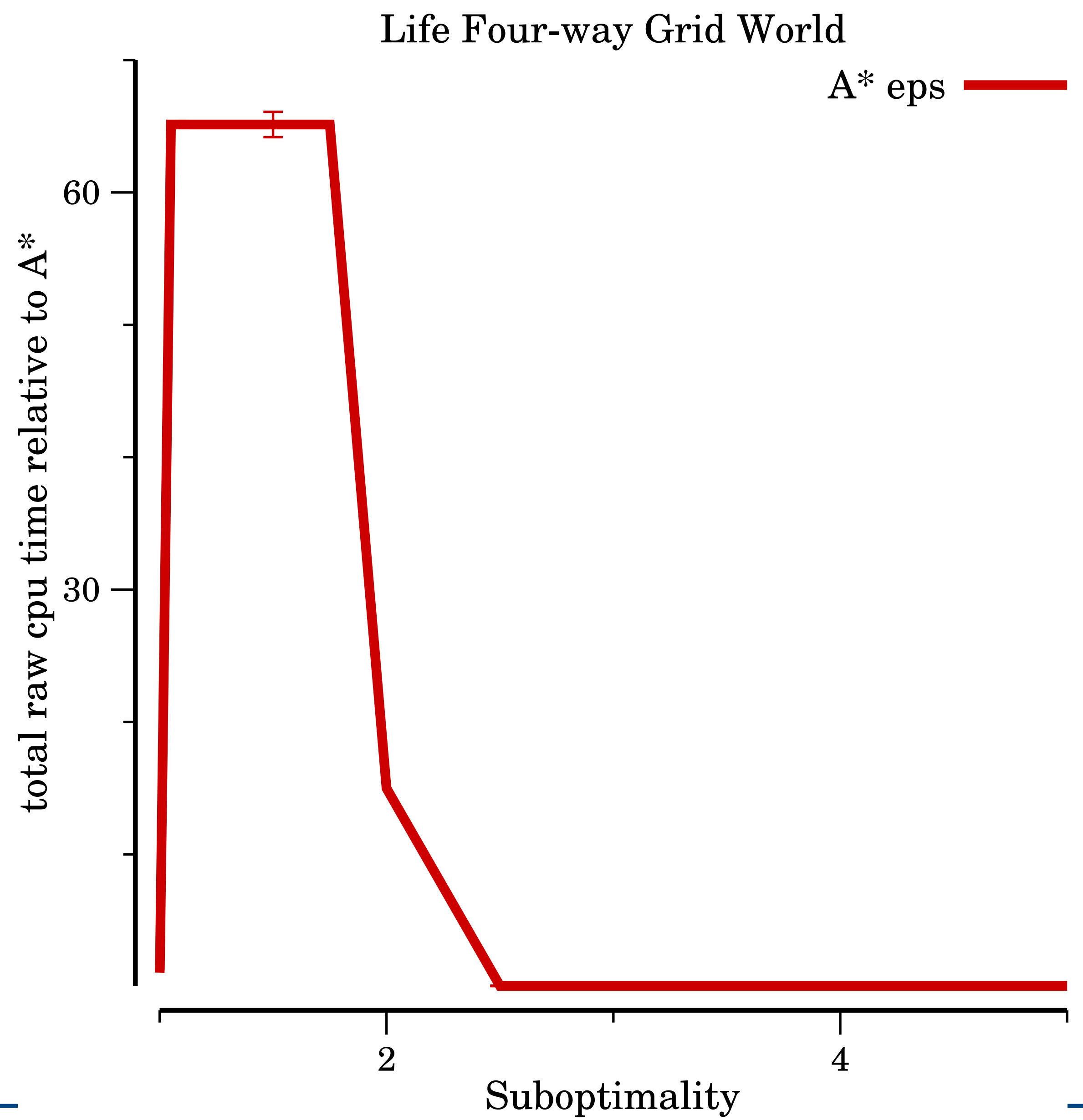
intuition: of all solutions within the bound, the nearest should be the fastest to find.

$f(n) = g(n) + h(n)$
$best_f$: generated but unexpanded node with minimum f

best-first search on two lists:
open: all generated but unexpanded nodes, sorted on $f(n)$
focal: all nodes where $f(n) \leq w \cdot f(best_f)$ sorted on $\widehat{d}(n)$

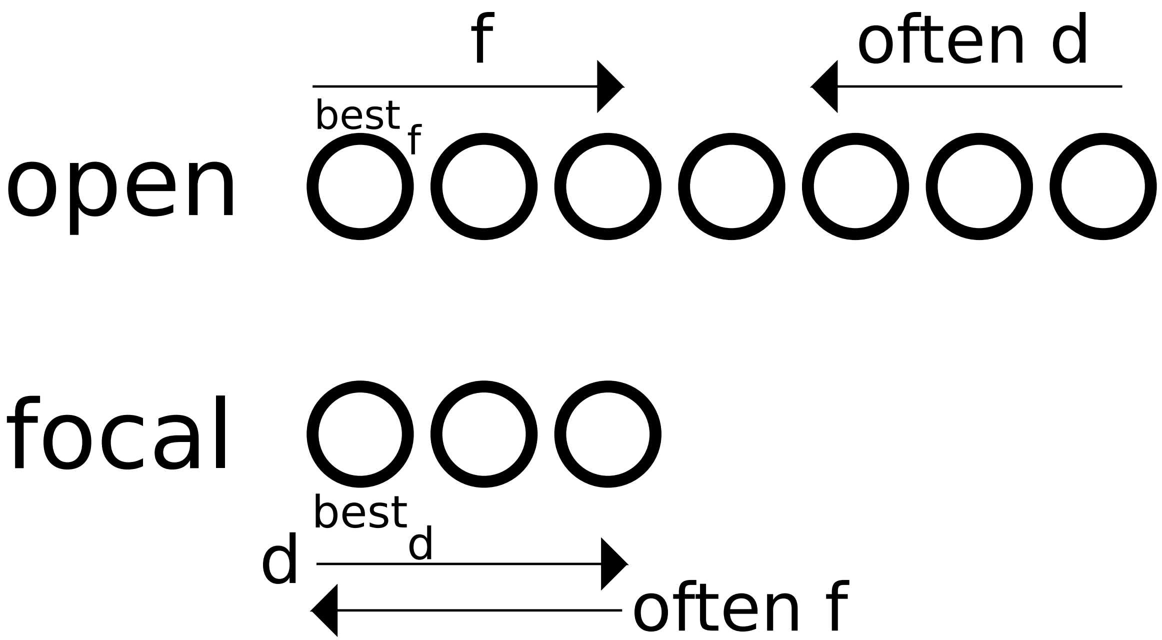
expand the best node from *focal*

Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure



Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure

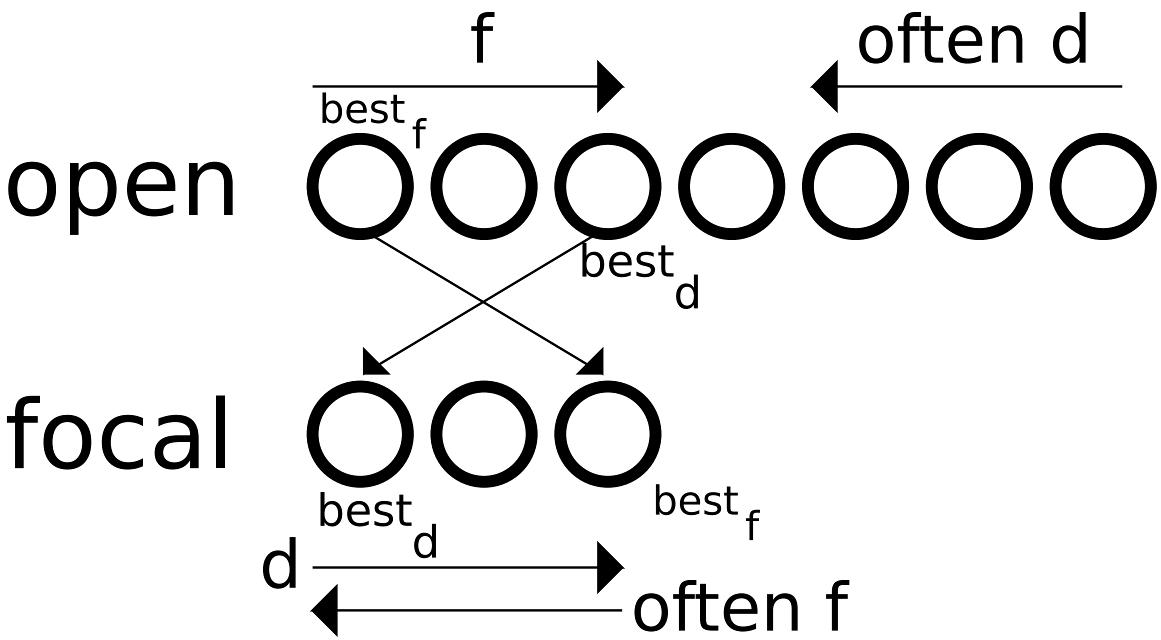
open: all generated but unexpanded nodes, sorted on $f(n)$
focal: all nodes where $f(n) \leq w \cdot f(best_f)$ sorted on $\widehat{d}(n)$



Why Doesn't A^*_ϵ Work Well?

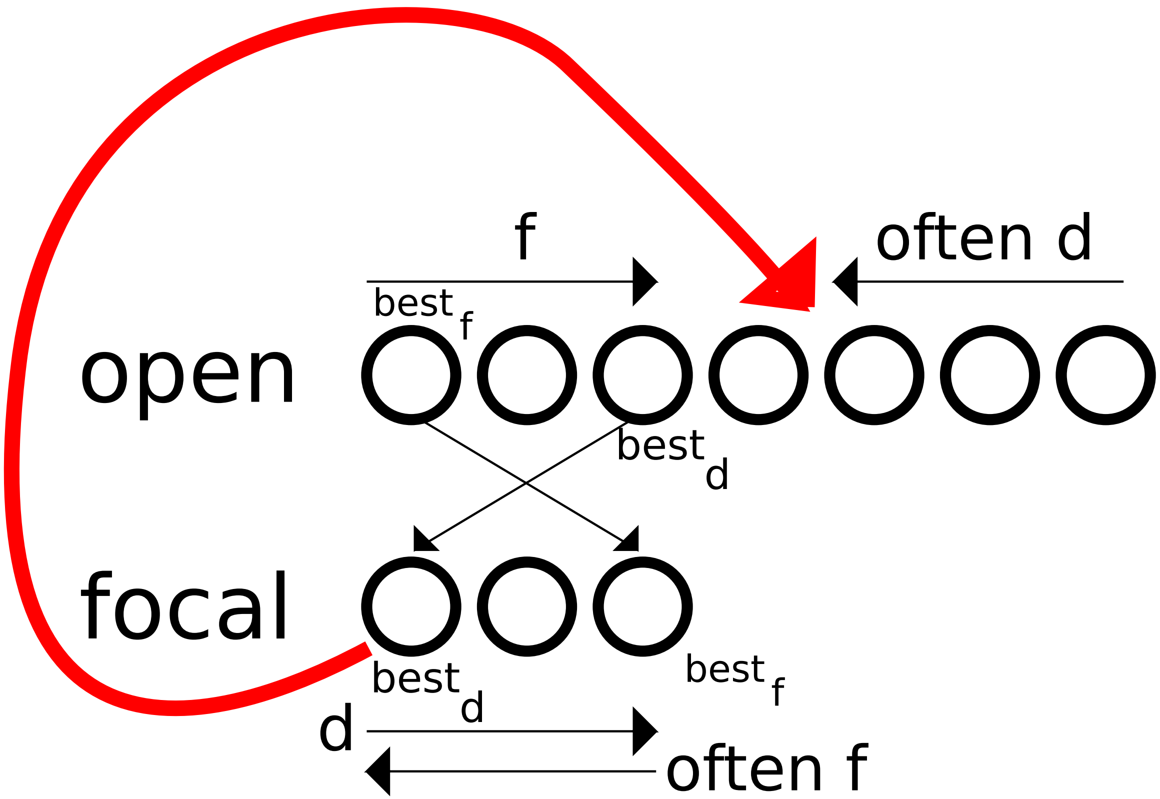
Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A^*_ϵ
- A^*_ϵ Failure

open: all generated but unexpanded nodes, sorted on $f(n)$
focal: all nodes where $f(n) \leq w \cdot f(best_f)$ sorted on $\widehat{d}(n)$



Outline
EES
Conclusion
Backup Slides
- EES Nodes
- Bound
- Overhead
- A_ϵ^*
- A_ϵ^* Failure

open: all generated but unexpanded nodes, sorted on $f(n)$
focal: all nodes where $f(n) \leq w \cdot f(best_f)$ sorted on $\widehat{d}(n)$



f rises as search progresses (h is admissible)
$best_{\widehat{d}}$'s children won't remain on focal