Using Distance Estimates in Search: A Re-evaluation
Jordan Thayer and Wheeler Ruml and Jeff Kreis

Can using distance-to-go estimates improve the performance of bounded suboptimal search?

Greedy on distance-to-go, \( d \), beats greedy on cost-to-go, \( h \).

Searching on distance works because cheap paths may be longer in terms of search effort.

Previous Work:
Dynamically weighted \( A^\ast \) decreases greediness as depth increases.

\( A^\ast \) expands the node closest to the goal that is within the suboptimality bound.

Our Contributions:

Tie-breaking on \( d \)
Finds optimal solutions in half the time of \( A^\ast \).

Surprising number of nodes in last \( f(n) \) layer of standard benchmarks.

Revised Dynamically Weighted \( A^\ast \)
Original algorithm rewards depth, not progress towards goal.

New formulation:
\[ f_{rdwA^\ast}(n) = g(n) + h(n) \cdot \max\left(\frac{d(n)}{d(root)} \cdot w, 1\right) \]

In temporal planning, using distance estimates avoids catastrophe at high weights.

When using distance estimates does not drastically improve the search, it often does not significantly harm it.

Node orderings for \( A^\ast \) interfere with one another.

Simple modifications of \( A^\ast \) improve performance drastically.