The Consistent Case in Bidirectional Search and a Bucket-to-Bucket Algorithm

Vidal Alcázar
vidal.alcazar@riken.jp

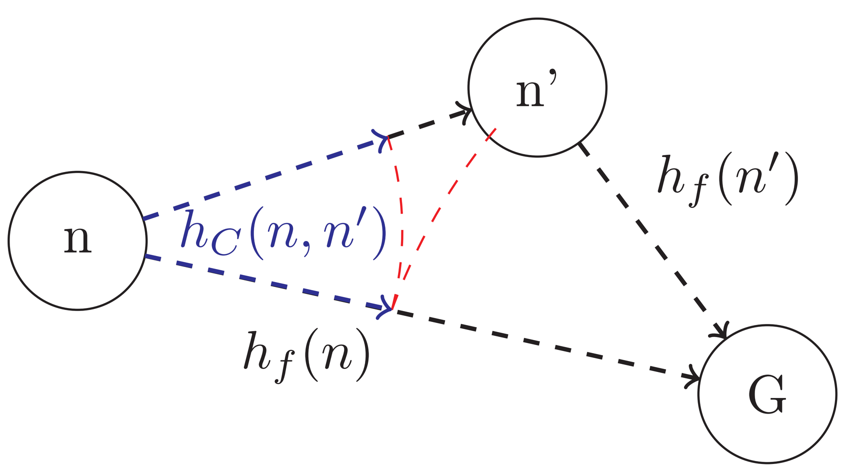
革新知能統合研究センター
Center for Advanced Intelligence Project

Introduction

Here we create a new definition of must-expand pairs when consistency is exploited, and analyze h_C and individual bounds. The lower bound of pairs can also be seen as an estimation of the lowest cost of any path between both states. This cost depends only on node values values; thus, by grouping nodes by these values in buckets, such an estimate can be computed for sets of nodes. This bucket-to-bucket computation, although as expensive as front-to-front in the worst case, allows implementing a *near-optimal* algorithm in the the I_{CON}/I_{CON} case. Experiments show that bucket-to-bucket is the state of the art in the Pancake Problem and offer an insightful measurement of how far front-to-end algorithms are from their theoretical limit.

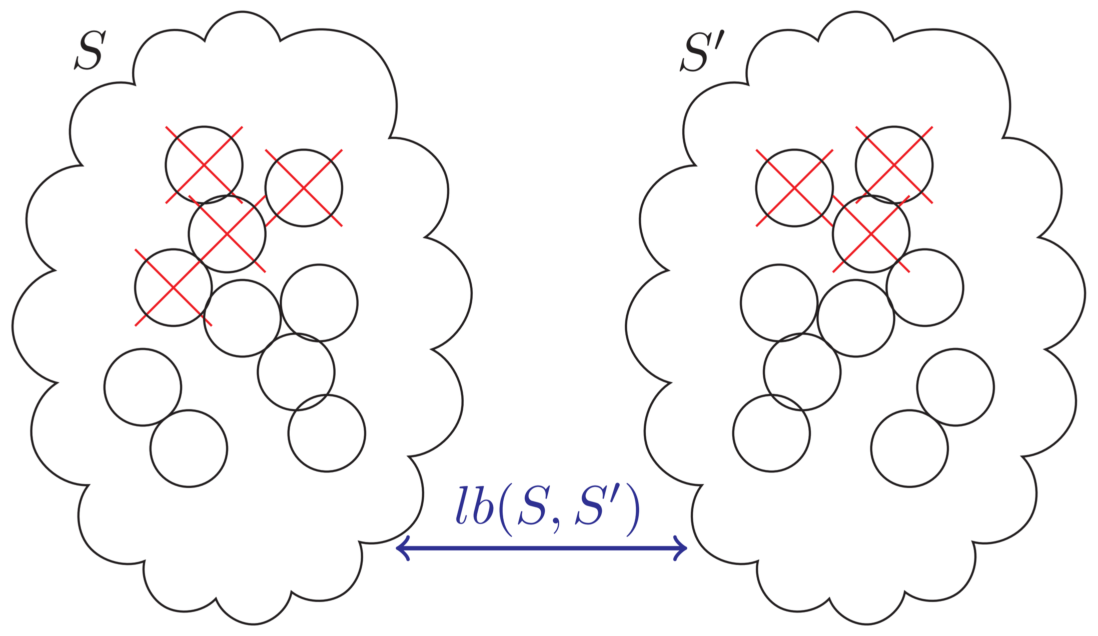
Heuristic h_C[2]

- $h_C(n, n')$ is the maximum among:
  - ϵ
  - $h_f(n) - h_f(n')$
  - $h_b(n') - h_b(n)$
- $lb(n, n') = g_f(n) + g_b(n') + h_C$



Individual Bounds[1]

- Individual bounds between sets of states
  - g bound: $gMin_x(S) + gMin_{\bar{x}}(S') + \epsilon$
  - f bounds: $fMin_x(S)$
  - KK bounds: $fMin_x(S) + dMin_{\bar{x}}(S')$
  - b bound: $(bMin_x(S) + bMin_{\bar{x}}(S'))/2$

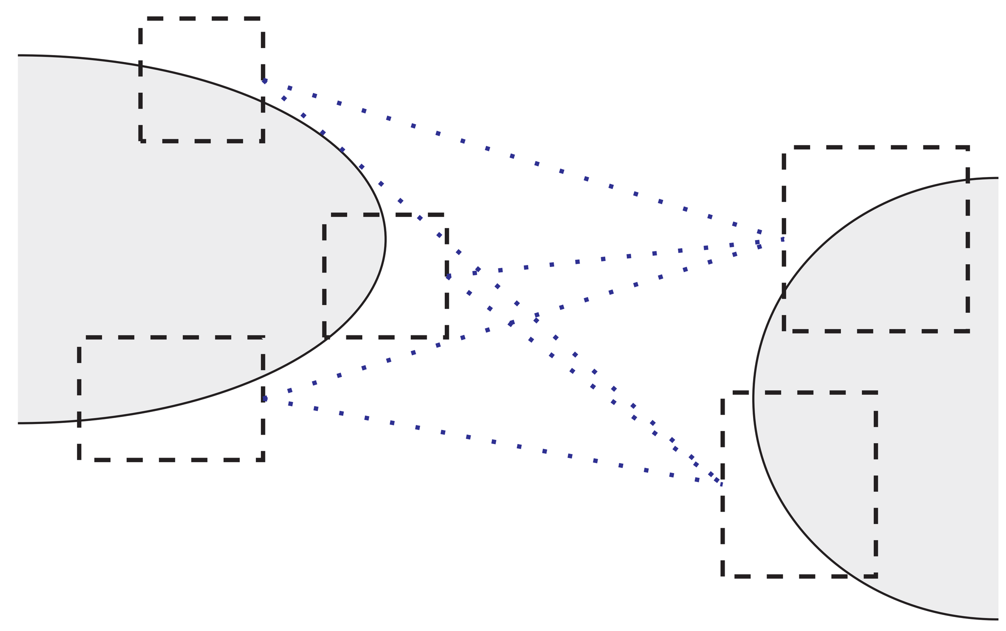


Theory of the I_{CON}/I_{CON} case

- Using individual bounds, $lb(n, n')$ is the maximum among:
  - $g_f(n) + g_b(n') + \epsilon$
  - $f_f(n) + d_b(n')$
  - $f_b(n') + d_f(n)$
- Individual bounds generalize over h_C
  - Even in undirected graphs! New bounds rf and rb defined
- All concepts from the I_{ADD}/I_{CON} case apply
  - The *must-expand graph* is not contiguous nor restrained

Bucket-to-Bucket Algorithms

- $lb(n, n')$ depends on node values and not the states!
  - Group nodes in buckets with the same node values
  - Compute the lower bound $lb(B, B')$ between buckets



- BTB (small): Choose smallest bucket
- BTB (conn): Choose most connected bucket
  - Prevent expansions in the other direction
- BTB (NBS): Expand nodes from both buckets until one is empty
  - Near-optimal algorithm for the the I_{CON}/I_{CON} case!
- BTB (DVC): Expand all edges with minimum lb before recomputing
  - Saves Bucket-to-Bucket computations, but it's less informed

Algorithm	GAP-1	GAP-2		GAP-3		GAP-4		GAP-5		GAP-6	
	$< C^*$	$< C^*$	n/s	$< C^*$	n/s	$< C^*$	n/s	$< C^*$	n/s	$< C^*$	n/s
BAE*	620	13420	168k	135k	85k	476k	76k	1087k	72k	1743k	70k
DBBS$_b$	325	7075	150k	64k	68k	248k	60k	574k	57k	901k	56k
BTB (NBS)	530	10608	130k	81k	64k	269k	57k	520k	54k	748k	51k
BTB (small)	439	9311	136k	70k	68k	229k	58k	441k	55k	654k	57k
BTB (conn)	317	6725	145k	59k	71k	208k	57k	399k	56k	576k	56k
BTB (DVC)	498	11433	151k	88k	71k	257k	57k	462k	56k	667k	57k

- Results:
  - State of the art in the Pancake Puzzle
  - Bucket-to-Bucket analogous to the *must-expand* concepts
  - Degenerates into Front-to-Front in some domains
  - Front-to-End algorithms close to their theoretical limit
    - BTB (DVC) alleviates this, potential for efficient and more informed versions
  - Front-to-Front $\longleftrightarrow$ Bucket-to-Bucket $\longleftrightarrow$ Front-to-End

References

[1] Vidal Alcázar, Patricia J. Riddle, and Mike Barley. A unifying view on individual bounds and heuristic inaccuracies in bidirectional search. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[2] Eshed Shaham, Ariel Felner, Nathan R. Sturtevant, and Jeffrey S. Rosenschein. Minimizing node expansions in bidirectional search with consistent heuristics. In *Proceedings of the Eleventh International Symposium on Combinatorial Search, SoCS*, pages 81–98, 2018.