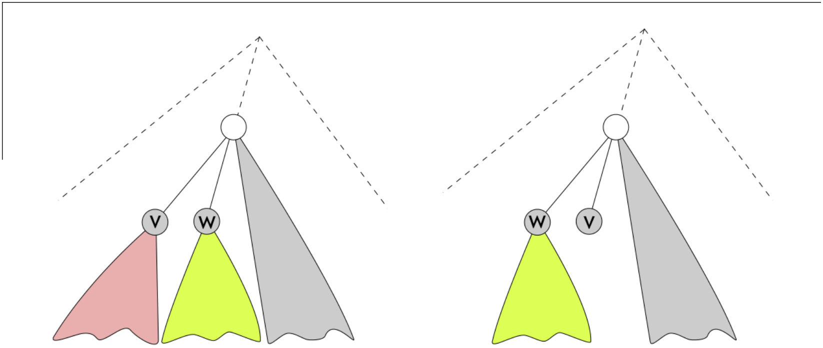
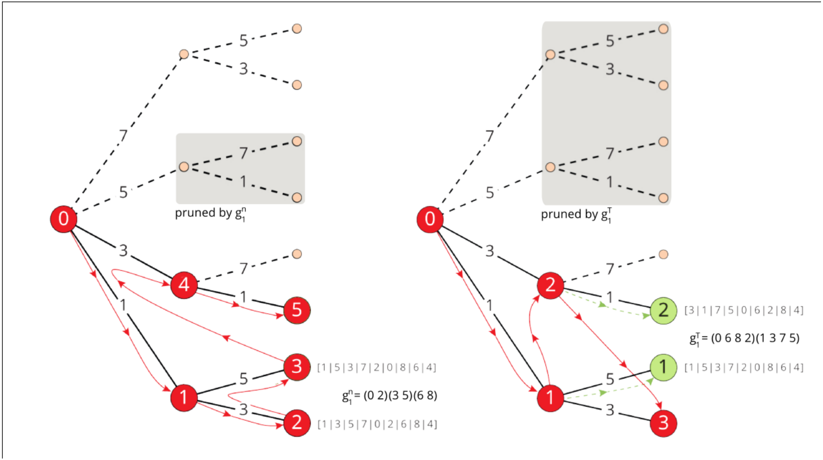
Traces

Traces Motivation

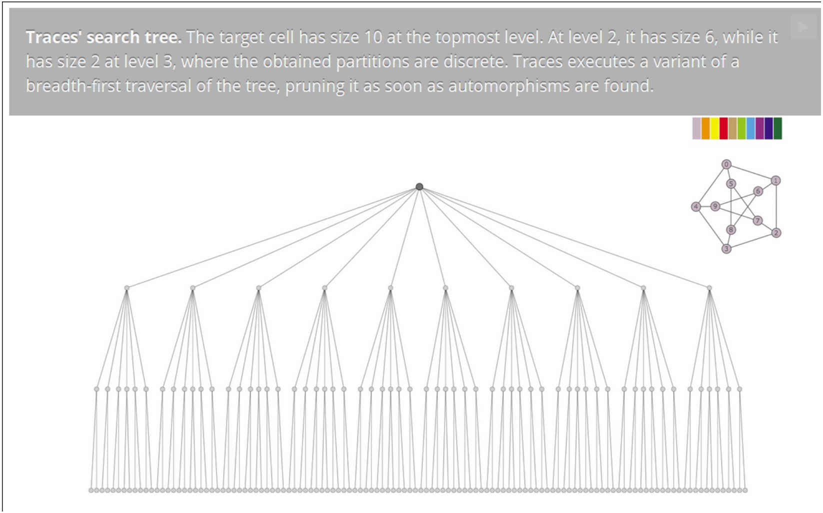


Assume that the canonical form is not associated to a leaf of the tree rooted at v. If v comes before w, then the whole subtree rooted at v is visited before it is realized that its construction could have been avoided.

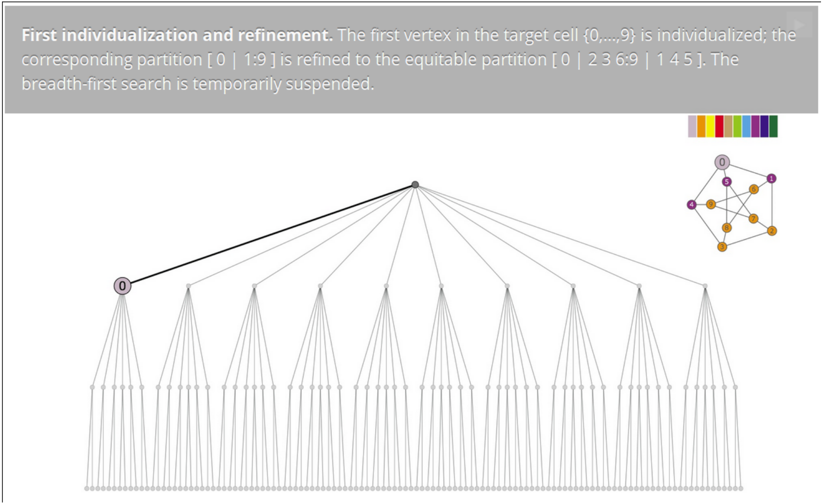
Search Tree Nauty (left) / Traces (right)



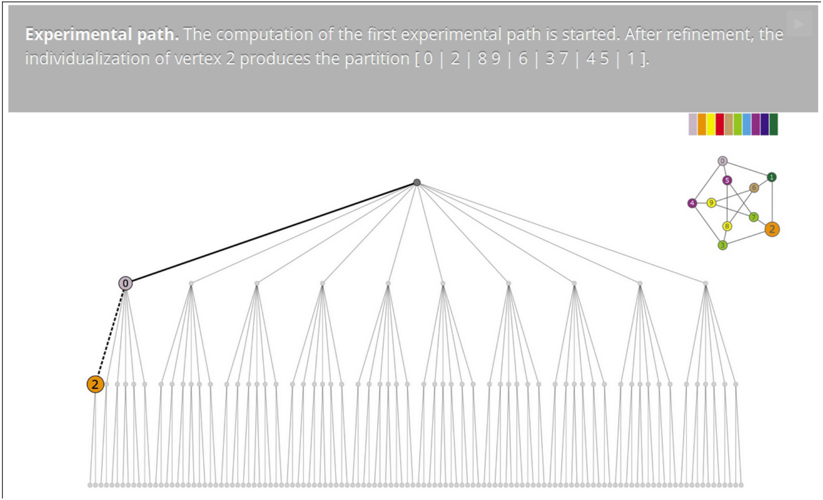
Traces


Traces' search tree. The target cell has size 10 at the topmost level. At level 2, it has size 6, while it has size 2 at level 3, where the obtained partitions are discrete. Traces executes a variant of a breadth-first traversal of the tree, pruning it as soon as automorphisms are found.

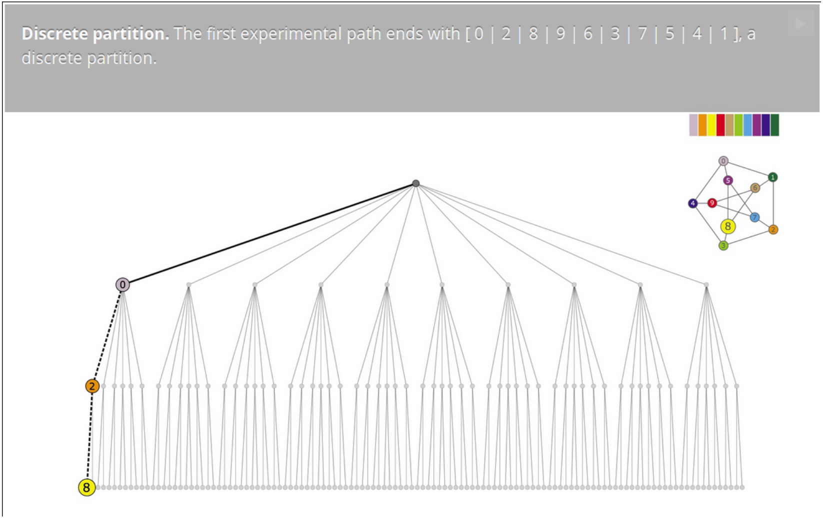
Traces


First individualization and refinement. The first vertex in the target cell {0,...,9} is individualized; the corresponding partition [0 | 1:9] is refined to the equitable partition [0 | 2 3 6:9 | 1 4 5]. The breadth-first search is temporarily suspended.
0
0
1
2
3
4
5
6
7
8
9

Traces

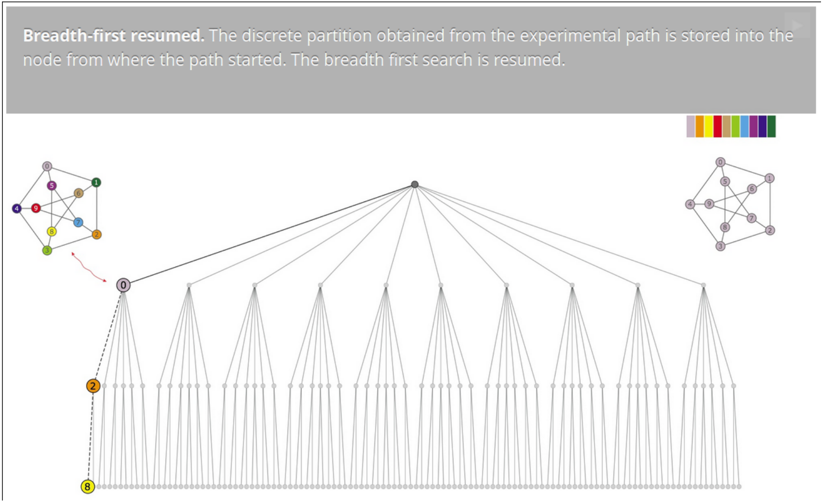

Experimental path. The computation of the first experimental path is started. After refinement, the individualization of vertex 2 produces the partition [0 | 2 | 8 9 | 6 | 3 7 | 4 5 | 1].
0
2

Traces

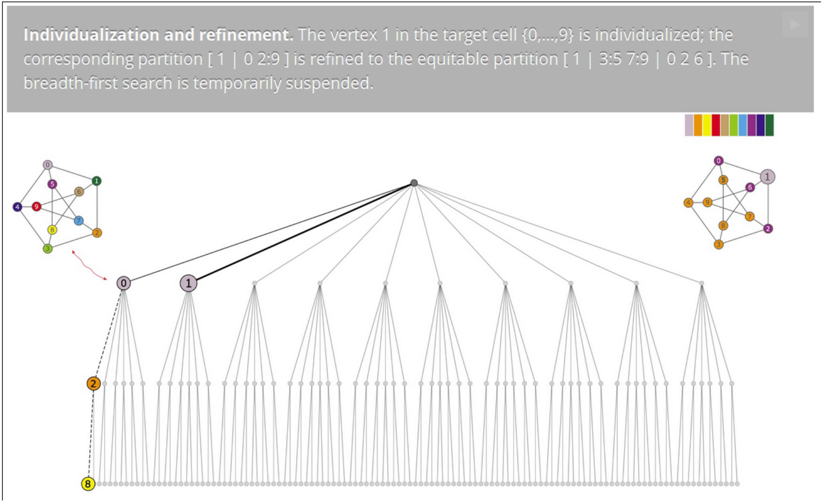


Discrete partition. The first experimental path ends with [0 | 2 | 8 | 9 | 6 | 3 | 7 | 5 | 4 | 1], a discrete partition.
0
2
8

Traces

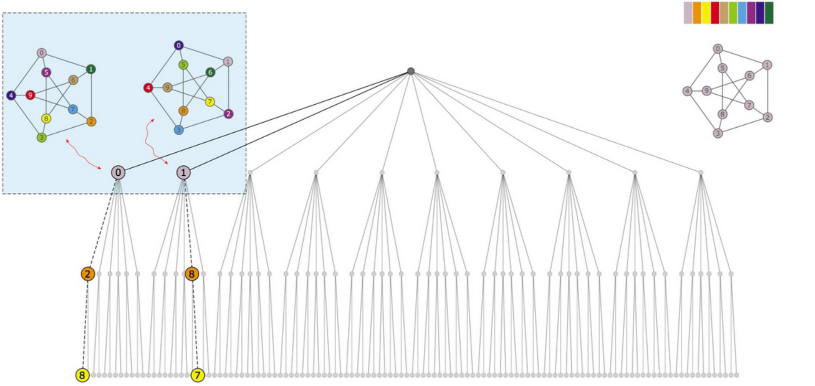

Breadth-first resumed. The discrete partition obtained from the experimental path is stored into the node from where the path started. The breadth first search is resumed.

Traces

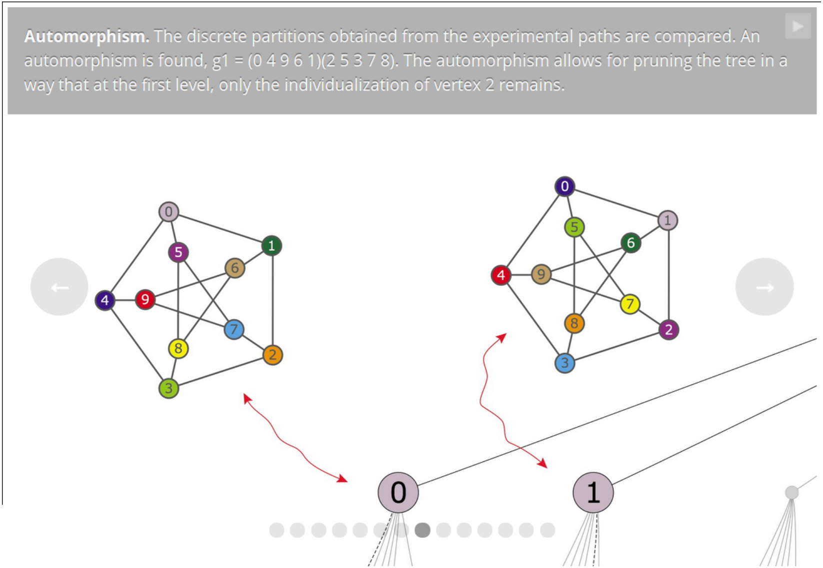

Individualization and refinement. The vertex 1 in the target cell {0,...,9} is individualized; the corresponding partition [1 | 0 2:9] is refined to the equitable partition [1 | 3:5 7:9 | 0 2 6]. The breadth-first search is temporarily suspended.

Traces

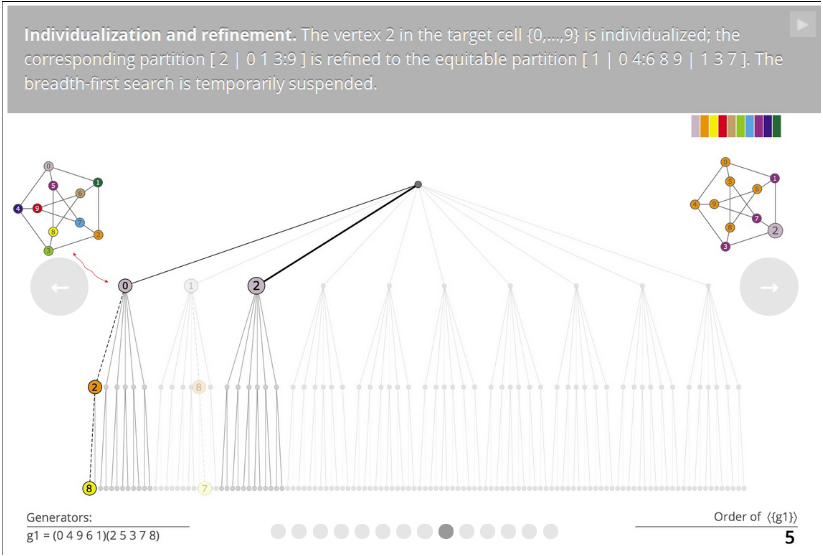
Experimental path. The experimental path from the current node is computed. It finally produces the discrete partition [1 | 8 | 7 | 4 | 9 | 5 | 3 | 2 | 0 | 6], which is stored into the current node itself.



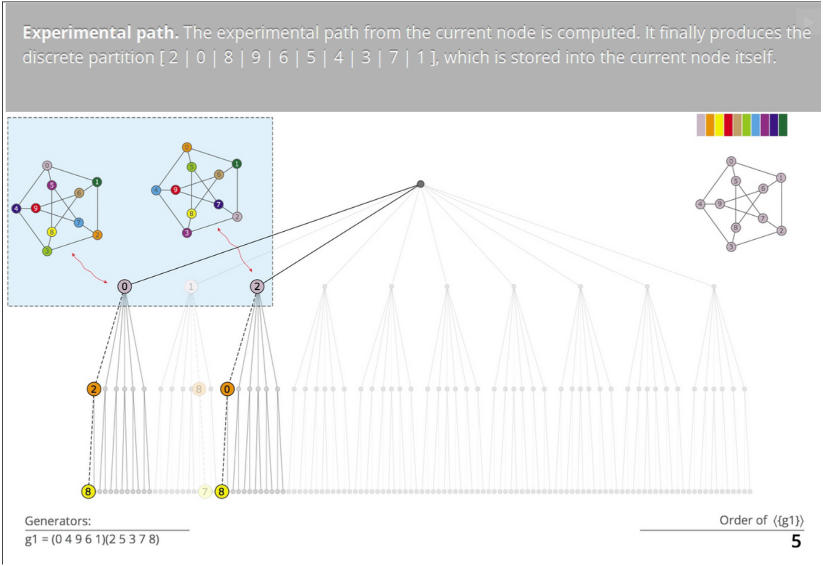
Traces


Automorphism. The discrete partitions obtained from the experimental paths are compared. An automorphism is found, g1 = (0 4 9 6 1)(2 5 3 7 8). The automorphism allows for pruning the tree in a way that at the first level, only the individualization of vertex 2 remains.
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
8
9
0
1

Traces

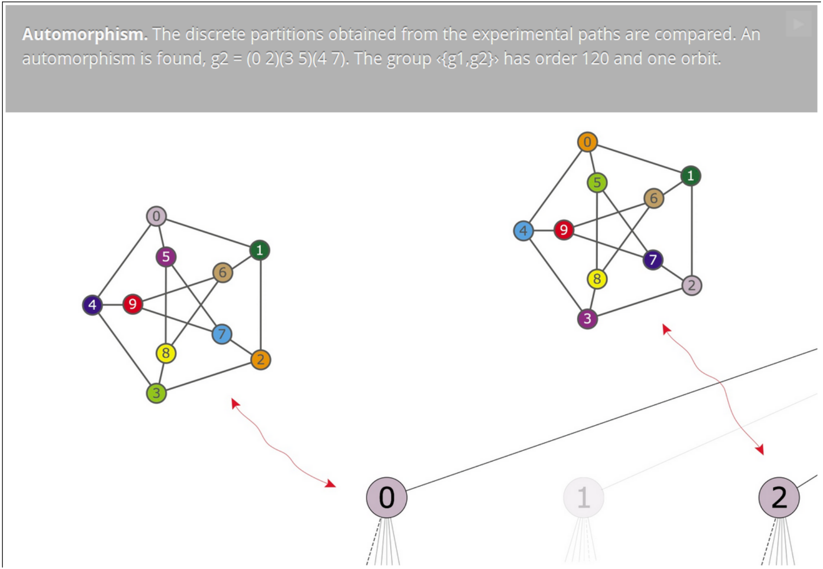

Individualization and refinement. The vertex 2 in the target cell {0,...,9} is individualized; the corresponding partition [2 | 0 1 3:9] is refined to the equitable partition [1 | 0 4:6 8 9 | 1 3 7]. The breadth-first search is temporarily suspended.
Generators:
g1 = (0 4 9 6 1)(2 5 3 7 8)
Order of ⟨{g1}⟩
5

Traces

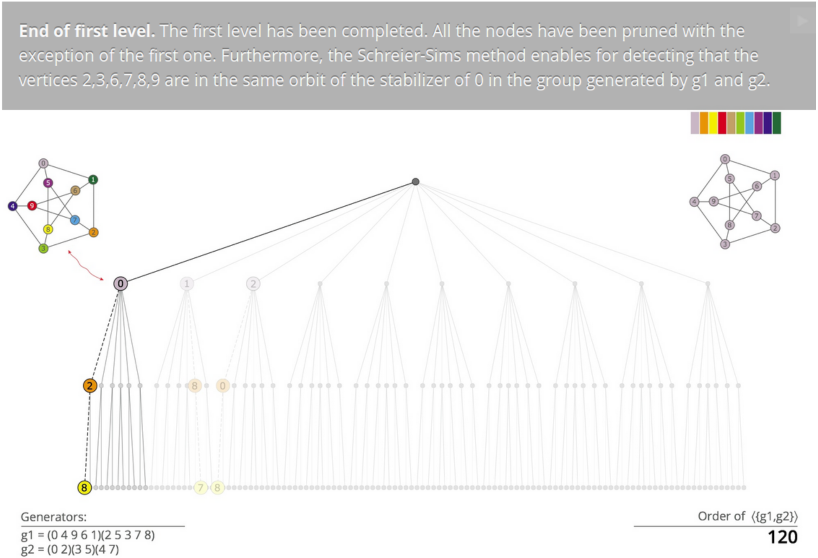


Experimental path. The experimental path from the current node is computed. It finally produces the discrete partition [2 | 0 | 8 | 9 | 6 | 5 | 4 | 3 | 7 | 1], which is stored into the current node itself.
Generators:
g1 = (0 4 9 6 1)(2 5 3 7 8)
Order of ⟨{g1}⟩
5

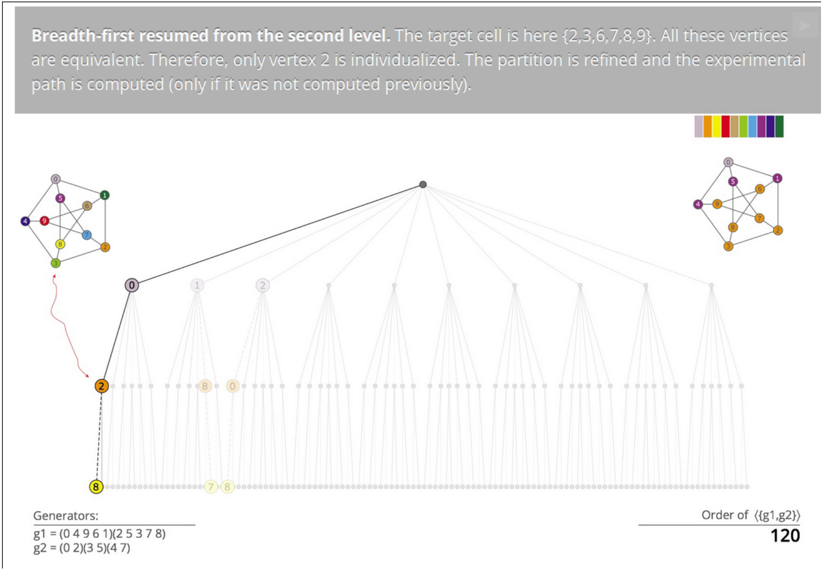
Traces


Automorphism. The discrete partitions obtained from the experimental paths are compared. An automorphism is found, g2 = (0 2)(3 5)(4 7). The group ‹{g1,g2}› has order 120 and one orbit.
0
1
2

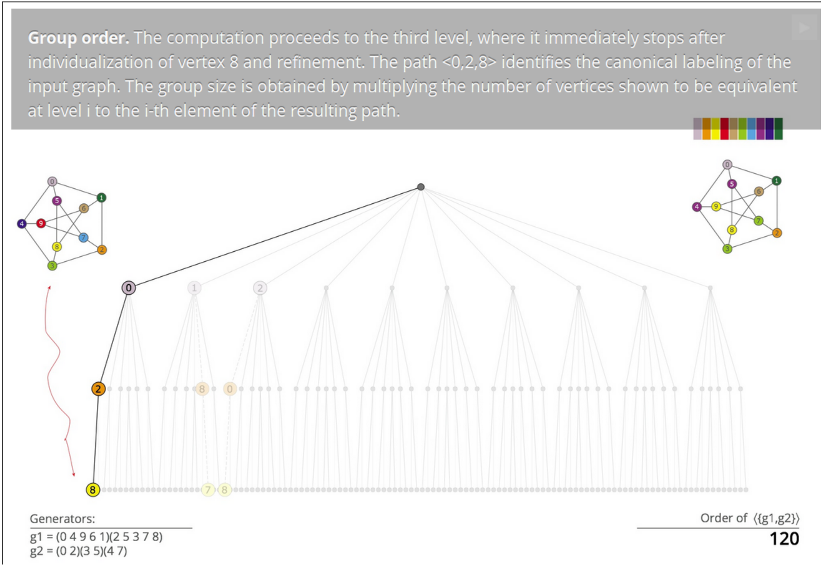
Traces


End of first level. The first level has been completed. All the nodes have been pruned with the exception of the first one. Furthermore, the Schreier-Sims method enables for detecting that the vertices 2,3,6,7,8,9 are in the same orbit of the stabilizer of 0 in the group generated by g1 and g2.
Generators:
g1 = (0 4 9 6 1)(2 5 3 7 8)
g2 = (0 2)(3 5)(4 7)
Order of ⟨{g1,g2}⟩
120

Traces


Breadth-first resumed from the second level. The target cell is here {2,3,6,7,8,9}. All these vertices are equivalent. Therefore, only vertex 2 is individualized. The partition is refined and the experimental path is computed (only if it was not computed previously).
Generators:
g1 = (0 4 9 6 1)(2 5 3 7 8)
g2 = (0 2)(3 5)(4 7)
Order of ⟨{g1,g2}⟩
120

Traces

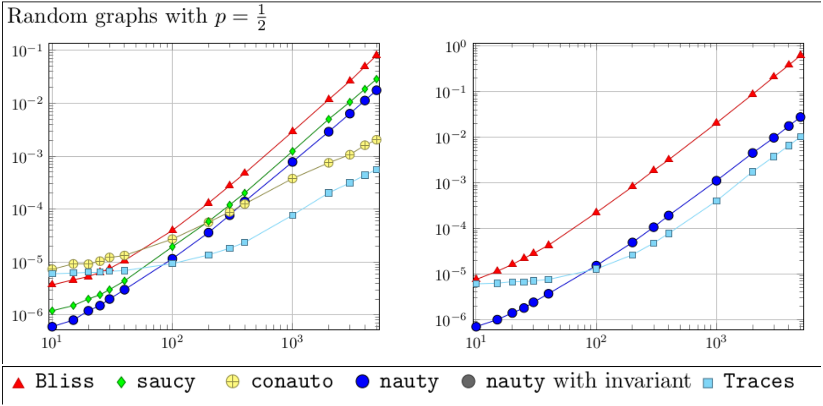


Group order. The computation proceeds to the third level, where it immediately stops after individualization of vertex 8 and refinement. The path <0,2,8> identifies the canonical labeling of the input graph. The group size is obtained by multiplying the number of vertices shown to be equivalent at level i to the i-th element of the resulting path.
Generators:
g1 = (0 4 9 6 1)(2 5 3 7 8)
g2 = (0 2)(3 5)(4 7)
Order of ⟨{g1,g2}⟩
120

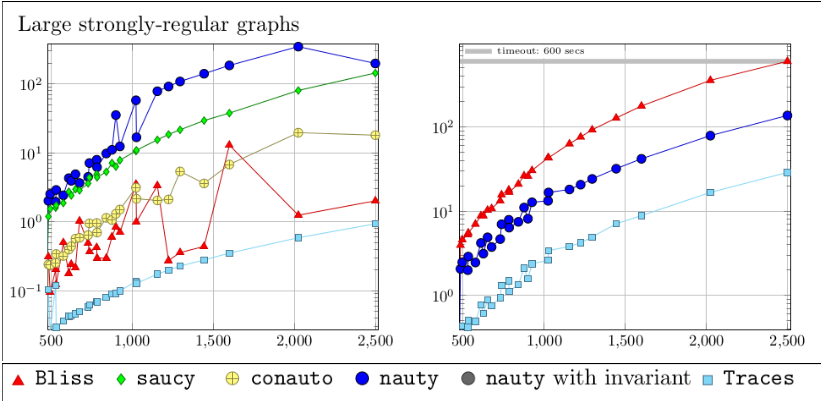
Performance Comparison

Runtime in Seconds

left: automorphism group right: canonical label



Performance Comparison



Performance Comparison