Maze Traversals

In this short note we will describe how we can solve every "maze", without using any graph properties.

A *maze* is an undirected connected graph, with two special nodes: the unique *entrance node* and the unique *exit node*. Furthermore, for each node its neighbours will be sorted in a circular way. This means that if a path reaches a node, we can enumerate in a clockwise way the edges that leave from this node, starting from the edge we just arrived from (which is referred to as the 0^{th} edge; in the entrance node we just fix some order). This can be achieved by ordering the edges through their target number, in the case of a graph where the nodes are labelled with unique integers. Each node also has a counter for its degree. Using modular arithmetic, we can uniquely identify the k^{th} edge ($k \in \mathbf{Z}$) when we arrive at some node.
We consider a maze to be solved, if we can find a path, starting at the entrance node, that visits the exit node. In that case, the path ends at this exit node, neglecting possible further edges.
An ordered finite series S of integers (in $\mathbf{Z}$) can be viewed as a path in *any* maze in the following way. Suppose we have $S = (s_1, s_2, \ldots, s_k)$ with $s_i \in \mathbf{Z}$ for $i = 1, 2, \ldots, k$, and $s_1 \neq 0$. Then we start our path in the entrance node of the given graph, and take the s_1^{th} edge. For the following nodes we proceed in a similar way. If for some $i > 0$ we have $s_i = 0$, the path proceeds by using the edge just traversed in backward direction — which corresponds with the 0^{th} edge. In this way we traverse any maze, perhaps even solving it. If some s-value happens to be 0 when we are in the entrance node, we just stay there.
We now note that this construction allows for paths to return to the entrance node, in any graph. As an example, consider $S = (2, 3, 7, 0, -7, -3)$. Suppose for the moment that all nodes have sufficiently large degrees, then the -7 causes the path to use the same edge (in backward direction) as it did on the way out. If the degrees are too small, this same phenomenon holds! If a sequence ends, the path also halts.
Next we observe that we can implement a sort of iterative deepening depth first traversal. Indeed, fix a depth $d \geq 1$ and a width $w \geq 1$. Now concatenate — in some order — all sequences $(s_1, s_2, \ldots, s_d, 0, -s_d, \ldots, -s_2)$ where $0 < s_i \leq w$ for all i with $1 \leq i \leq d$, giving a sequence $S_{d,w}$. Concatenate, for a fixed d, all $S_{d,w}$ with $w \leq d$, giving S_d. Finally, build an infinite sequence $\mathcal{S}$ by concatenating all sequences S_d for $d = 1, 2, \ldots$ This sequence then solves all possible mazes. Note that we can even omit prefixes (consisting of complete $S_{d,w}$'s). The sequence can be made finite by imposing an appropriate upper bound on the d-values.
The sequence $\mathcal{S}$ begins with (use $S_1 = (1, 0)$):

$$(1, 0, 1, 1, 0, -1, \ldots)$$

Walter Kosters, August 19, 2015, `w.a.kosters@liacs.leidenuniv.nl`