Name	Description
Tanner Greedy	Constructs the graph using PEG*
Tanner Sorted Asc	Tanner Greedy, but sorts the degrees of nodes ascending before applying the algorithm, in order of input.
Tanner Sorted Dsc	Tanner Greedy, but sorts the degrees of nodes descending before applying the algorithm
Greedy	Applies a first open connection made rule, but on the bilinear graph, not a tree like graph, in order of input.
Greedy Sort Asc	Greedy, but sorts the degrees of nodes ascending before applying the algorithm
Greedy Sort Dsc	Greedy, but sorts the degrees of nodes descending before applying the algorithm
Greedy Factors Asc Nodes Dsc	Greedy, but sorts the degrees of variable nodes ascending and descending for check nodes before applying the algorithm
Greedy Factors Asc Nodes Dsc	Greedy, but sorts the degrees of variable nodes descending and ascending for check nodes before applying the algorithm
Sort Asc on Step	Greedy, but apply an ascending sort after each bond is made to all degrees.
Sort Dsc on Step	Greedy, but apply an descending sort after each bond is made to all degrees.
Flip Sort on step Asc	Greedy, but apply a sort after each bond is made to all degrees, switching the type of sorting each time and starting with ascending.
Flip Sort on step Dsc	Greedy, but apply a sort after each bond is made to all degrees, switching the type of sorting each time and starting with descending.
Half Sort on step	Greedy, but apply an ascending sort after each bond is made to the first half of the degrees.
Random	All bonds are randomly made
*	PEG here uses the described algorithm from the research paper, but adapted to properly handle the intended input

Table 1: Algorithms and their brief descriptions

Name	Description
bond	Receives two indices for a variable and a check node, then draws a line between them, as well as adding a highlight effect on hover a node

clean	Empties the graph container of elements and resets all values to default (for new input)
display_matrix	Displays the matrices as text in their respective containers
draw_graph	Draws a given matrix as a graph (with circles, squares, lines and outlines)
draw_line	Draws a line between two given nodes
draw_outline	Draws a square outline over a module
draw_shapes	Contains the function for drawing a square (variable node) or a circle (check nodes)
export	Generates the file containing a given matrix (with or without compression) and allows the download of the file
filters_change	Event listeners for filters that impact the CSS of the tool
get_matrix	Functions for reading the input file and generation of all types of matrices
main	General flow of the code and the initialization values
shift_check	Constraints and validity check for shifting a bond
window_functions	Adjustments to the containers to fit the graph and text matrices properly as the data gets larger

Table 2: JS files for Nodes and Factors Graph

Name	Description
algorithms_non_tanner	Contains the algorithms for generating a matrix without using the Tanner Graph
algorithms_tanner	Contains the algorithms for generating a matrix using the Tanner Graph
bond	Receives two indices for a variable and a check node, then draws a line between them, as well as adding a highlight effect on hover a node
clean	Empties the graph container of elements and resets all values to default (for new input)
display_matrix	Displays the matrices as text in their respective containers
draw_line	Draws a line between two given nodes
draw_shapes	Contains the function for drawing a square (variable node) or a circle (check nodes)
export	Generates the file containing a given matrix (with or without compression) and allows the download of the file
get_file_data	Reads the input data and generated the nodes arrays
main	General flow of the code and the initialization values
window_functions	Adjustments to the containers to fit the graph and text matrices properly as the data gets larger

Table 3: JS files for PEG Generator