Test Set Report: short

October 24, 2020

Test Run Characteristics

Test Set Name:	`short`
Settings:	`default`
Mode:	`readdec`
GCG Version:	`3.1.0`
Feasibility Tolerance:	default
LP Solver:	spx2
Time Limit:	3600s
Memory Limit:	6144MB
Node Limit:	2100000000

Test Set Solving Overview

Number of Instances:	14 (14 passed, 0 failed, 0 timeout)
Total Solving Time:	19.7s
Average Solving Time:	1.3s

Report Timestamp:	24-10-2020 15-01-01
Test Run Binary ID:	gcg-3.1.0.linux.x86_64.gnu.opt.spx2.tim-vb

Contents

1 Instance Information

1.1 Instance: noswot

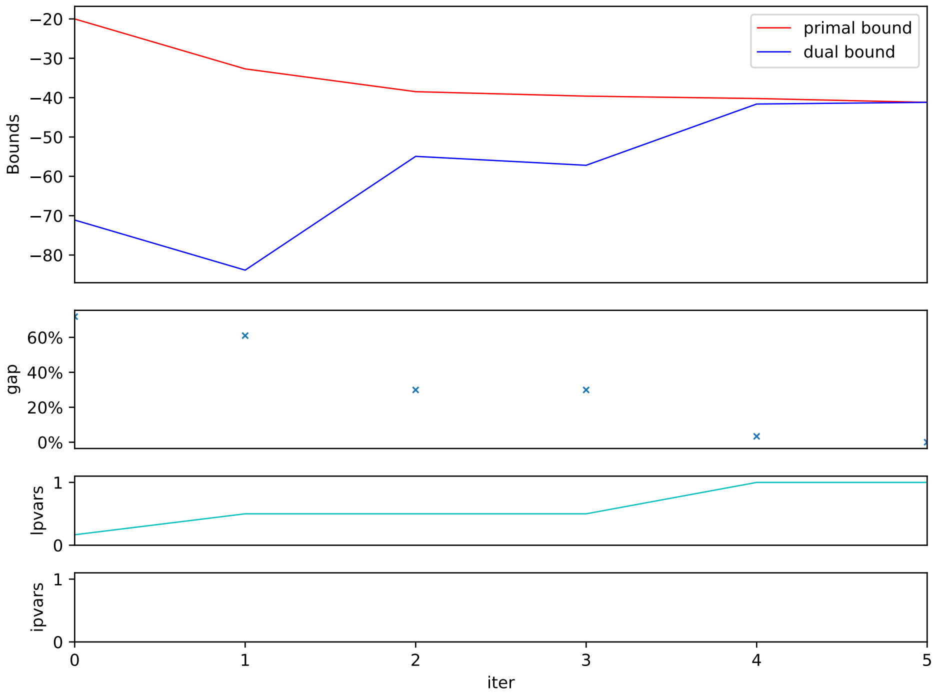


Figure 1: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/noswot.default.bounds.iter.pdf`

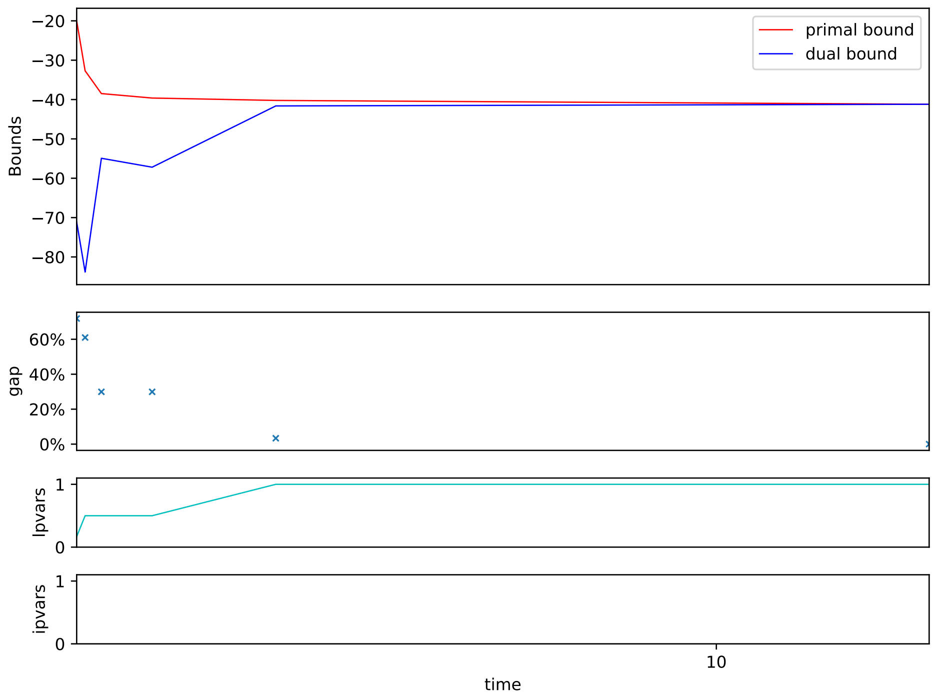


Figure 2: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/noswot.default.bounds.time.pdf`

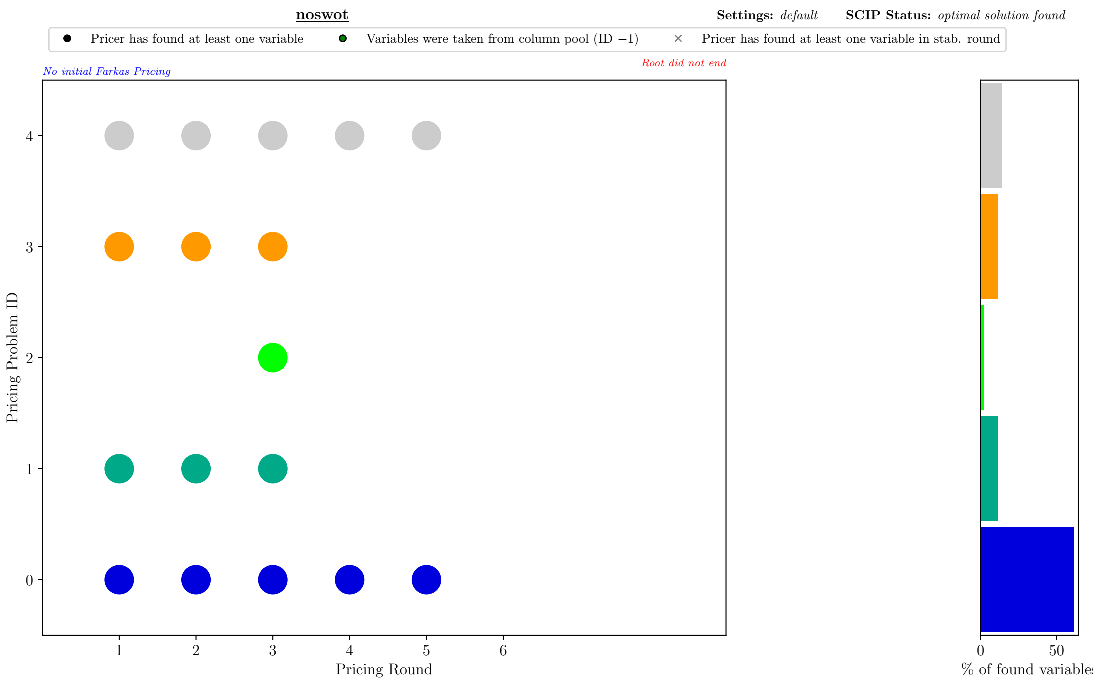


Figure 3: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/noswot.default.bubble.pdf`

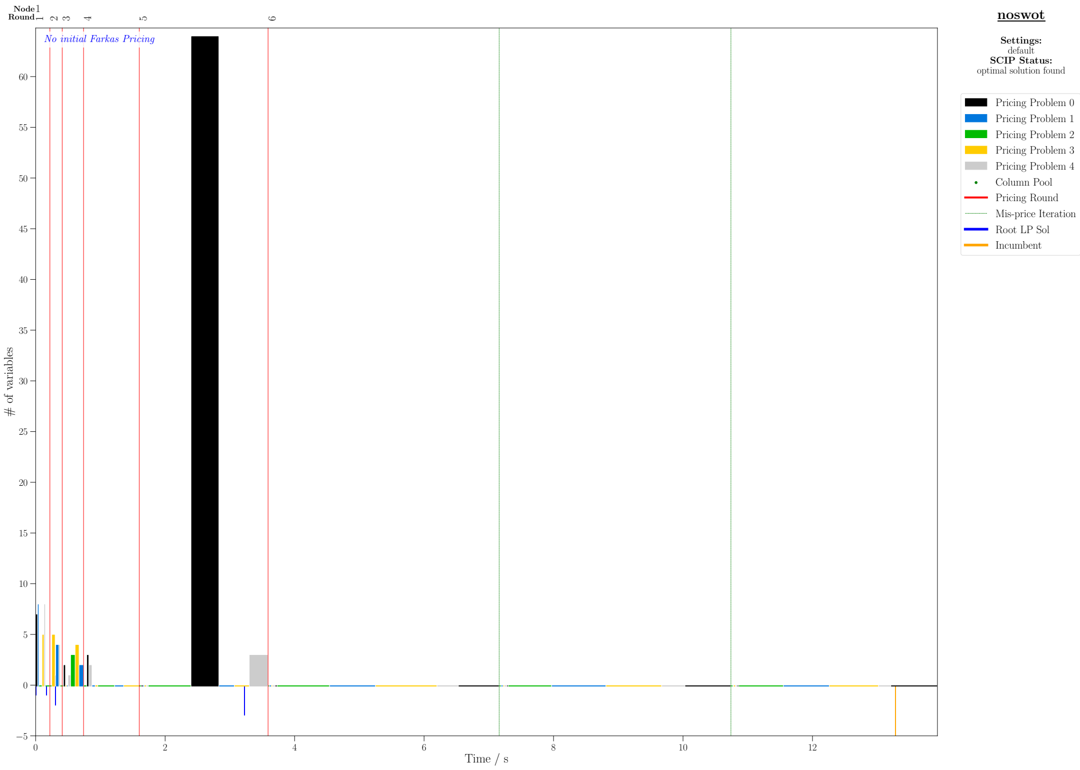


Figure 4: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/noswot.default.complete.pdf`

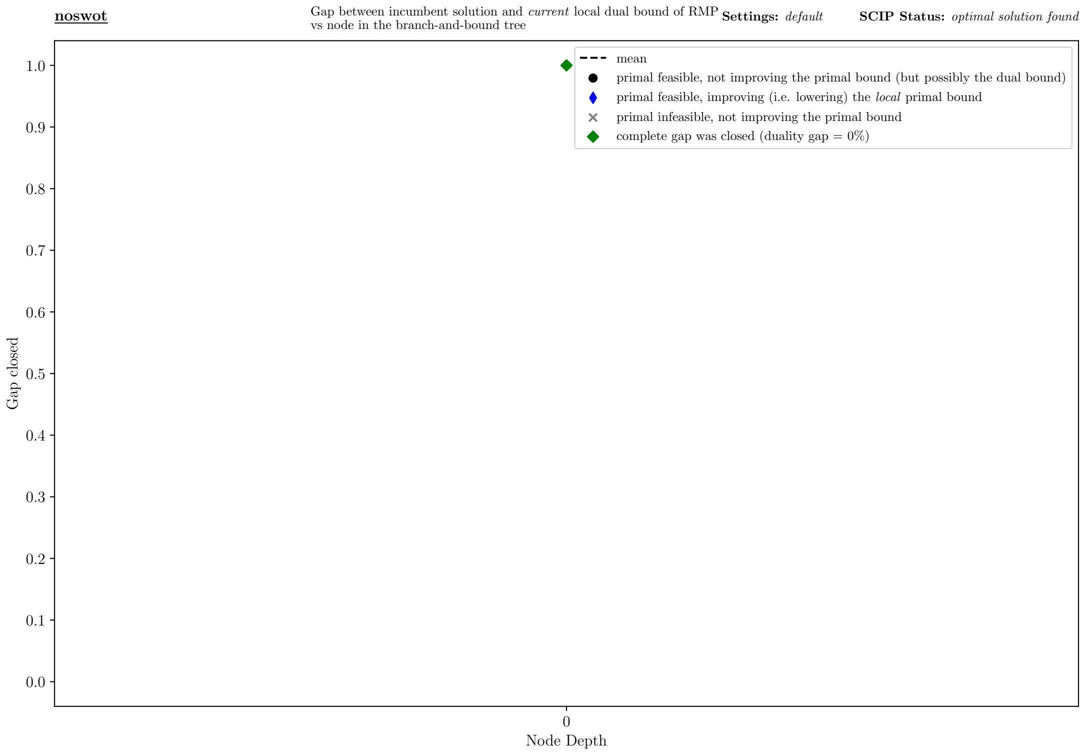


Figure 5: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/noswot.default.depth.pdf`

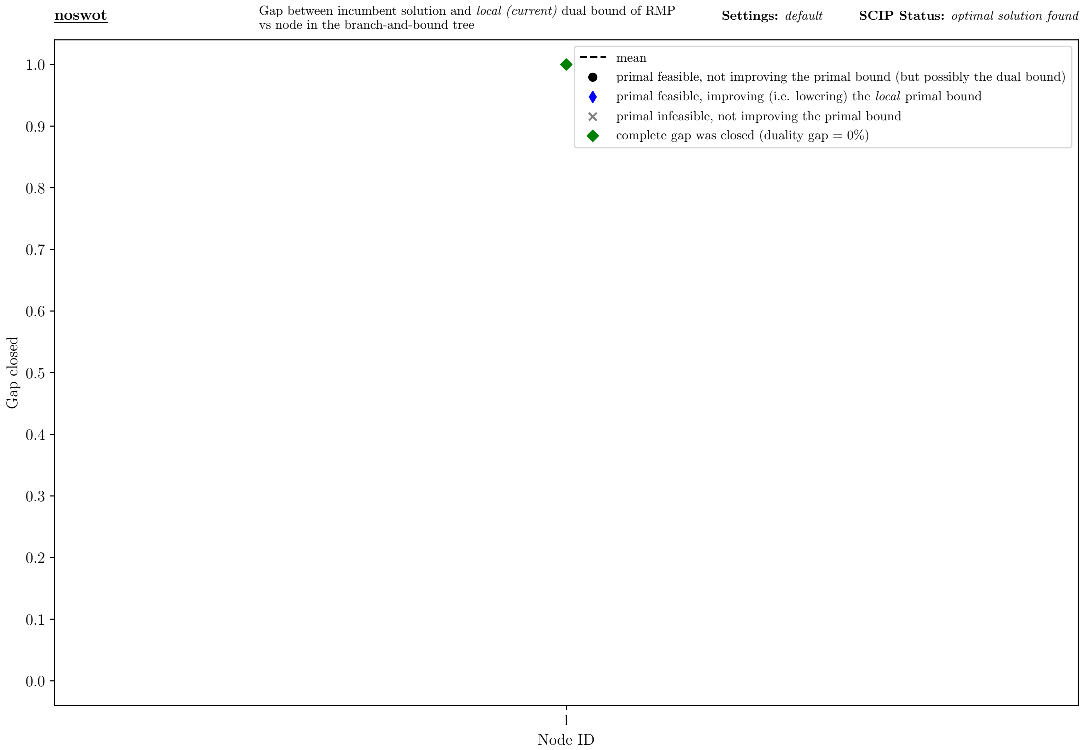


Figure 6: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/noswot.default.nodeID.pdf`

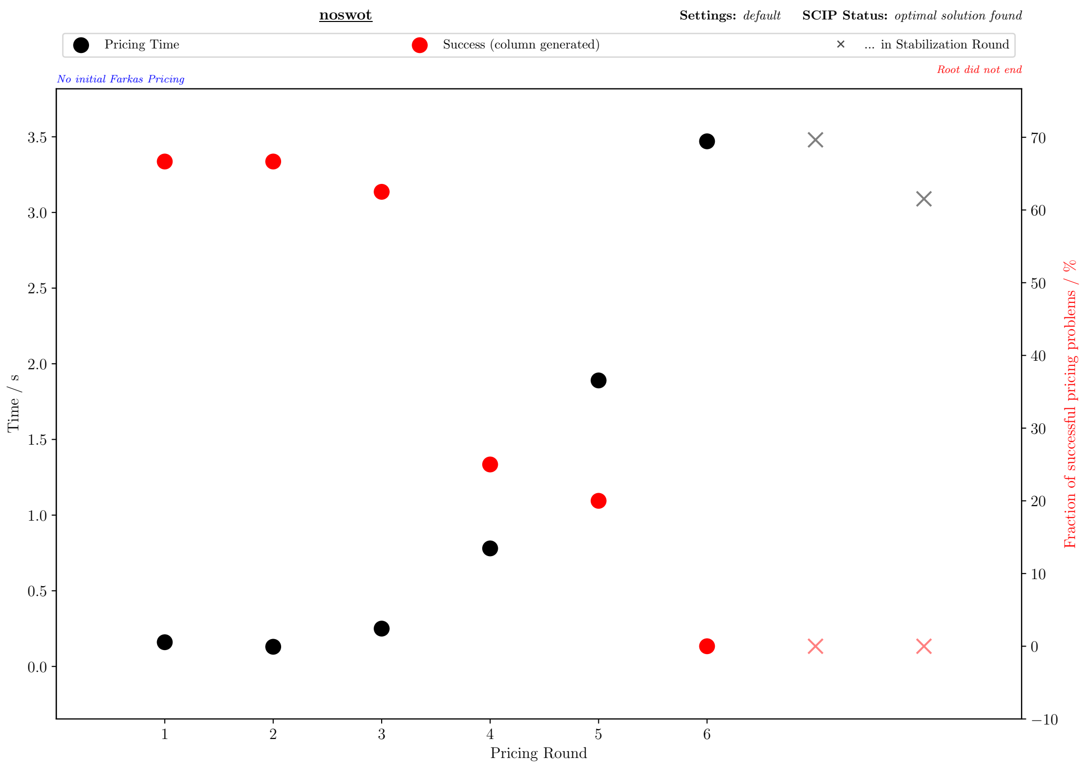


Figure 7: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/noswot.default.summary.pdf`

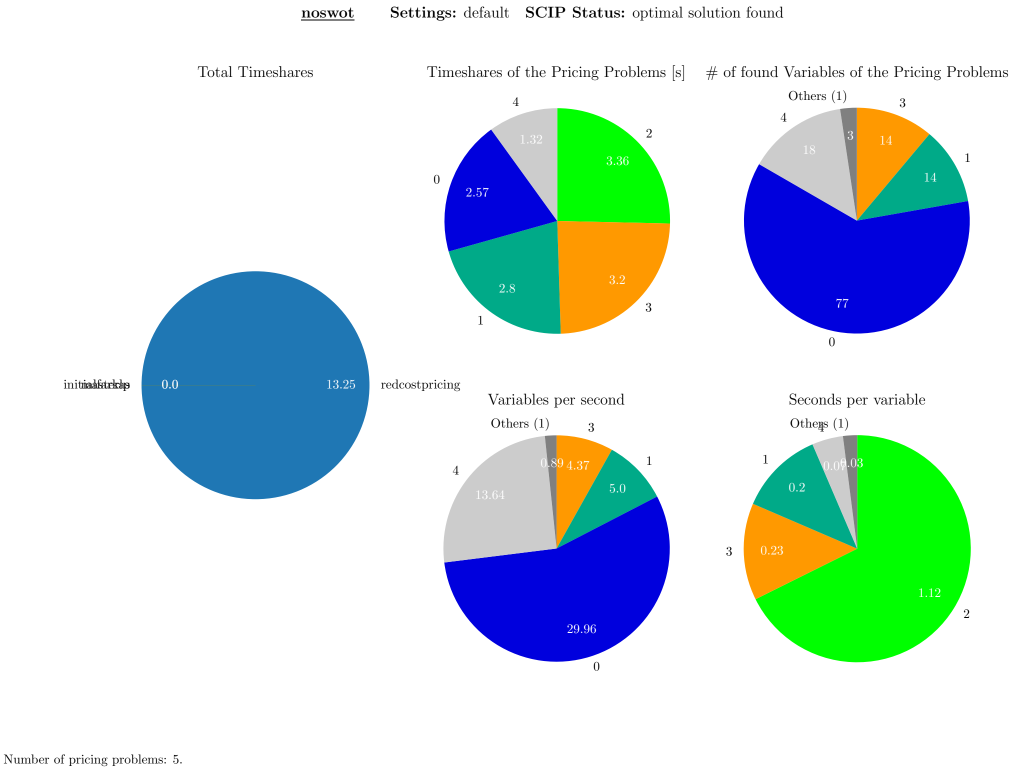
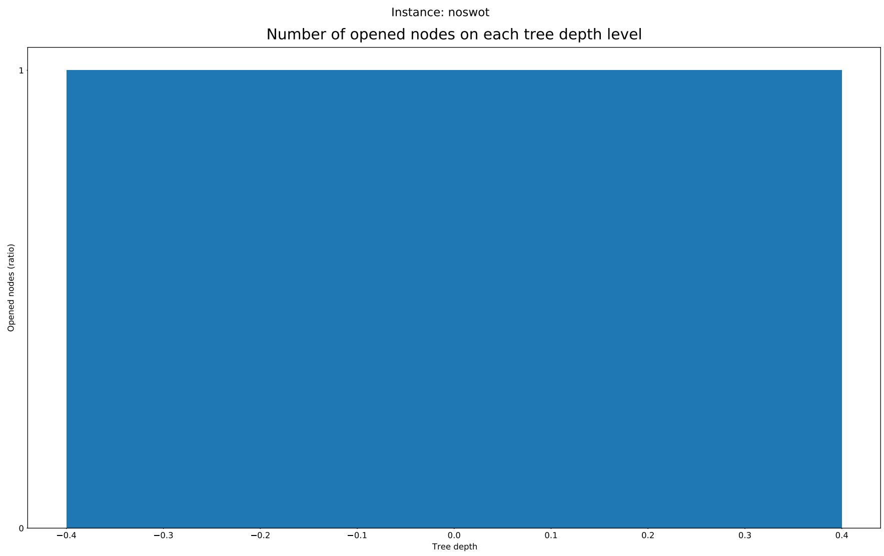
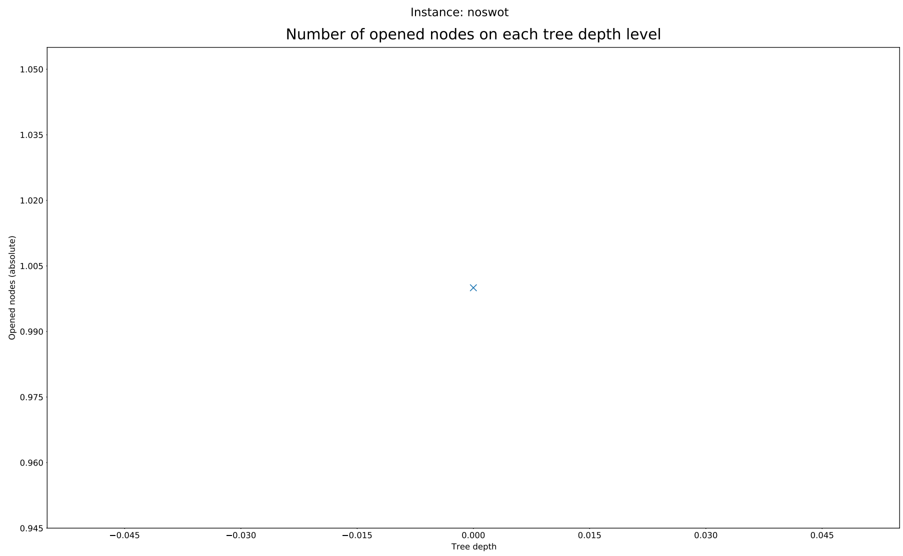


Figure 8: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/noswot.default.time.pdf`



The total number of opened nodes was 1.

Figure 9: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/noswot.default.tree.bar.pdf`



The total number of opened nodes was 1.

Figure 10: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/noswot.default.tree.plot.pdf`

1.2 Instance: N1C1W4_M.BPP

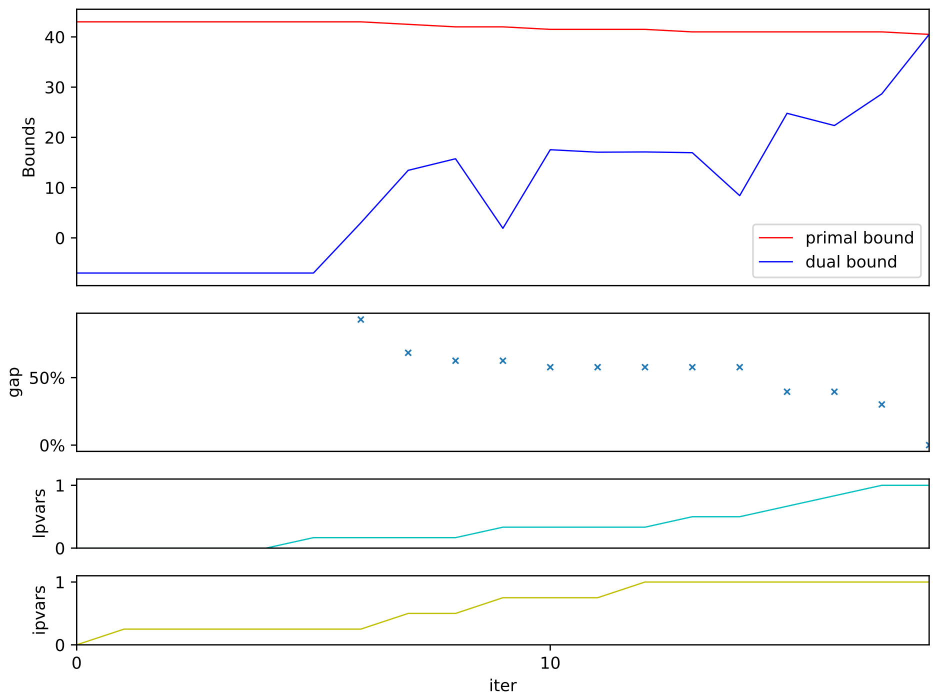


Figure 11: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/N1C1W4_M.BPP.default.bounds.iter.pdf`

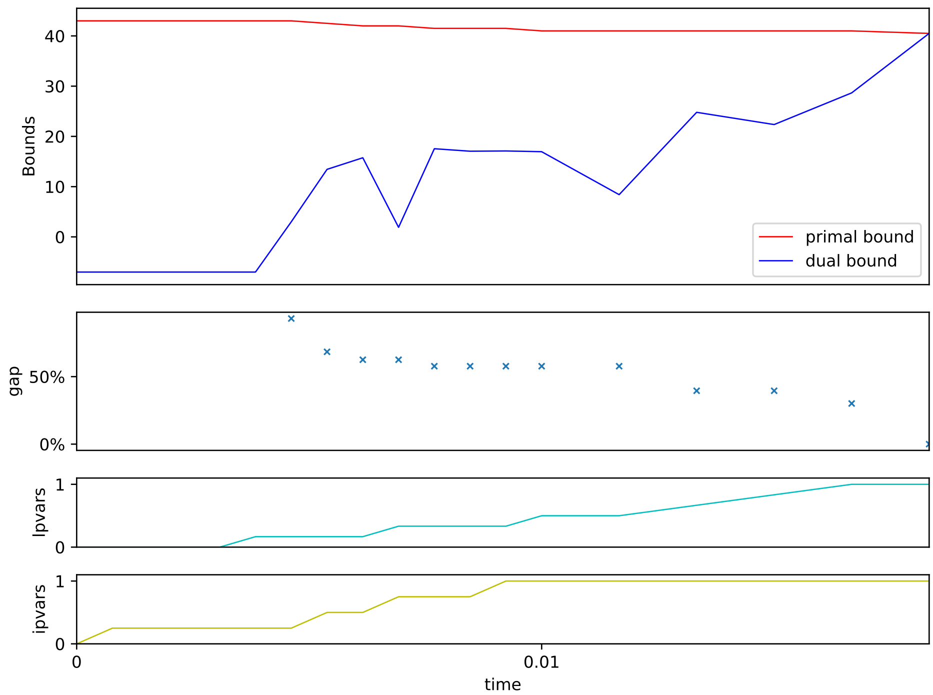


Figure 12: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds” printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/N1C1W4_M.BPP.default.bounds.time.pdf`

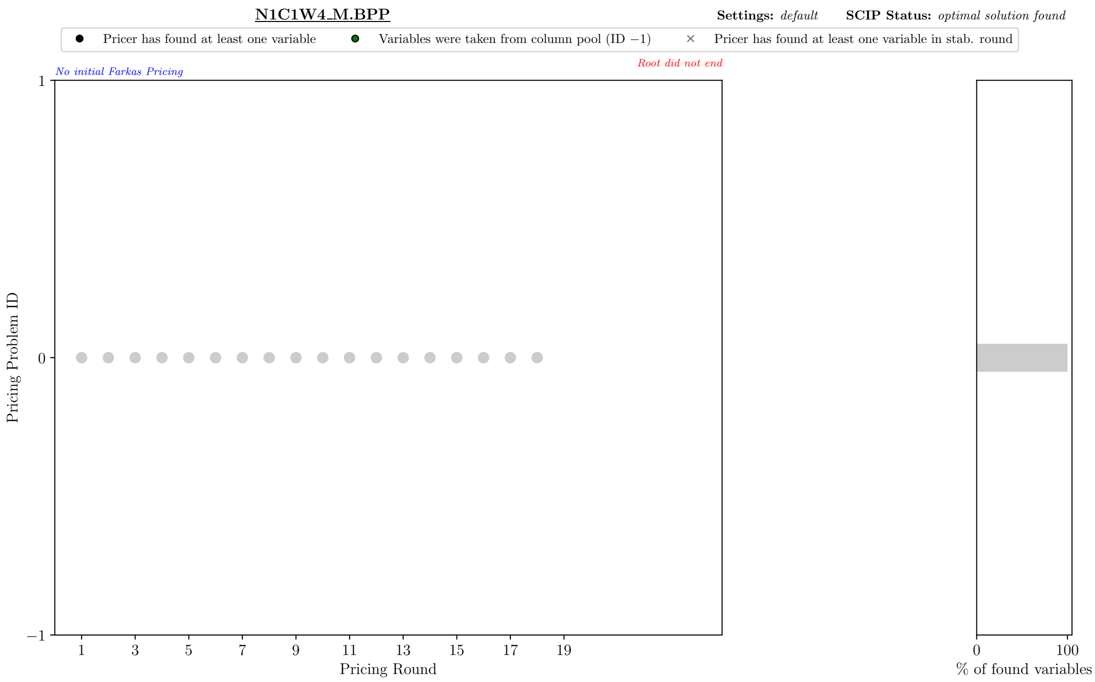


Figure 13: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.bubble.pdf`

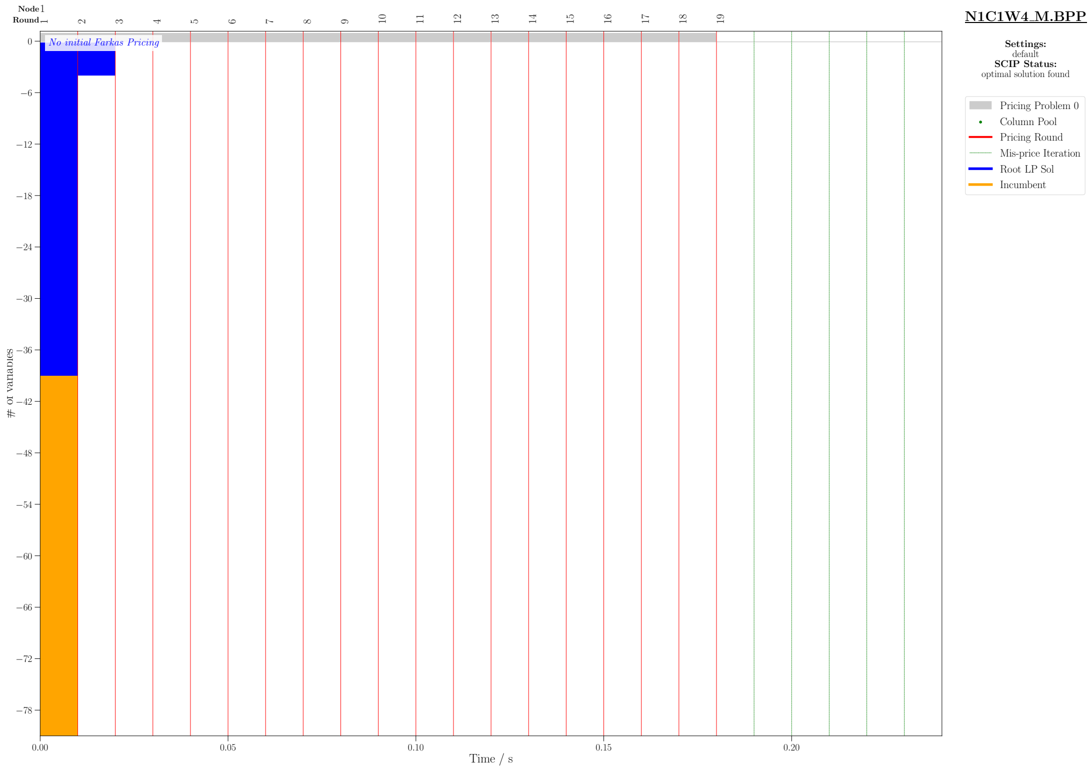


Figure 14: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.complete.pdf`

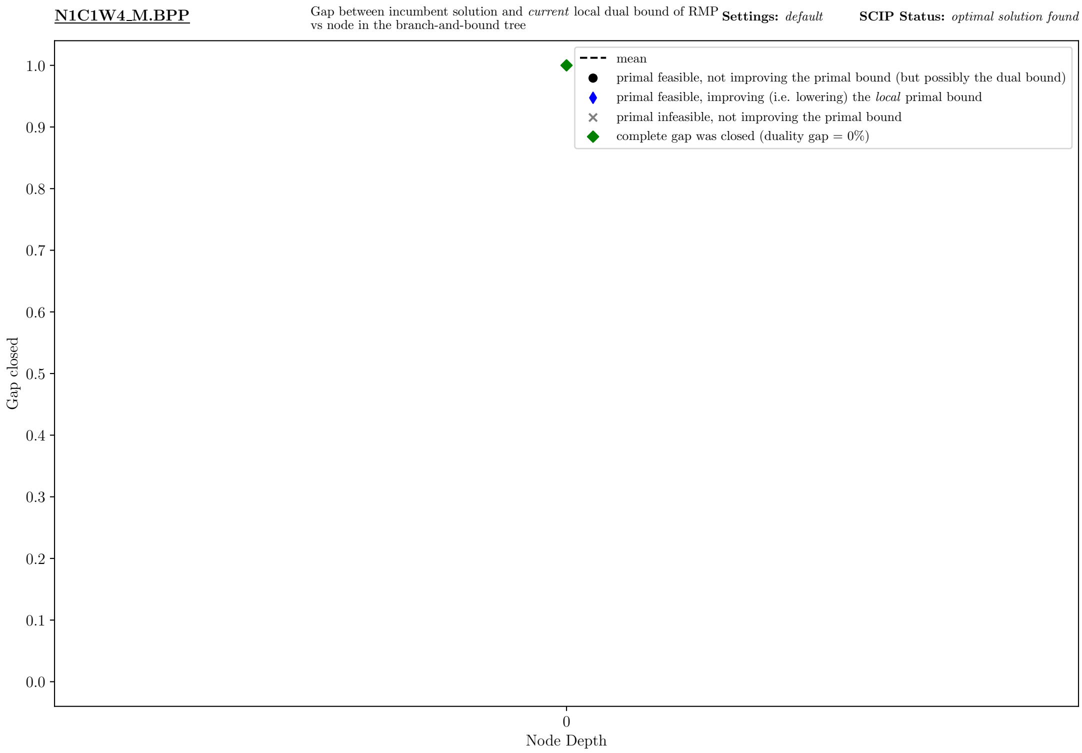


Figure 15: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.depth.pdf`

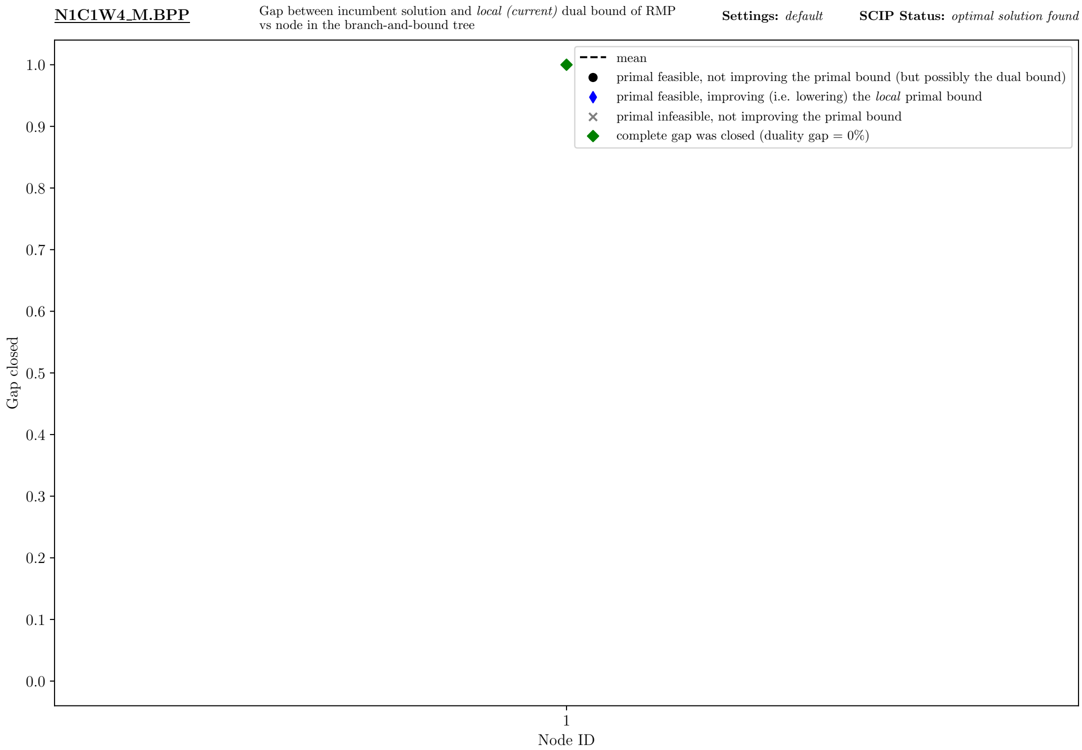


Figure 16: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.nodeID.pdf`

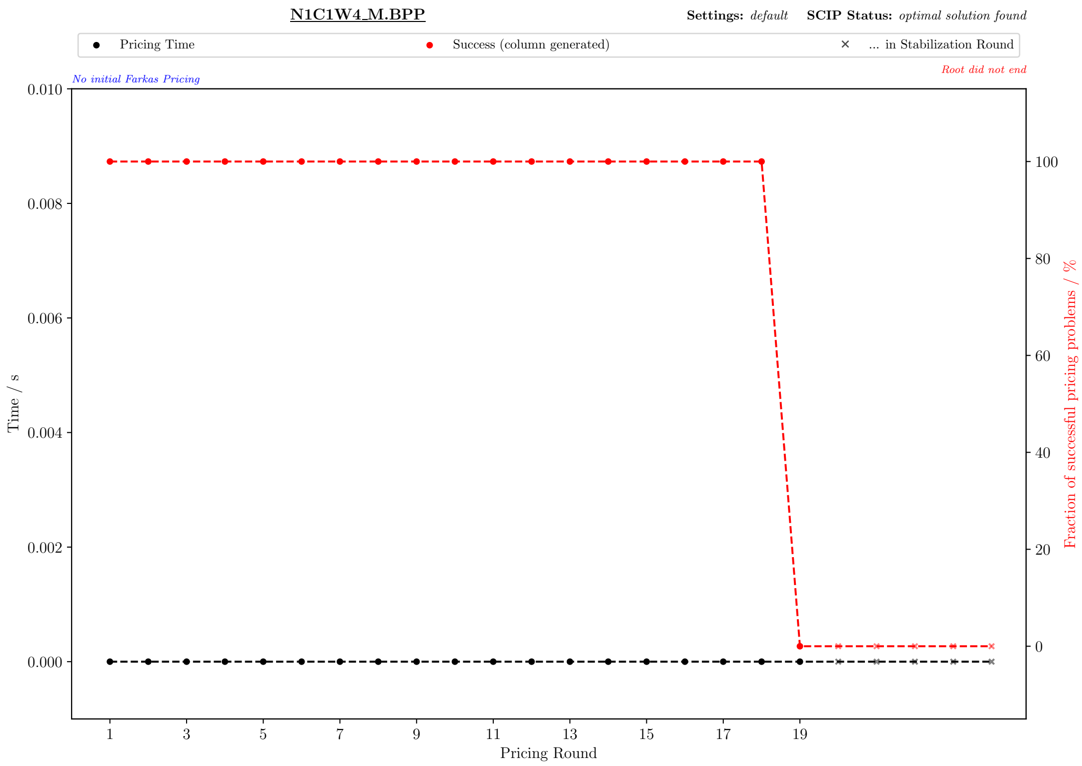


Figure 17: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.summary.pdf`

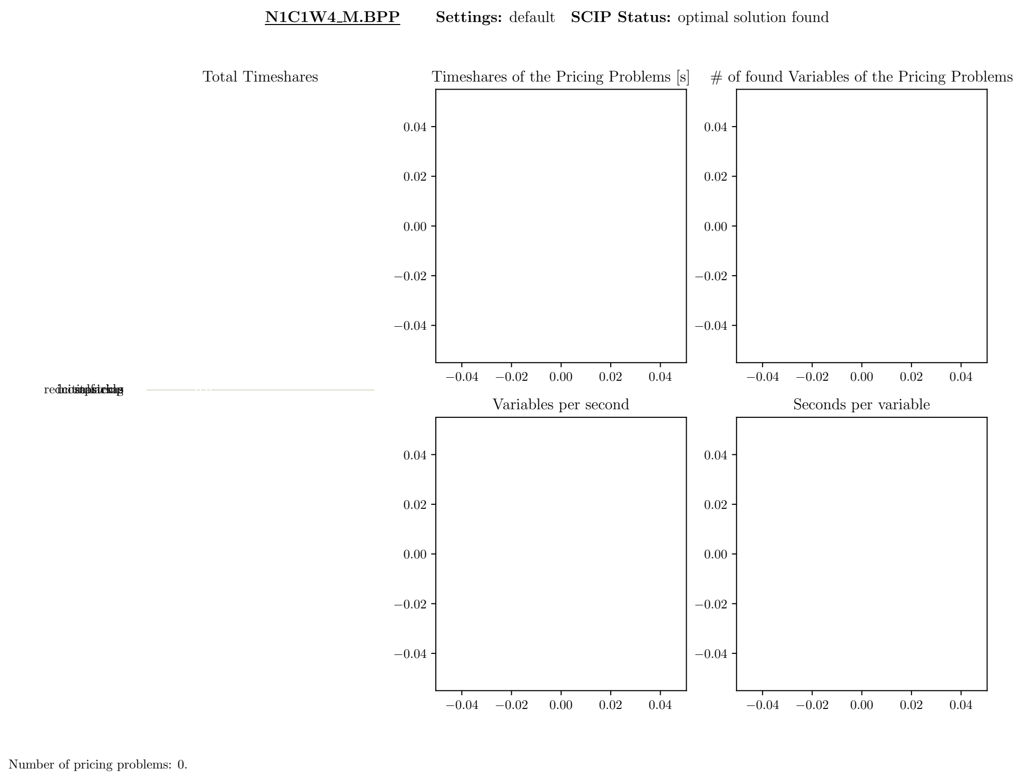


Figure 18: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time ($11°$ of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/N1C1W4_M.BPP.default.time.pdf`

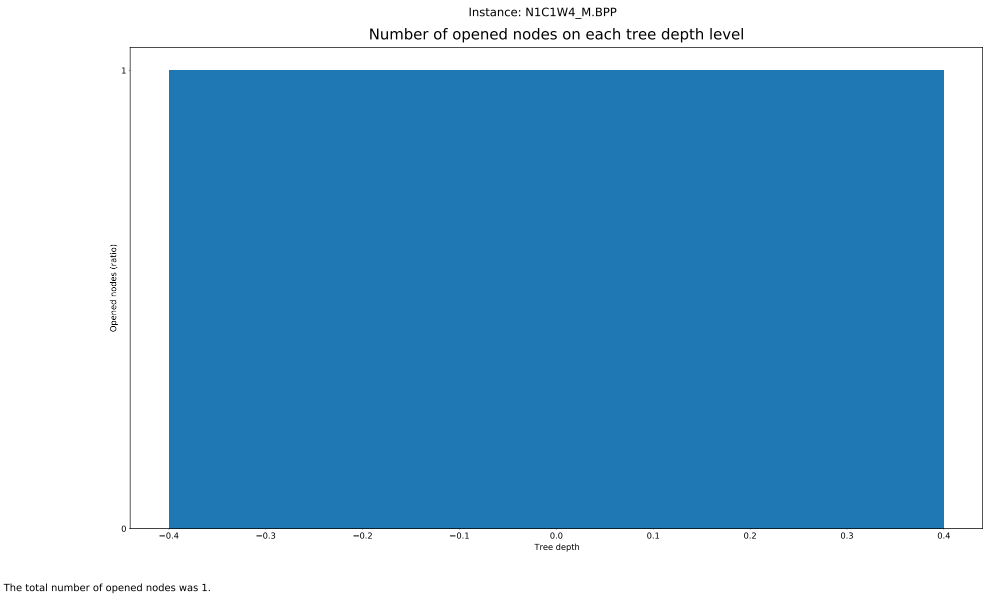


Figure 19: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/N1C1W4_M.BPP.default.tree.bar.pdf`

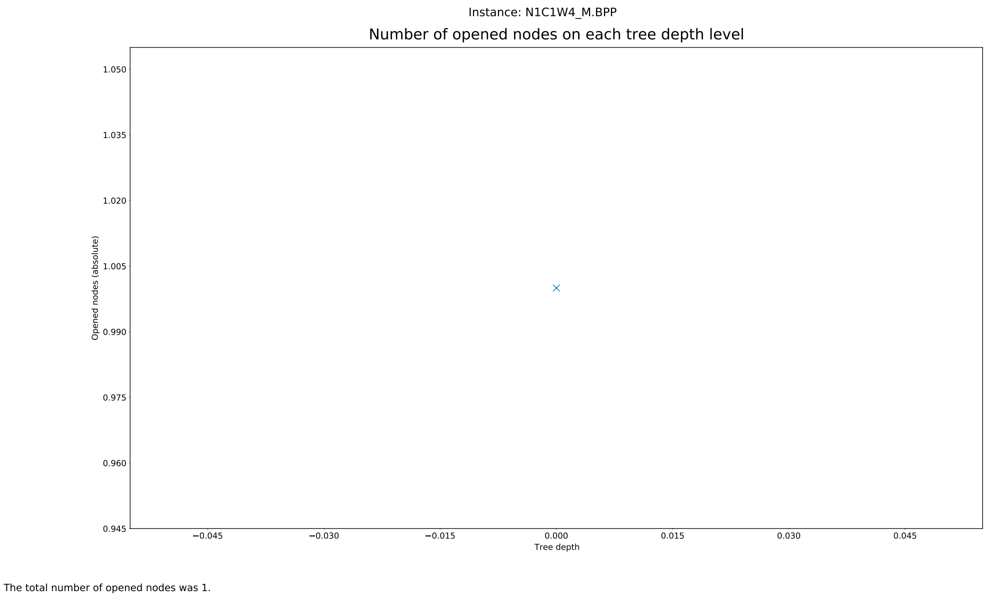


Figure 20: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/N1C1W4_M.BPP.default.tree.plot.pdf`

1.3 Instance: N1C2W2_O.BPP

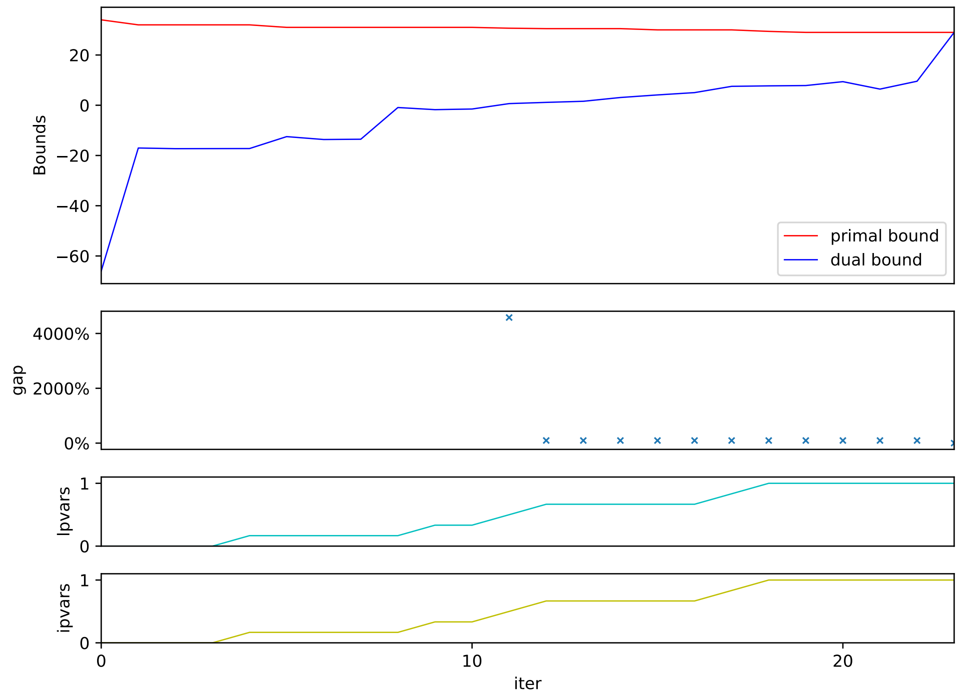


Figure 21: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/N1C2W2_O.BPP.default.bounds.iter.pdf`

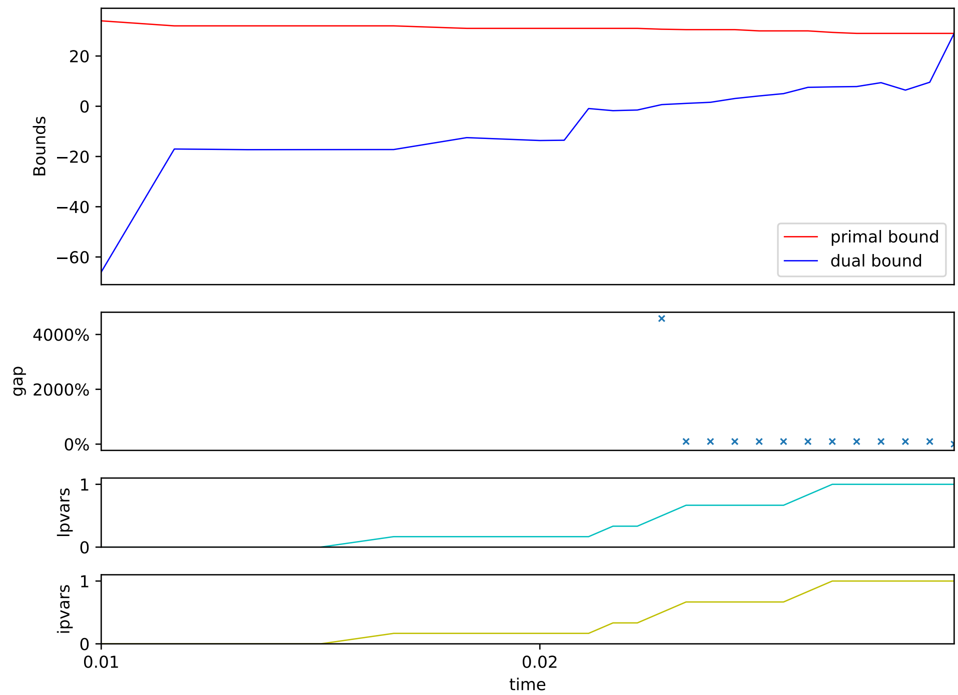


Figure 22: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/N1C2W2_O.BPP.default.bounds.time.pdf`

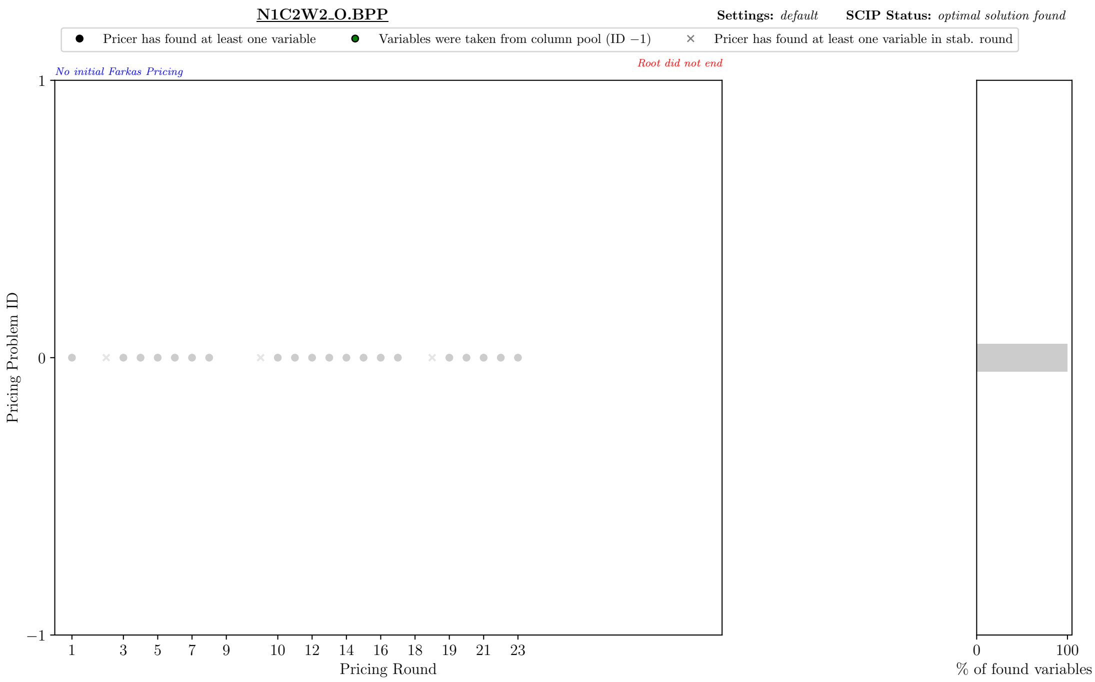


Figure 23: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.bubble.pdf`

Figure 24: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.complete.pdf`

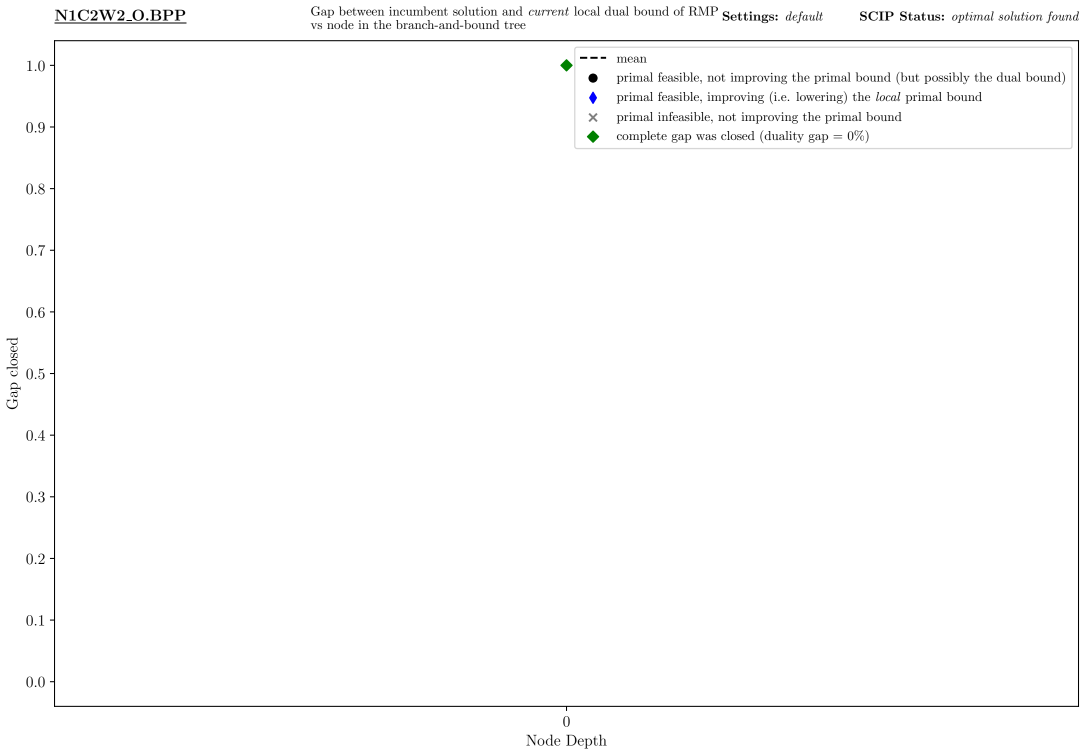


Figure 25: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.depth.pdf`

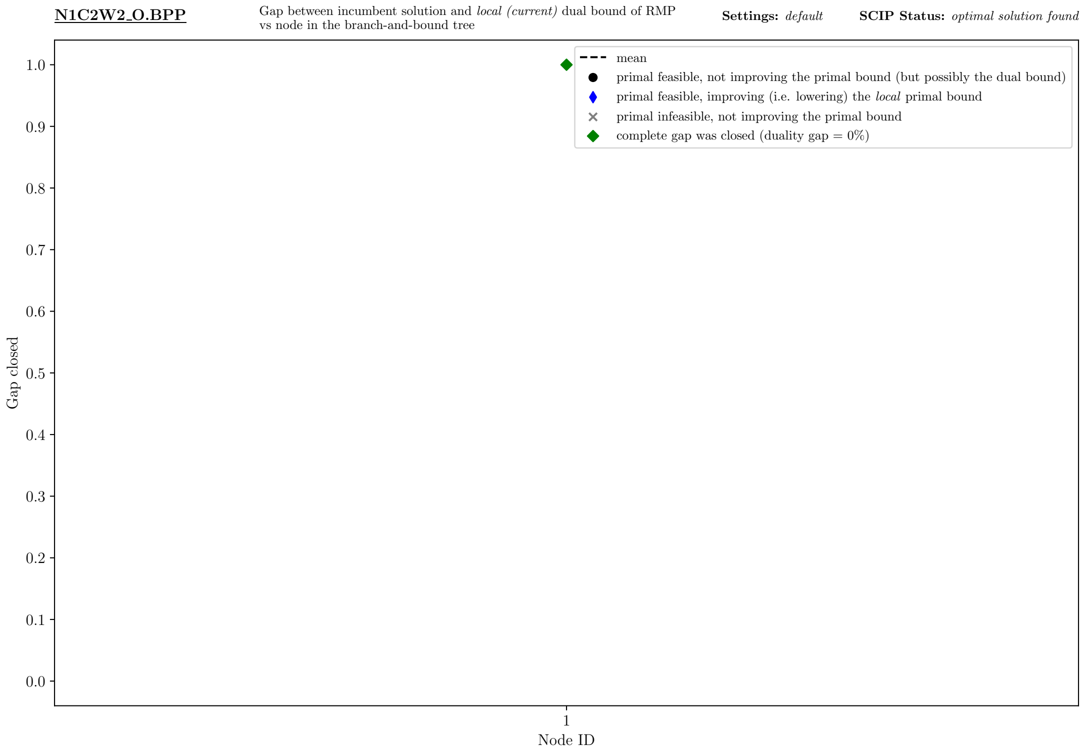


Figure 26: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.nodeID.pdf`

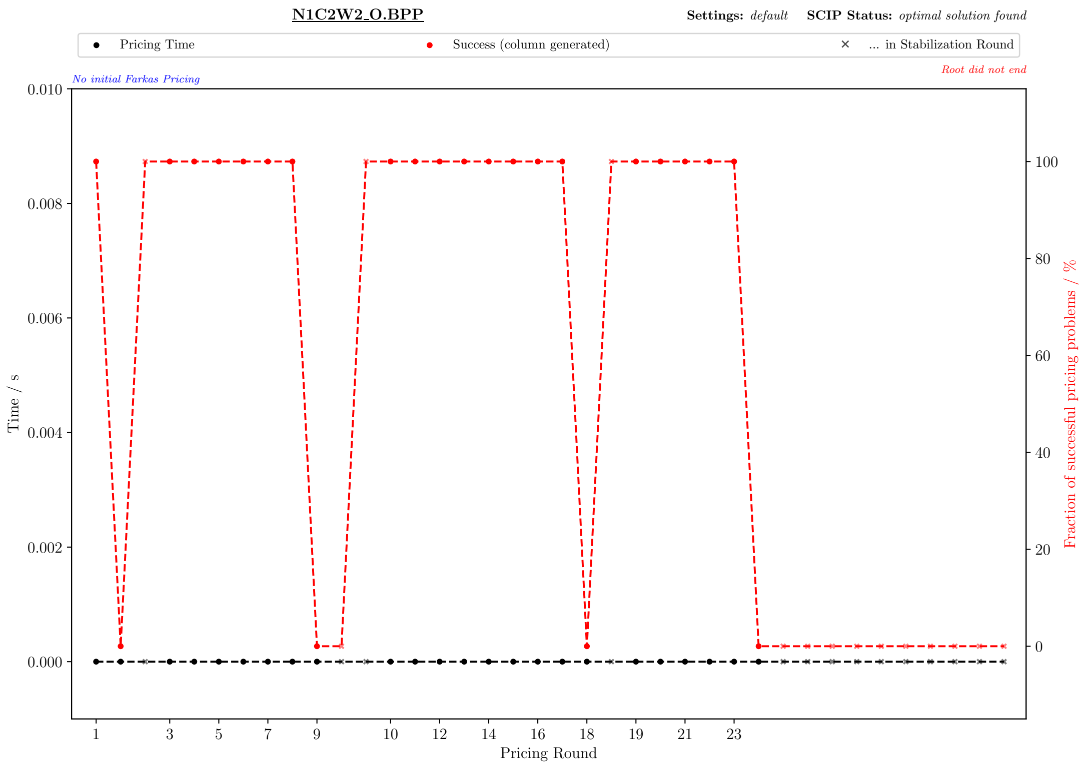


Figure 27: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.summary.pdf`

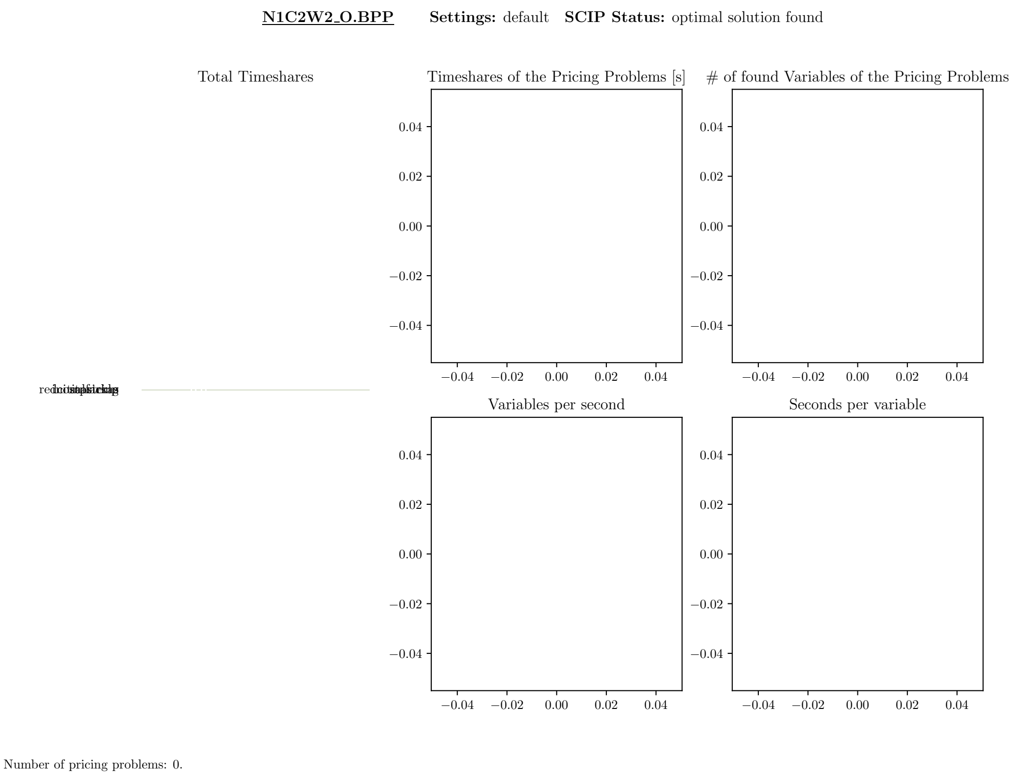
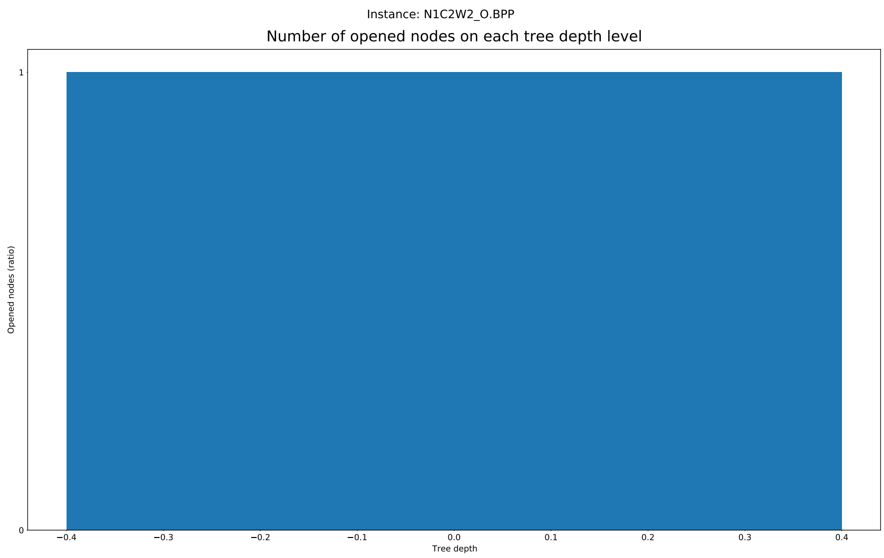


Figure 28: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/N1C2W2_O.BPP.default.time.pdf`



The total number of opened nodes was 1.

Figure 29: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/N1C2W2_O.BPP.default.tree.bar.pdf`

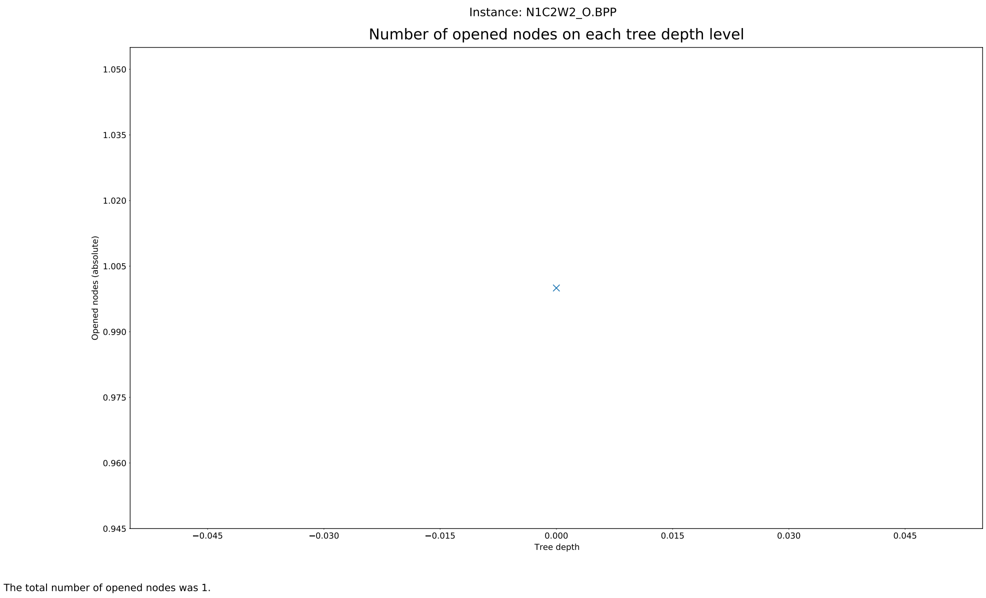


Figure 30: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/N1C2W2_O.BPP.default.tree.plot.pdf`

1.4 Instance: N1C3W1_A

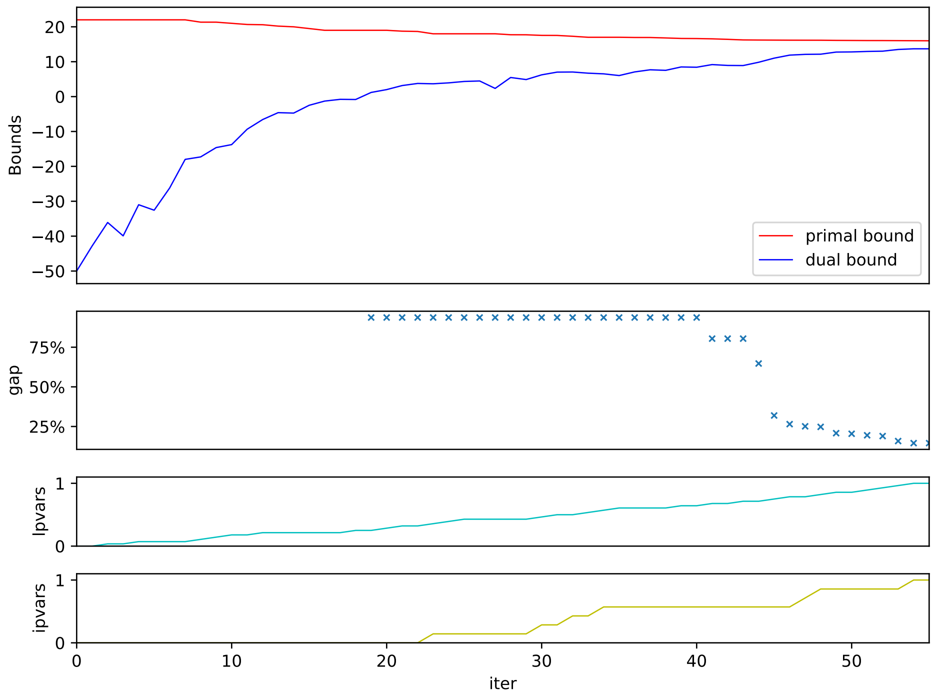


Figure 31: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/N1C3W1_A.default.bounds.iter.pdf`

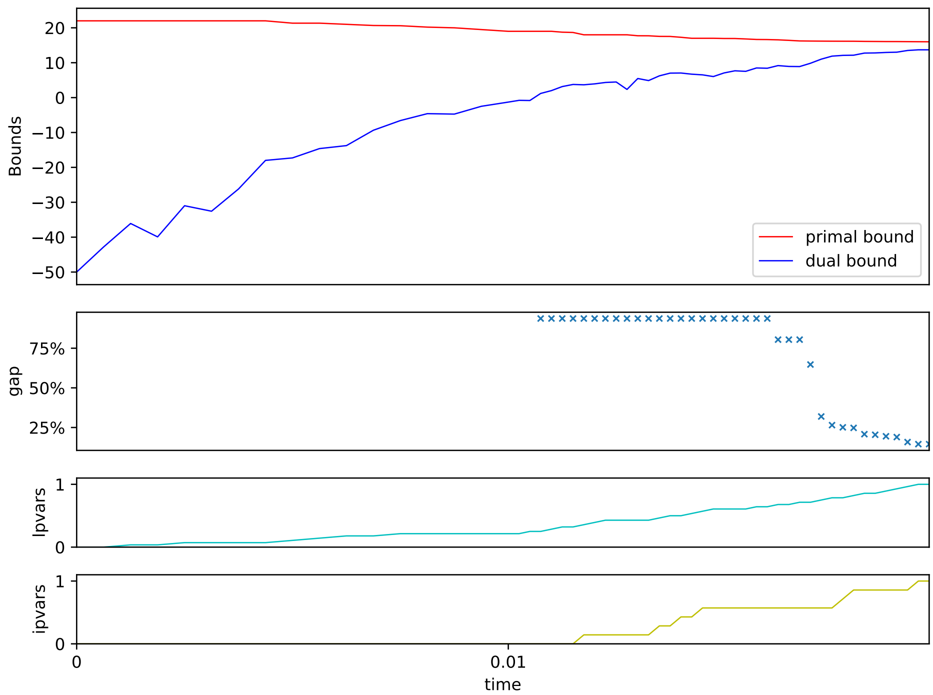


Figure 32: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/N1C3W1_A.default.bounds.time.pdf`

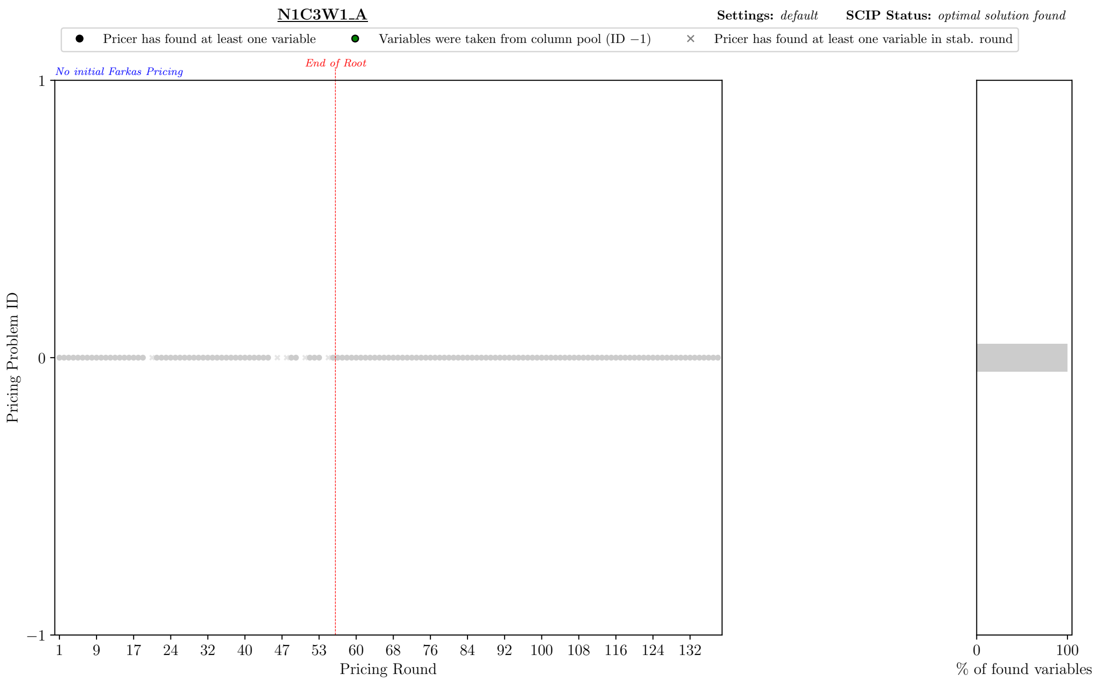


Figure 33: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/N1C3W1_A.default.bubble.pdf`

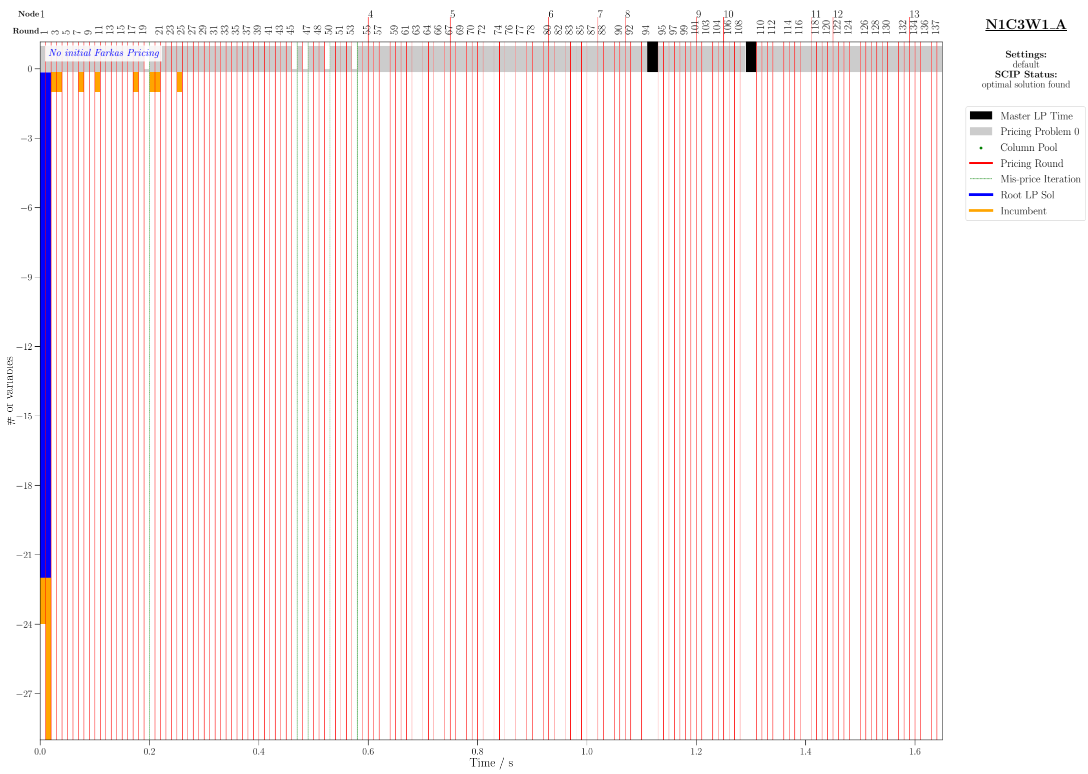


Figure 34: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/N1C3W1_A.default.complete.pdf`

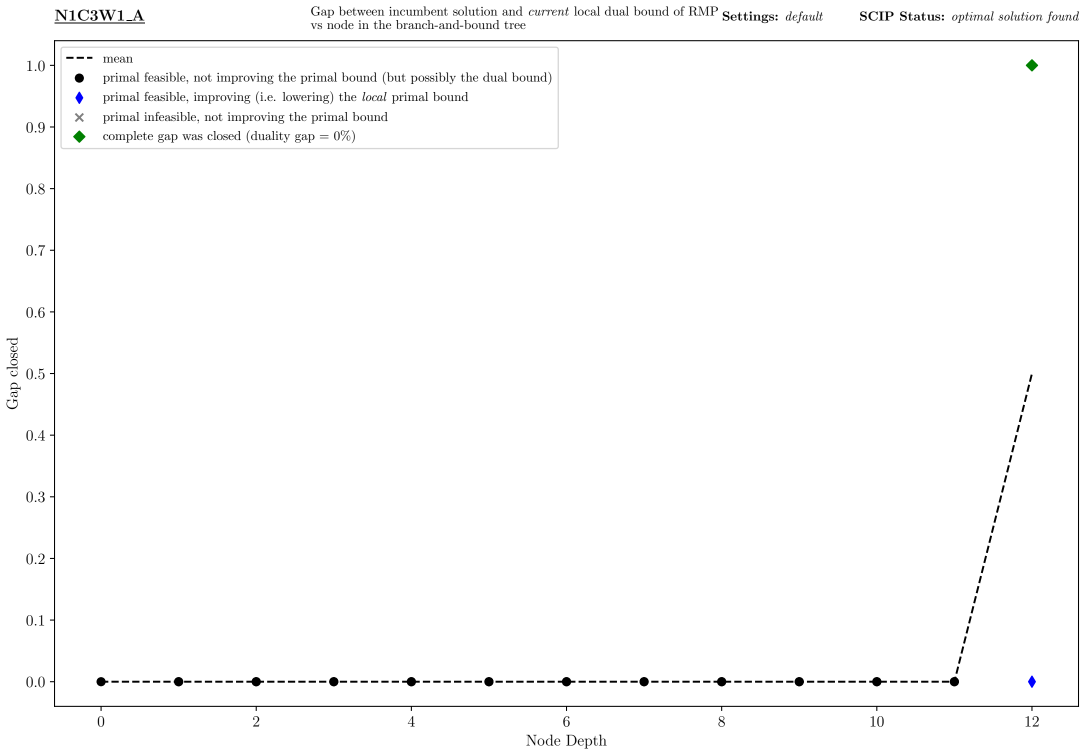


Figure 35: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/N1C3W1_A.default.depth.pdf`

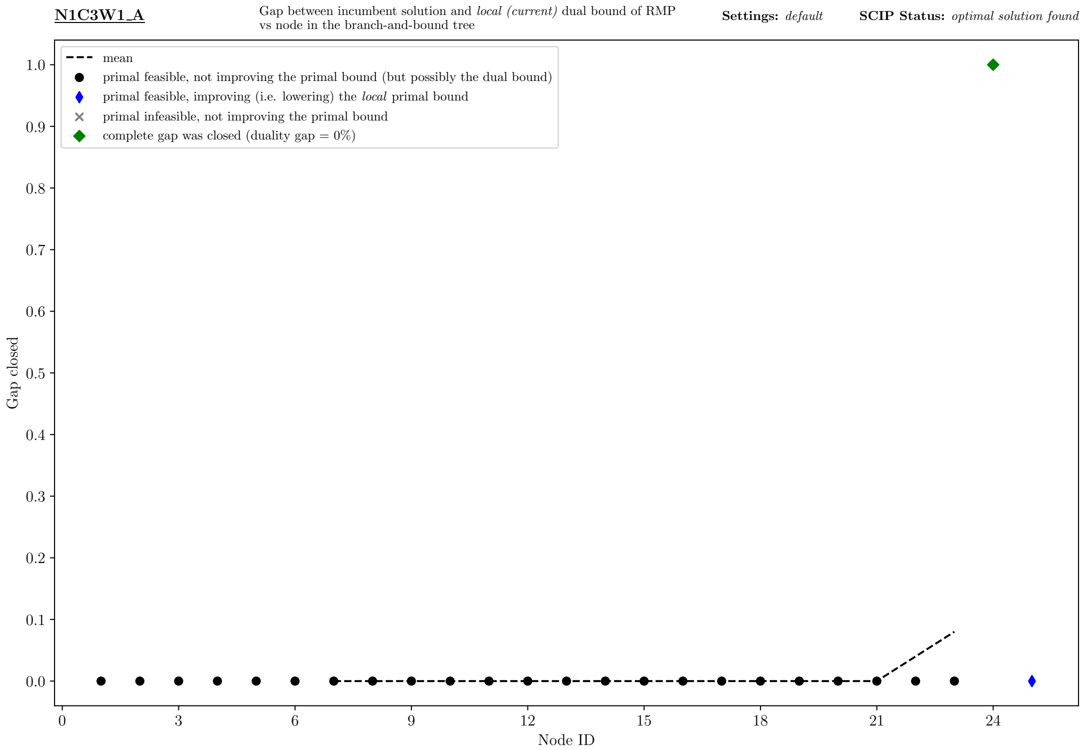


Figure 36: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/N1C3W1_A.default.nodeID.pdf`

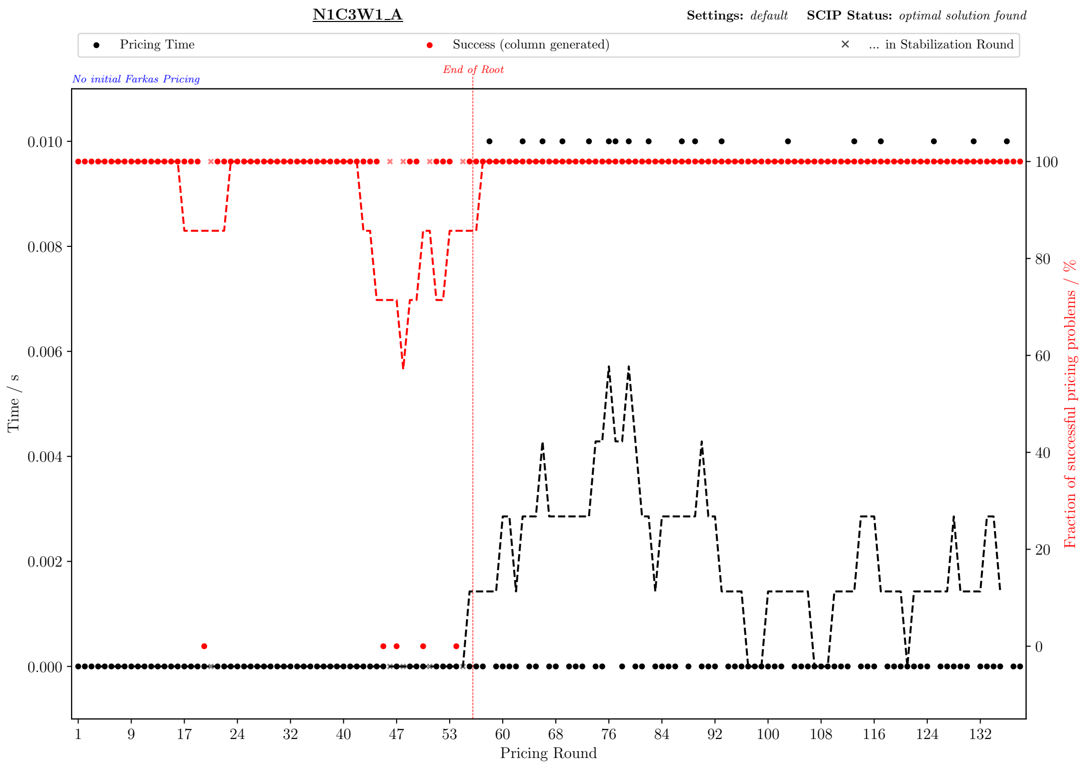


Figure 37: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/N1C3W1_A.default.summary.pdf`

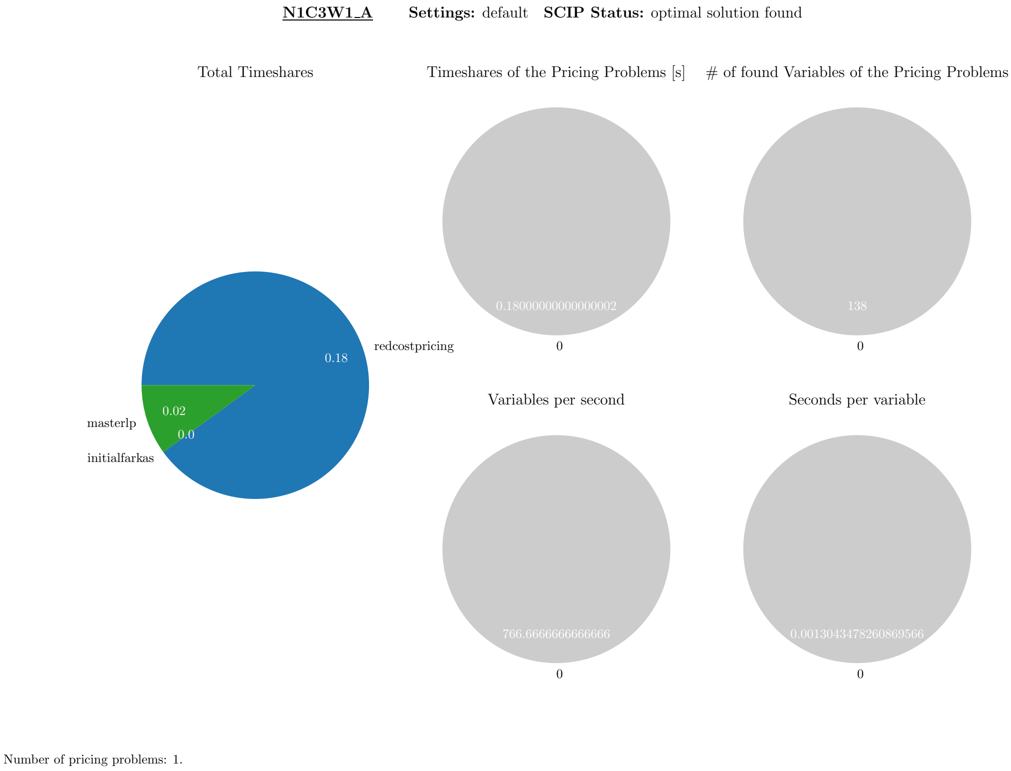
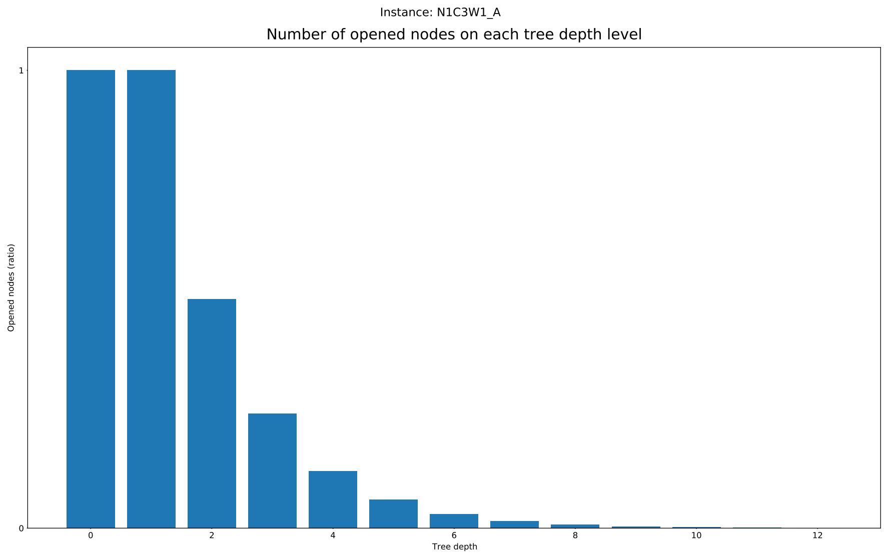
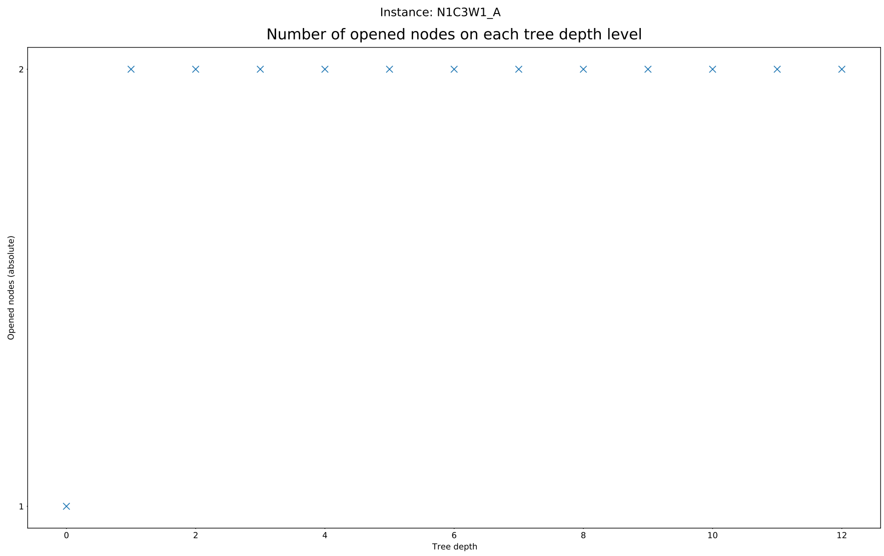


Figure 38: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time ($11°$ of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/N1C3W1_A.default.time.pdf`



The total number of opened nodes was 25.

Figure 39: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/N1C3W1_A.default.tree.bar.pdf`



The total number of opened nodes was 25.

Figure 40: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/N1C3W1_A.default.tree.plot.pdf`

1.5 Instance: p1250-2

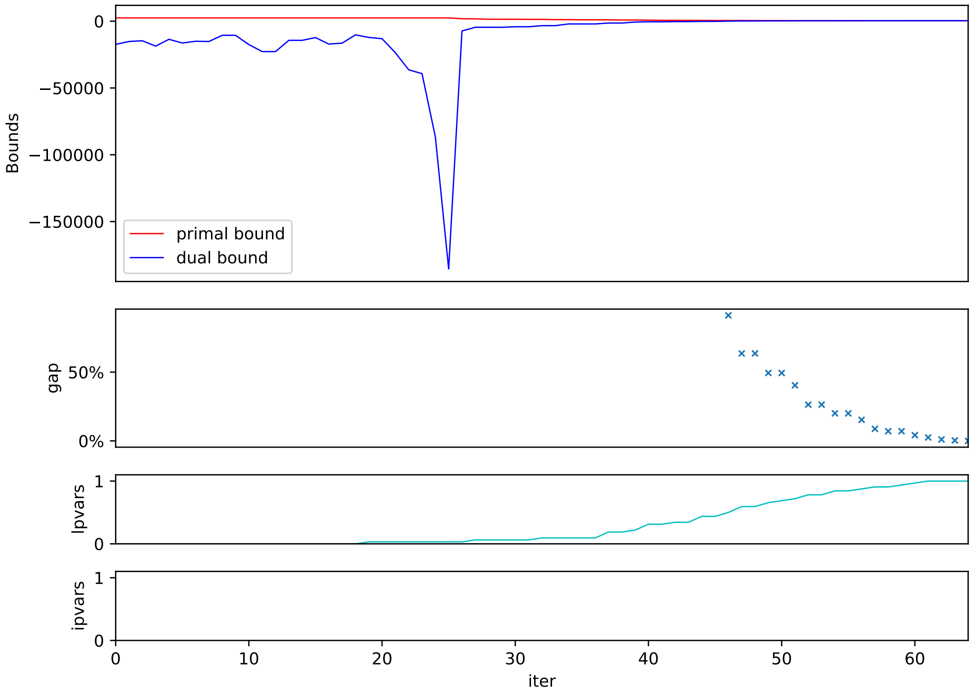


Figure 41: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/p1250-2.default.bounds.iter.pdf`

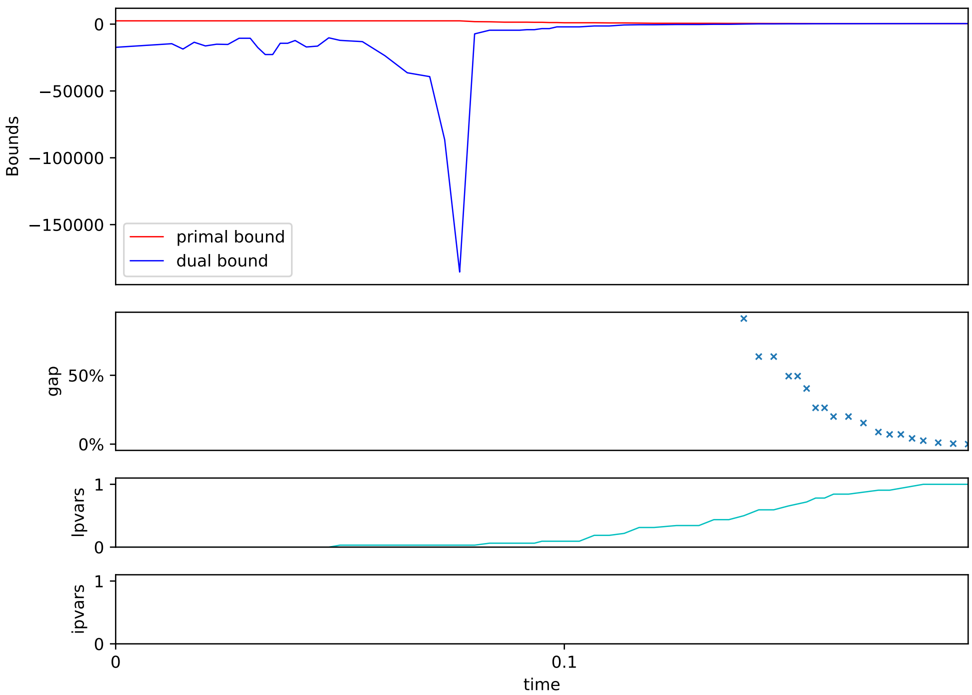


Figure 42: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds” printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/p1250-2.default.bounds.time.pdf`

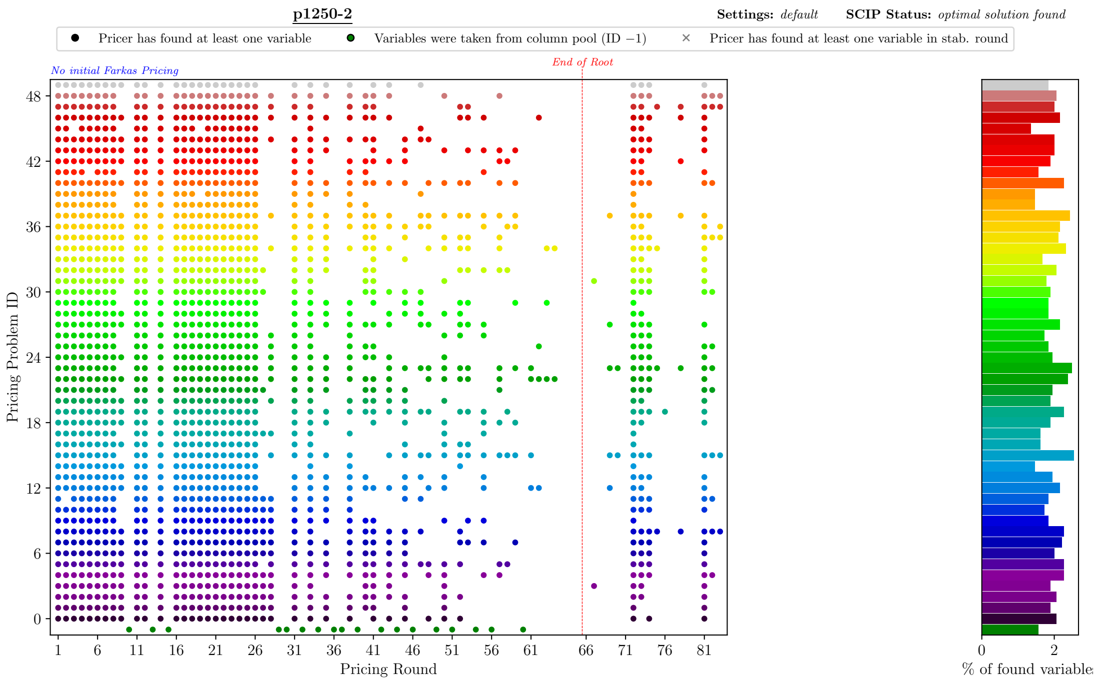


Figure 43: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/p1250-2.default.bubble.pdf`

Figure 44: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/p1250-2.default.complete.pdf`

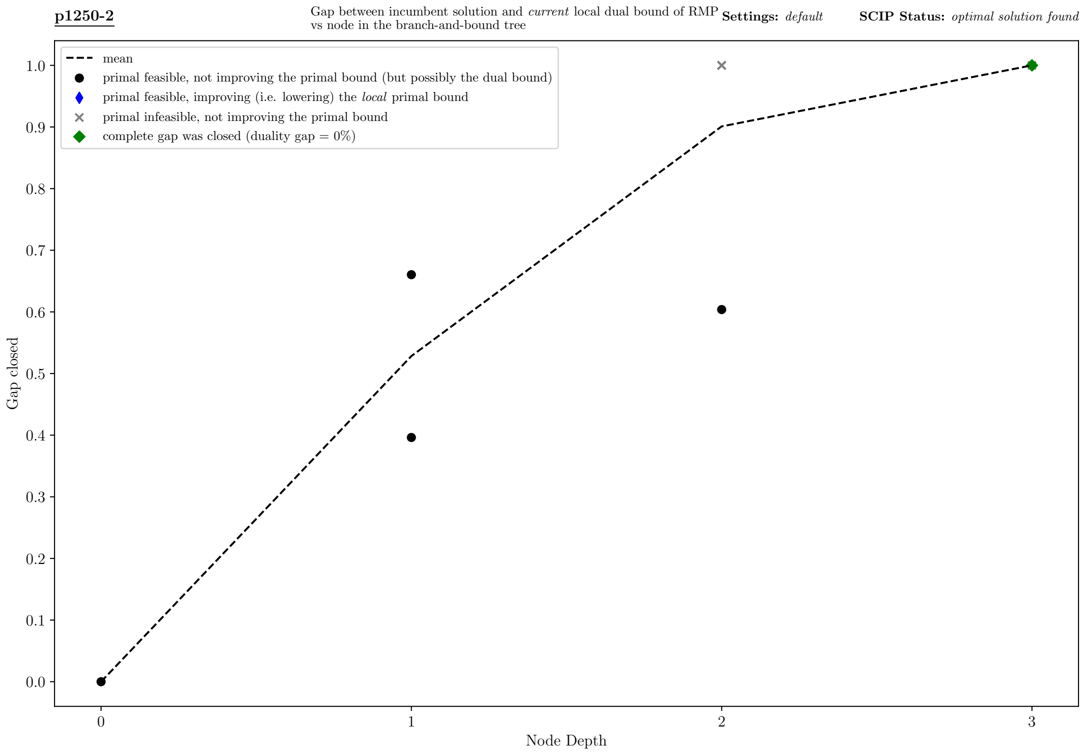


Figure 45: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/p1250-2.default.depth.pdf`

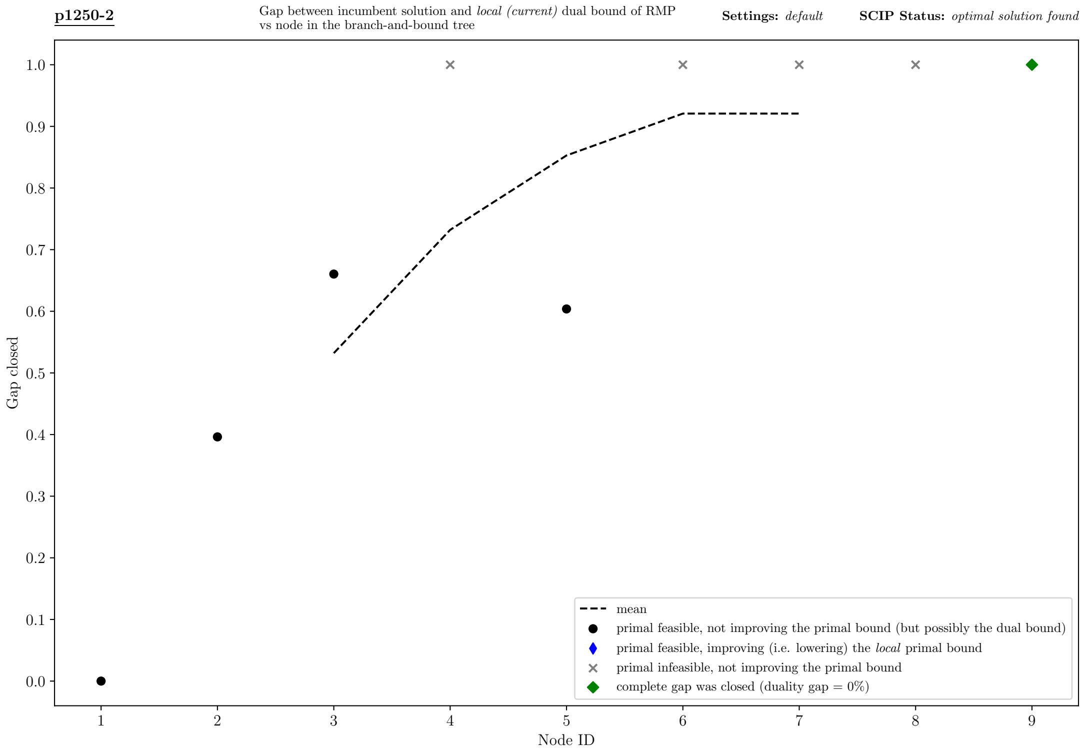


Figure 46: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/p1250-2.default.nodeID.pdf`

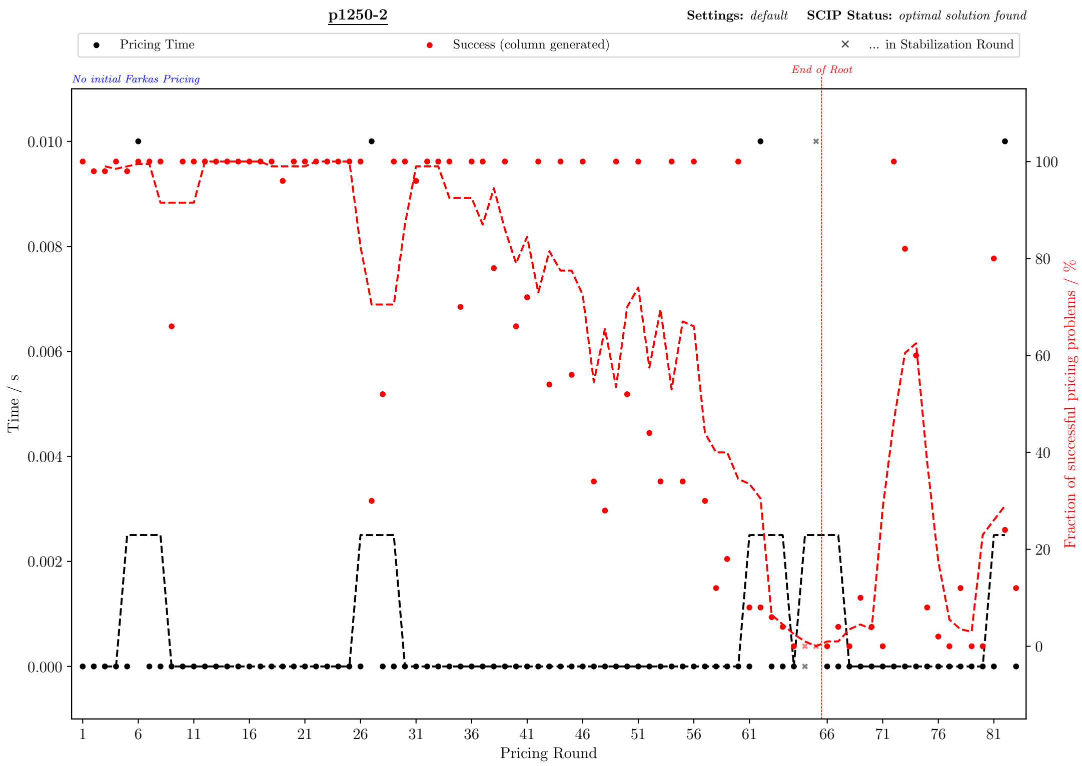


Figure 47: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/p1250-2.default.summary.pdf`

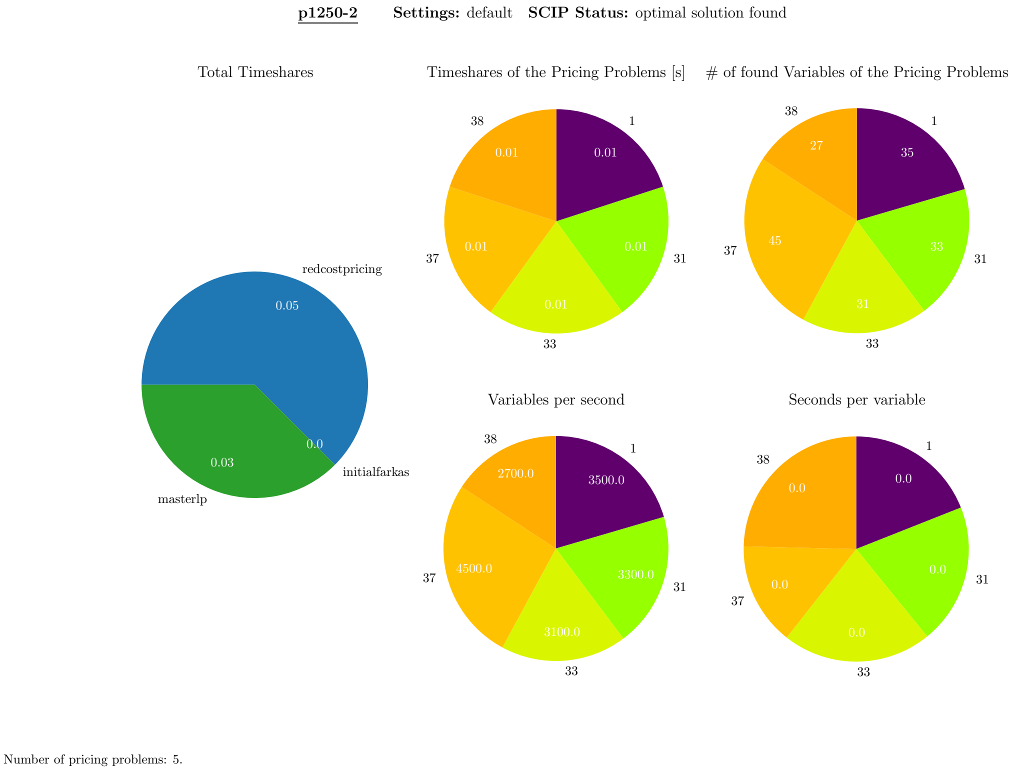
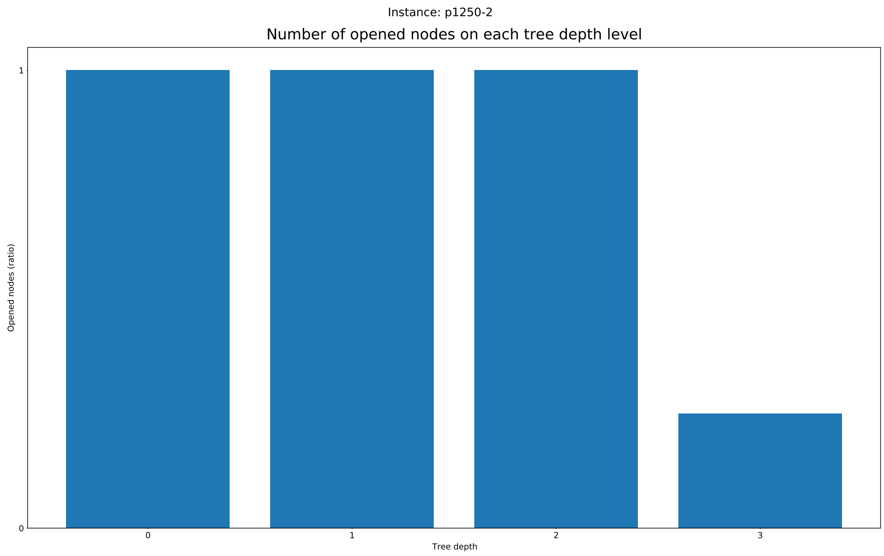
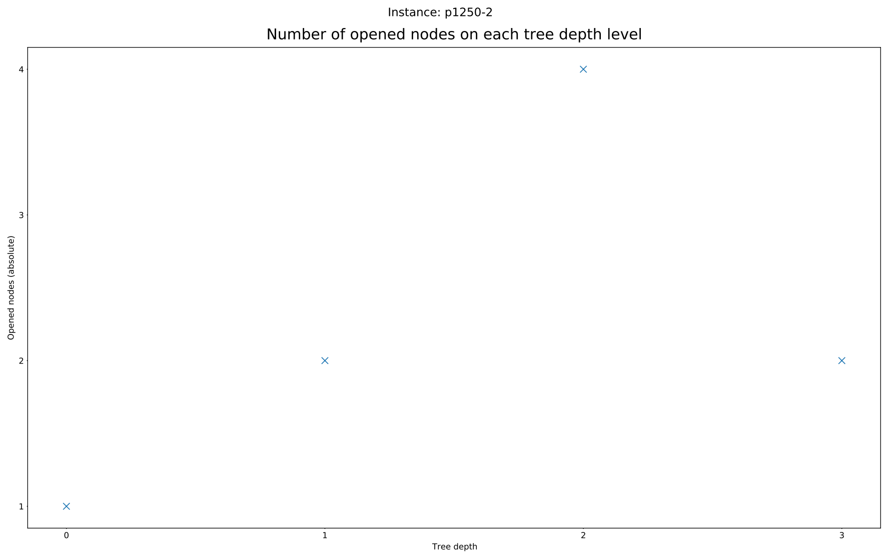


Figure 48: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time ($11°$ of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/p1250-2.default.time.pdf`



The total number of opened nodes was 9.

Figure 49: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/p1250-2.default.tree.bar.pdf`



The total number of opened nodes was 9.

Figure 50: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/p1250-2.default.tree.plot.pdf`

1.6 Instance: p1650-2.txt

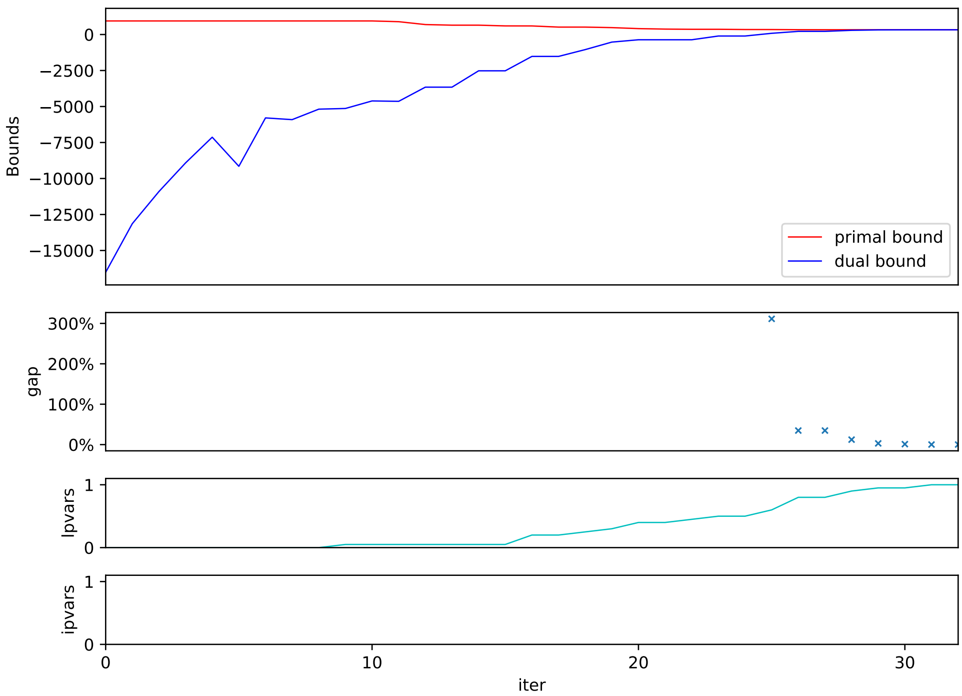


Figure 51: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/p1650-2.txt.default.bounds.iter.pdf`

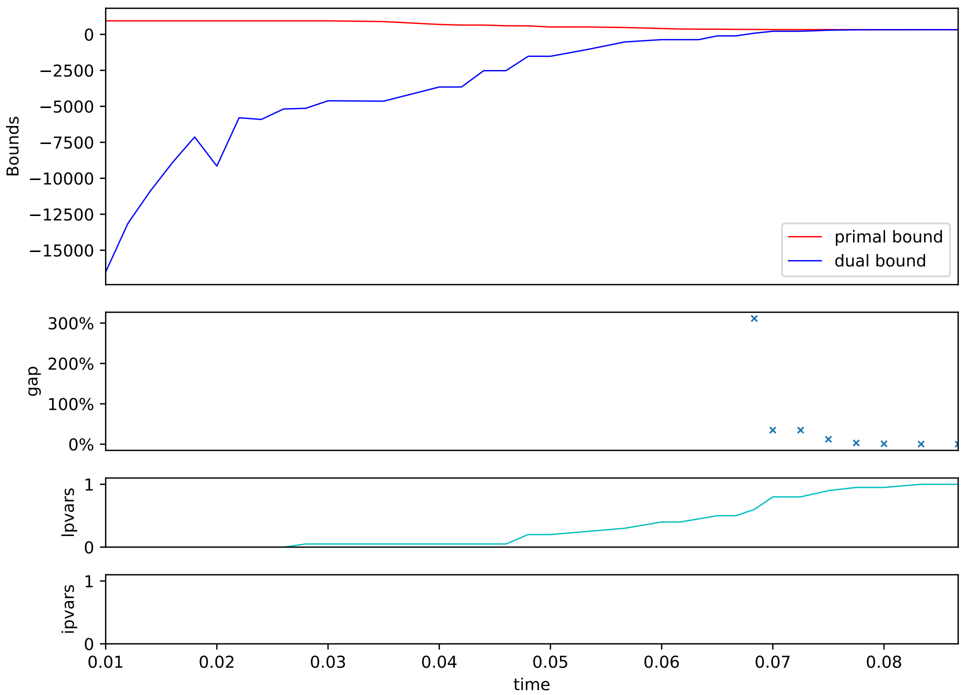


Figure 52: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/p1650-2.txt.default.bounds.time.pdf`

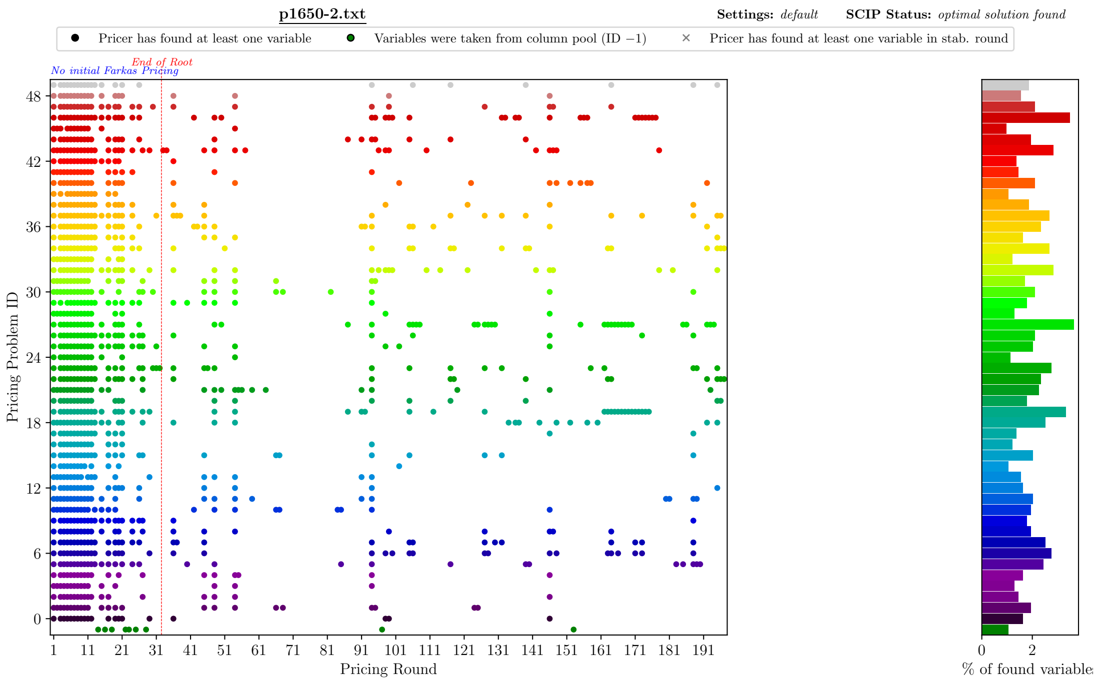


Figure 53: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/p1650-2.txt.default.bubble.pdf`

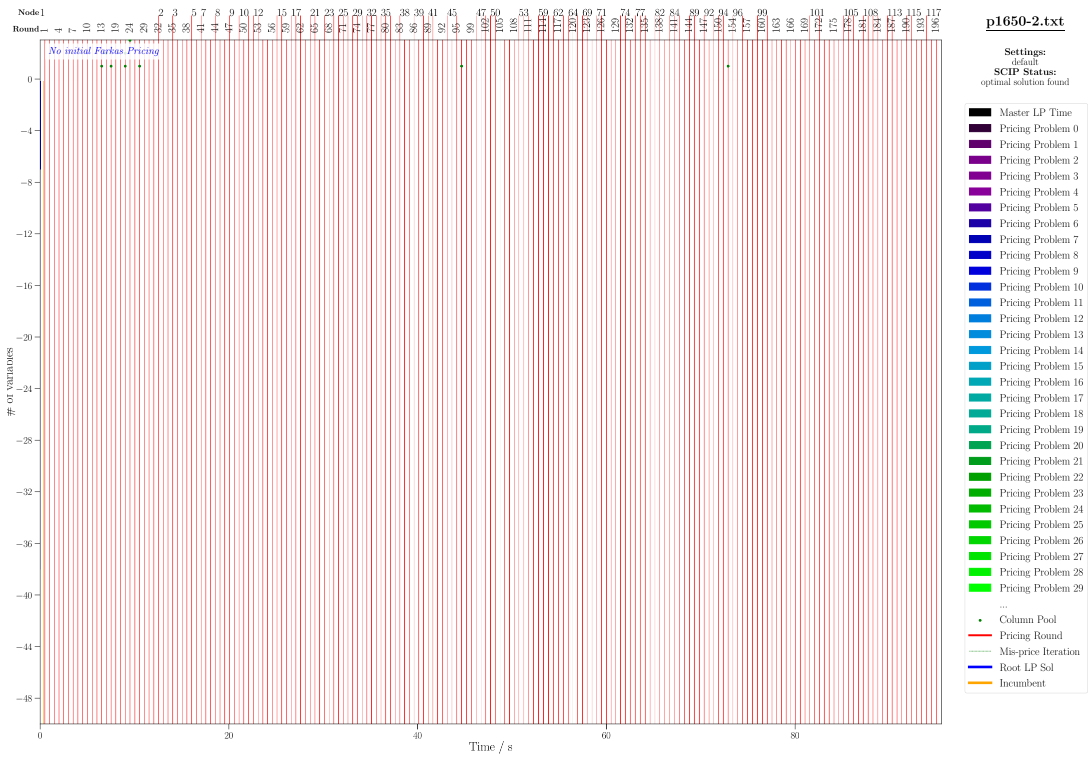


Figure 54: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/p1650-2.txt.default.complete.pdf`

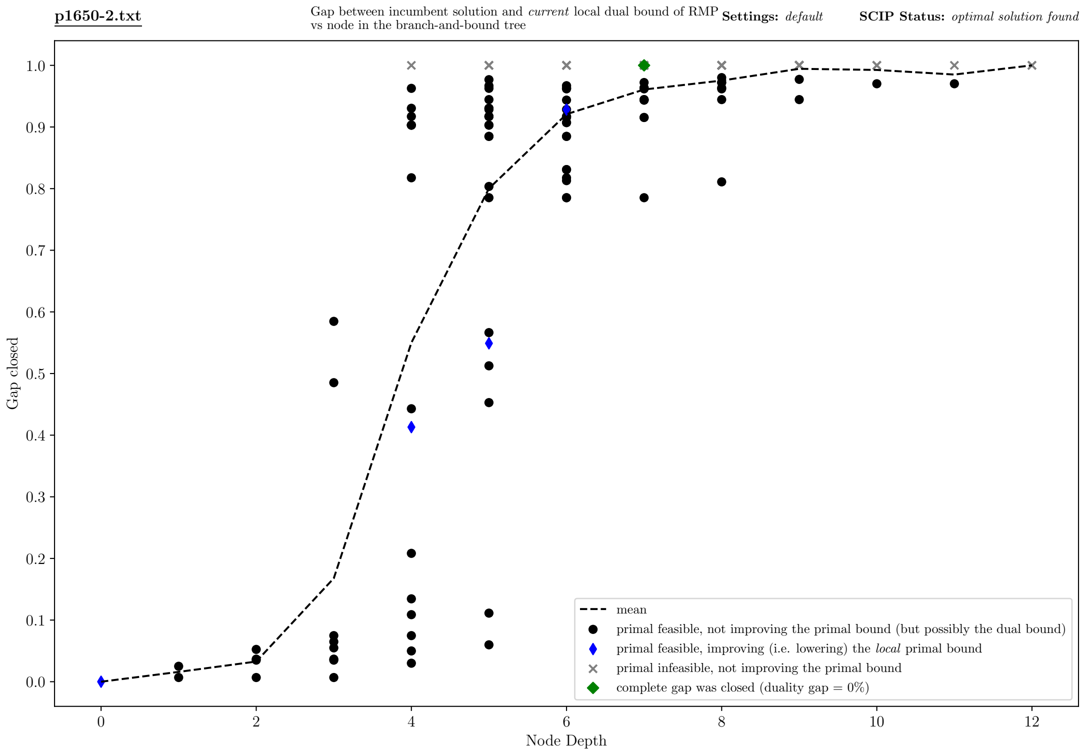


Figure 55: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/p1650-2.txt.default.depth.pdf`

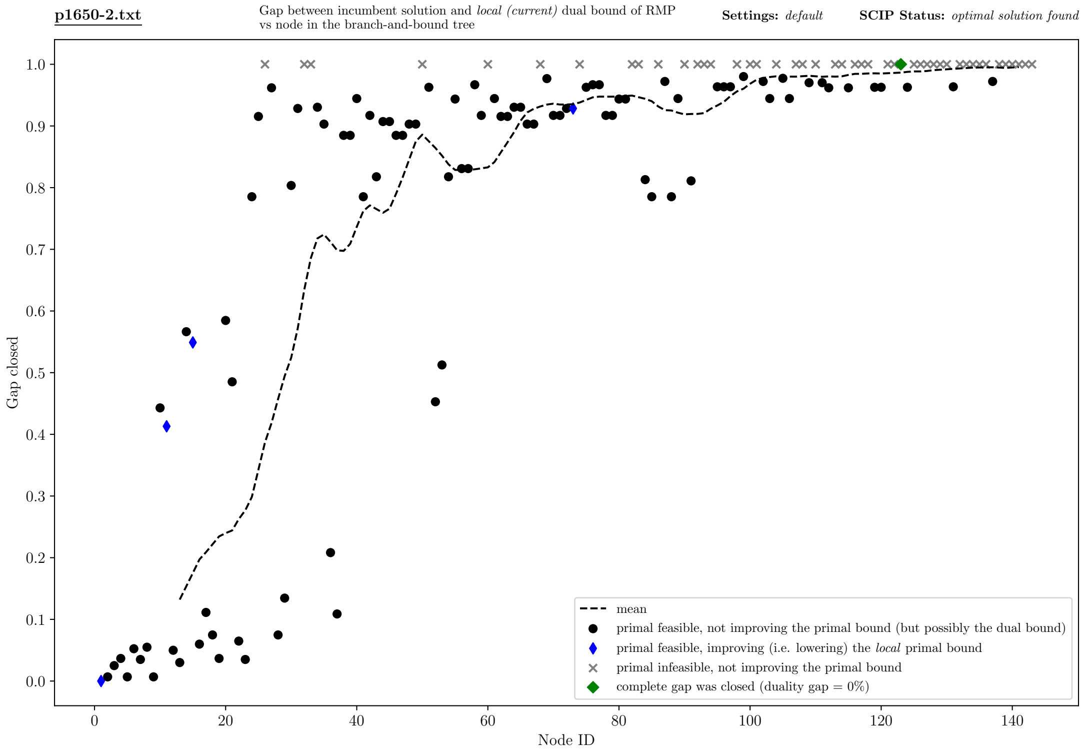


Figure 56: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/p1650-2.txt.default.nodeID.pdf`

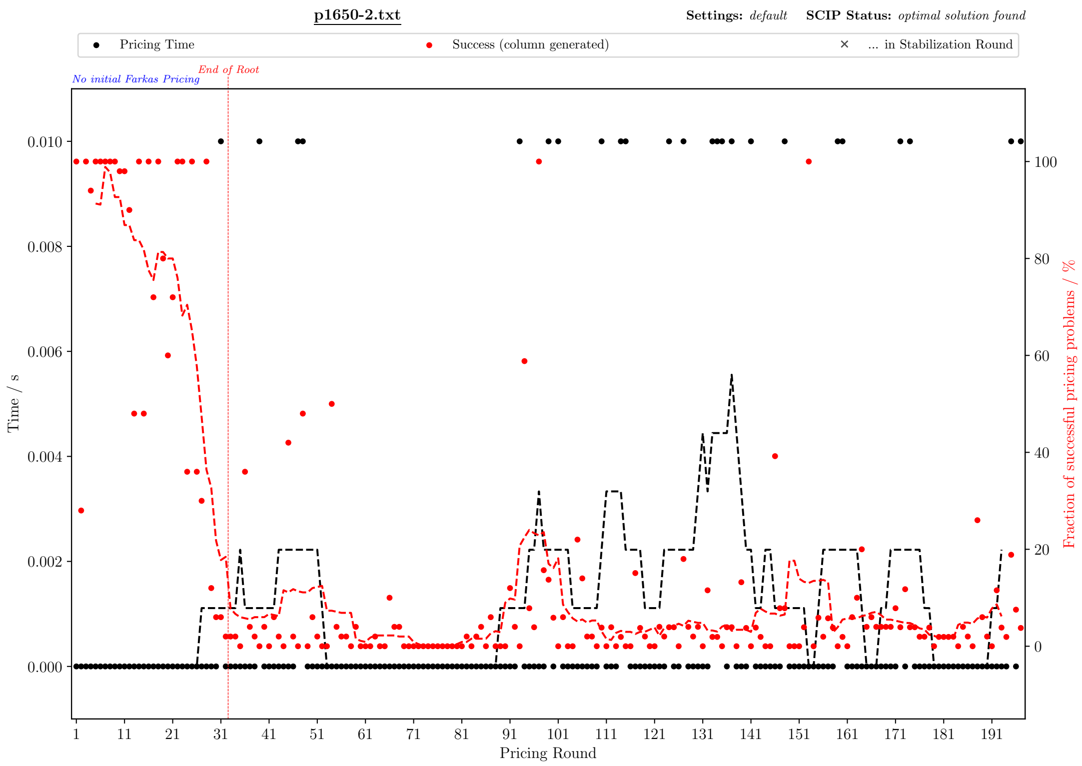


Figure 57: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/p1650-2.txt.default.summary.pdf`

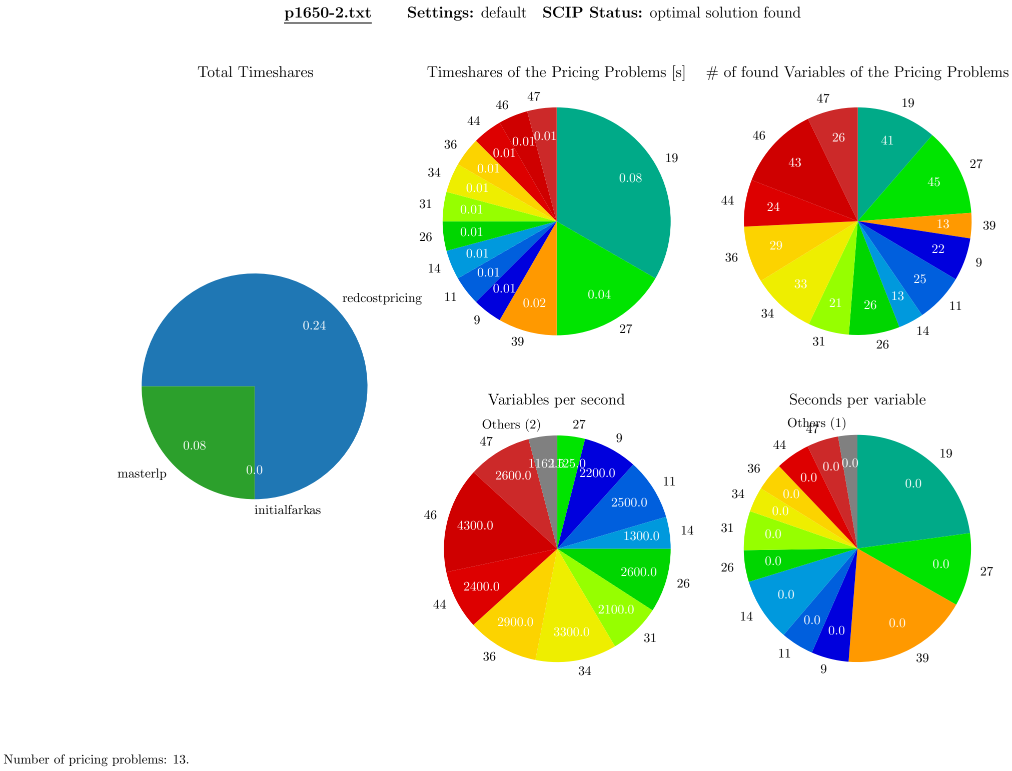


Figure 58: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/p1650-2.txt.default.time.pdf`

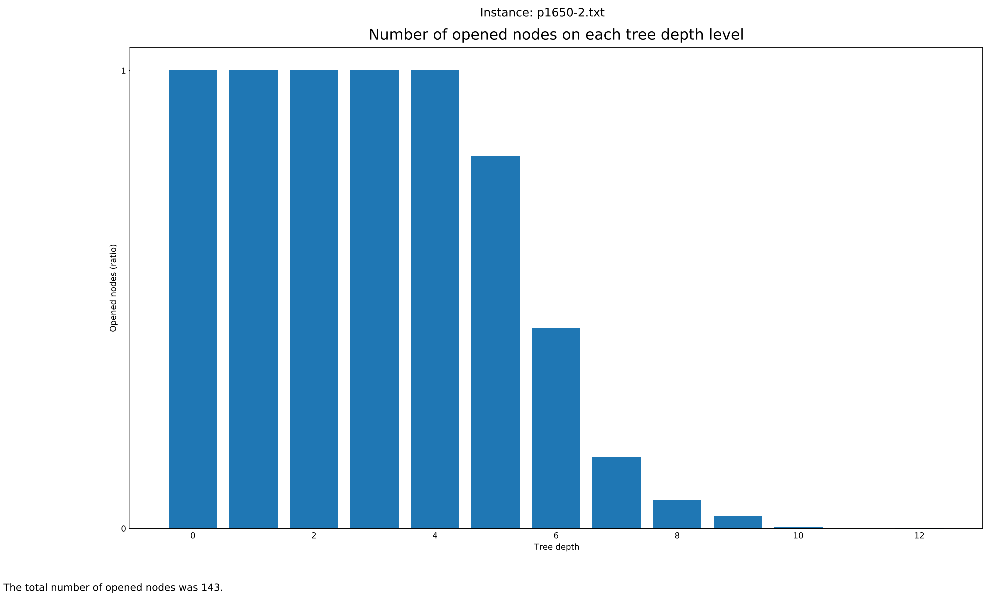
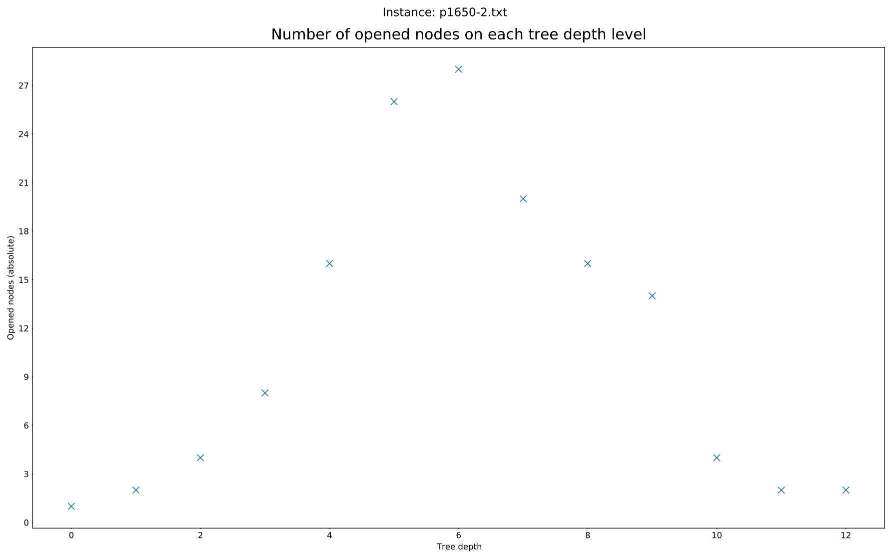


Figure 59: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/p1650-2.txt.default.tree.bar.pdf`



The total number of opened nodes was 143.

Figure 60: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/p1650-2.txt.default.tree.plot.pdf`

1.7 Instance: p2050-1.txt

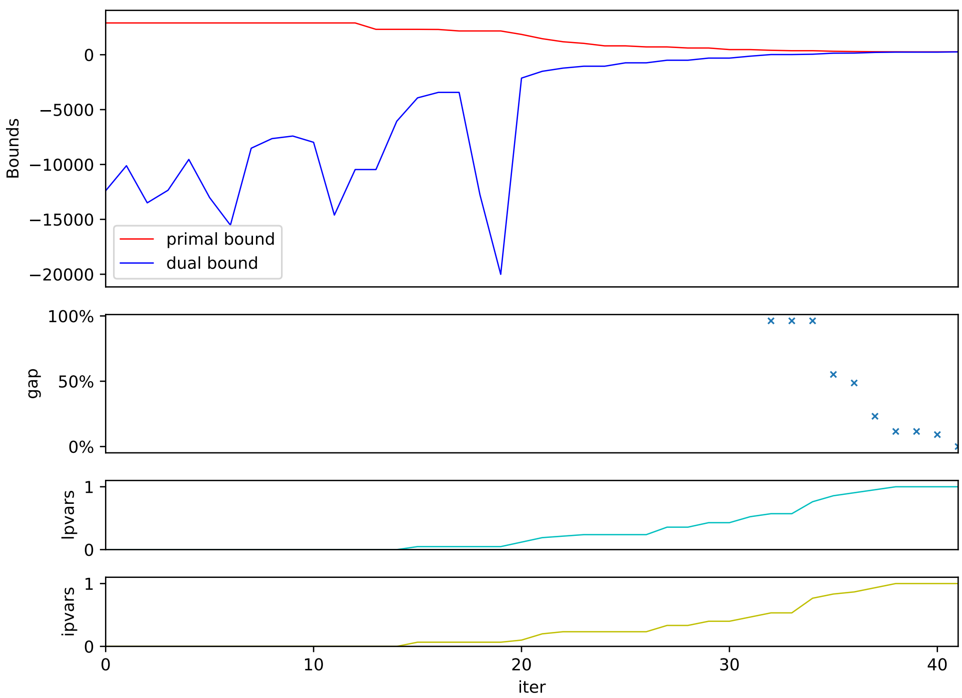


Figure 61: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/p2050-1.txt.default.bounds.iter.pdf`

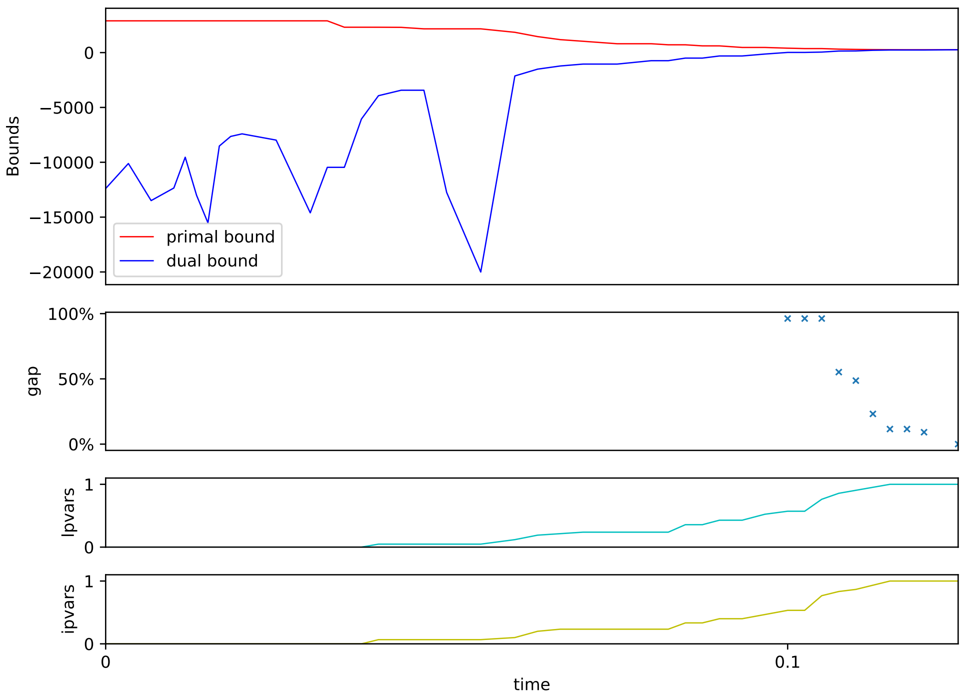


Figure 62: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/p2050-1.txt.default.bounds.time.pdf`

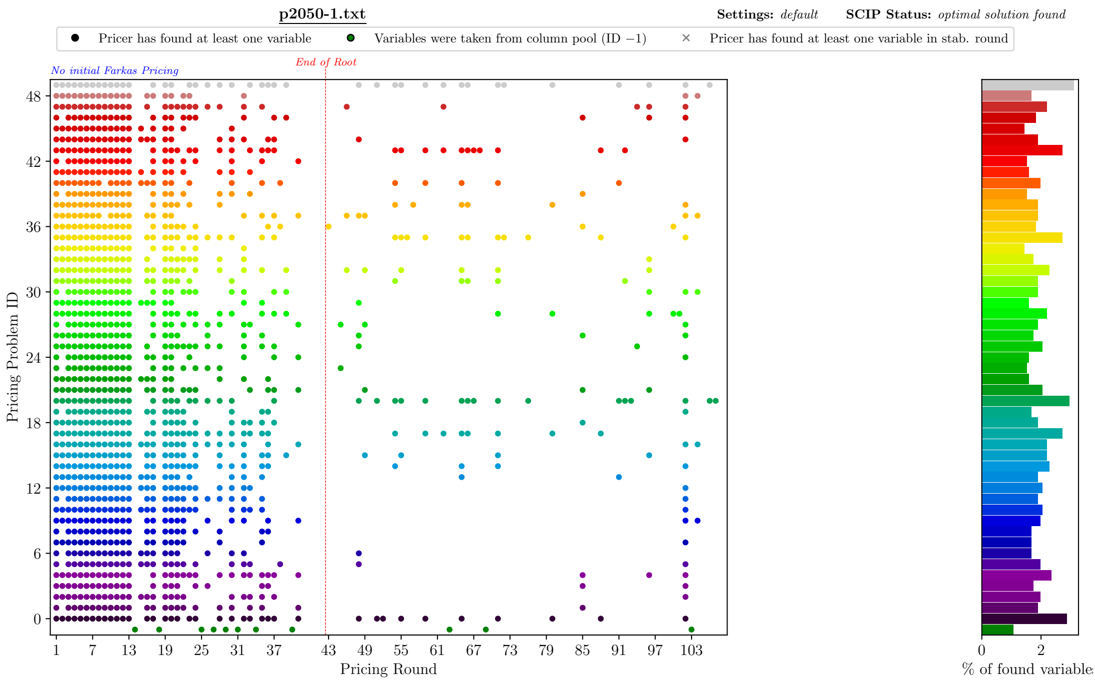


Figure 63: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/p2050-1.txt.default.bubble.pdf`

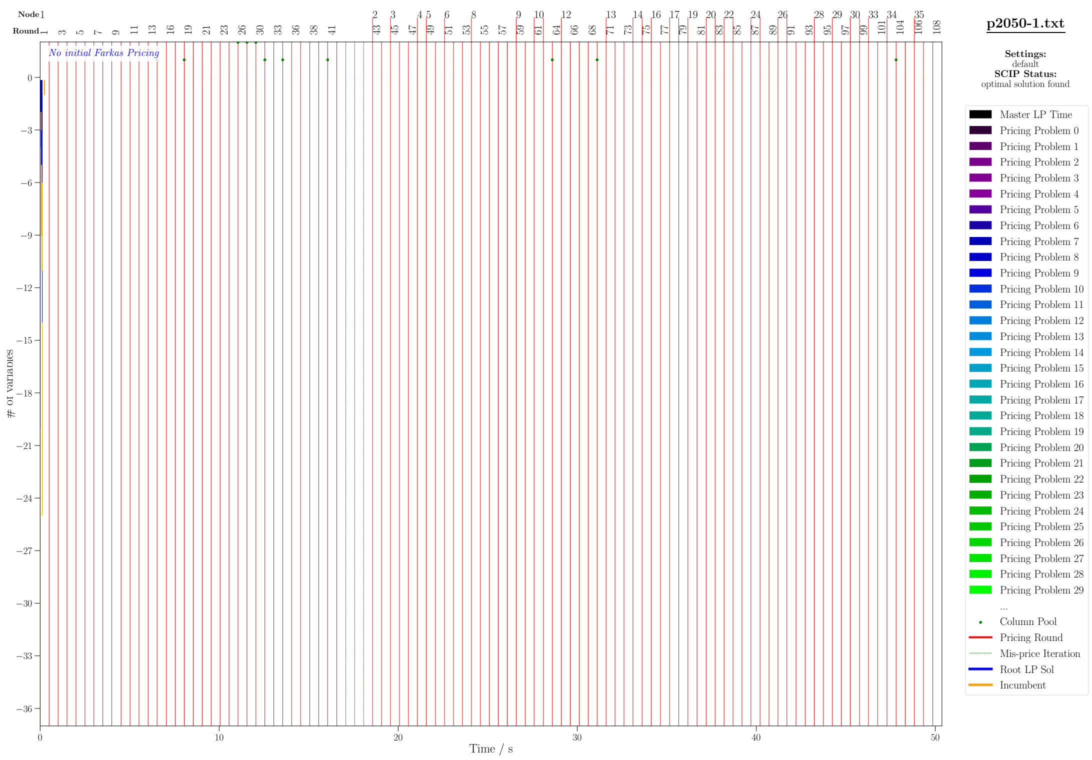


Figure 64: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/p2050-1.txt.default.complete.pdf`

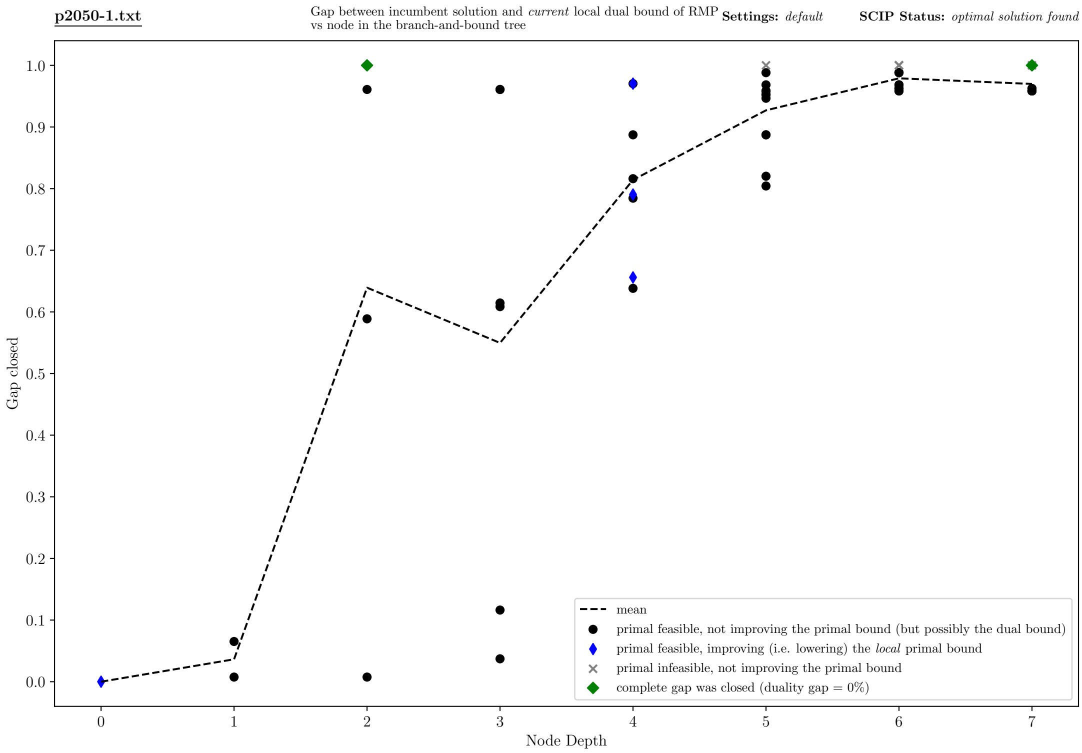


Figure 65: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/p2050-1.txt.default.depth.pdf`

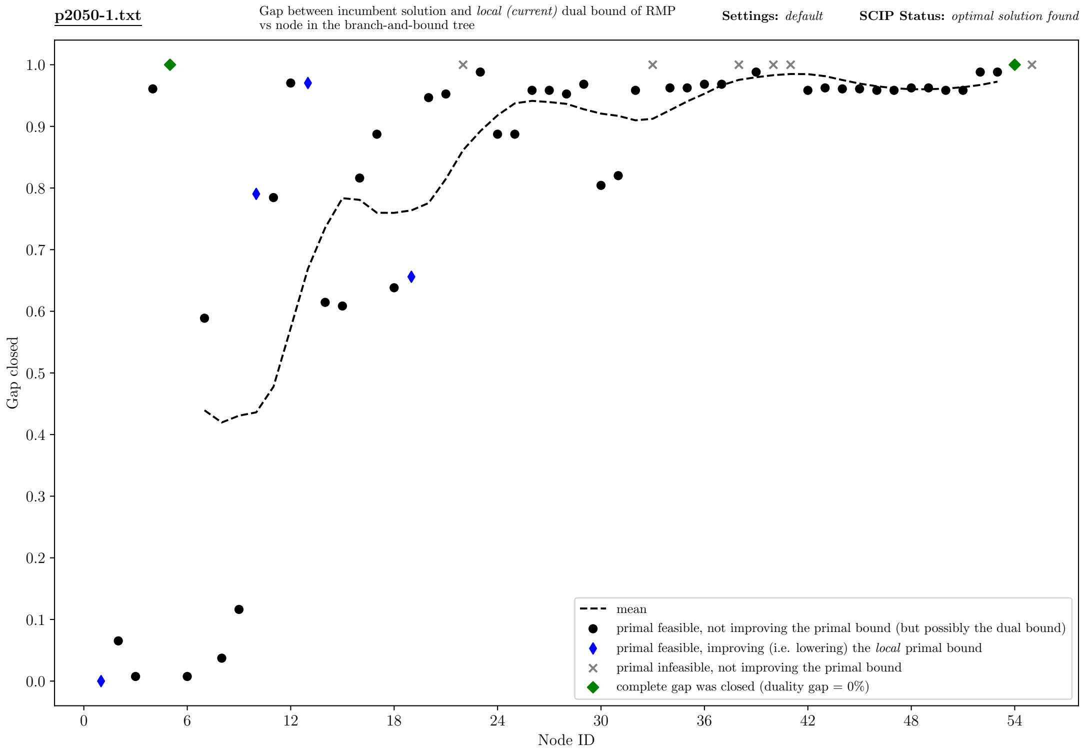


Figure 66: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/p2050-1.txt.default.nodeID.pdf`

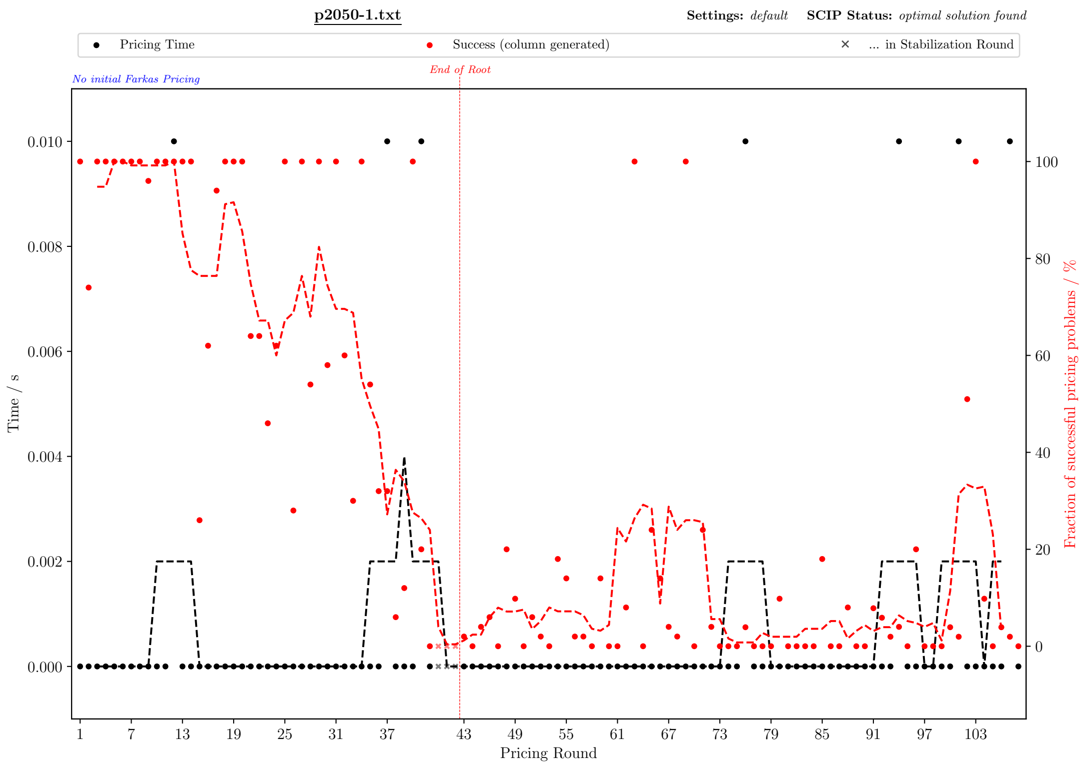


Figure 67: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/p2050-1.txt.default.summary.pdf`

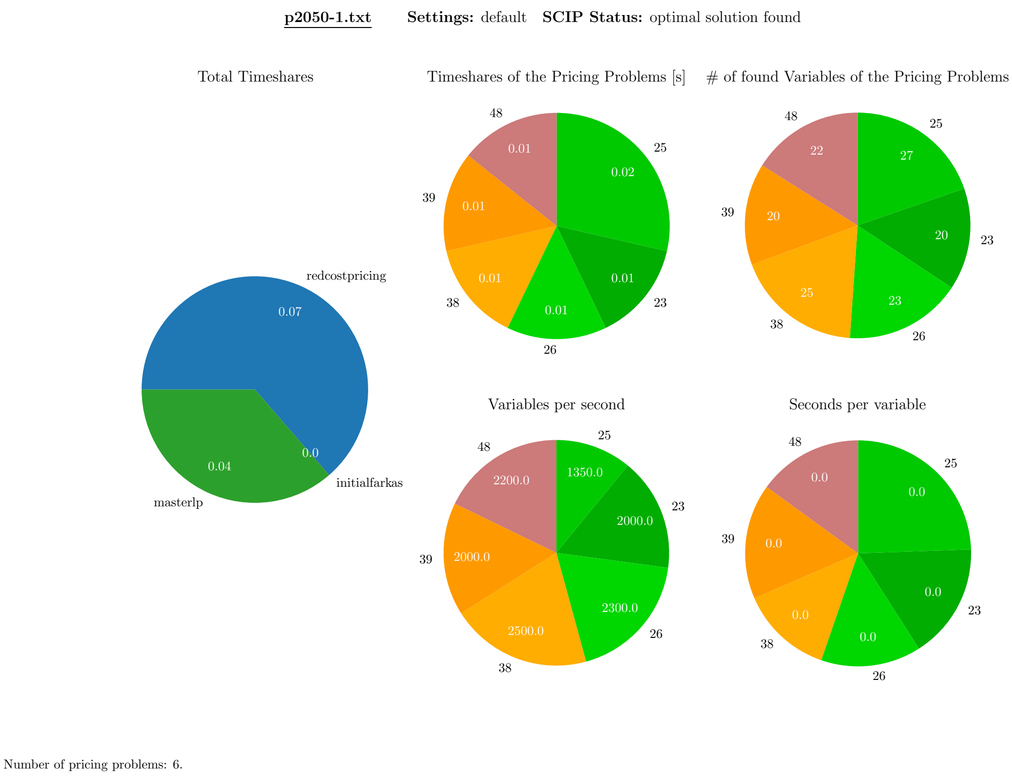


Figure 68: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/p2050-1.txt.default.time.pdf`

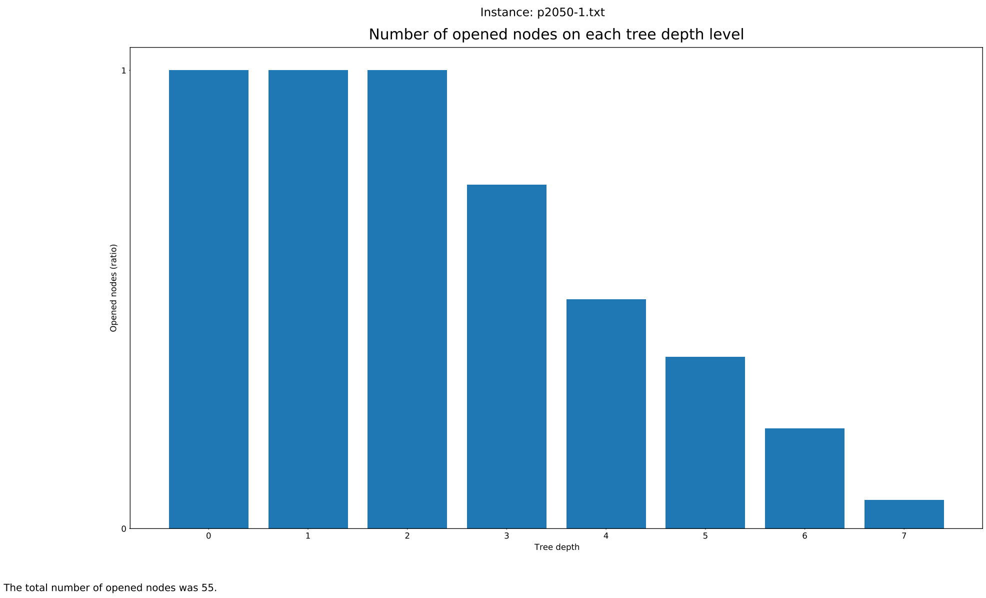


Figure 69: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/p2050-1.txt.default.tree.bar.pdf`

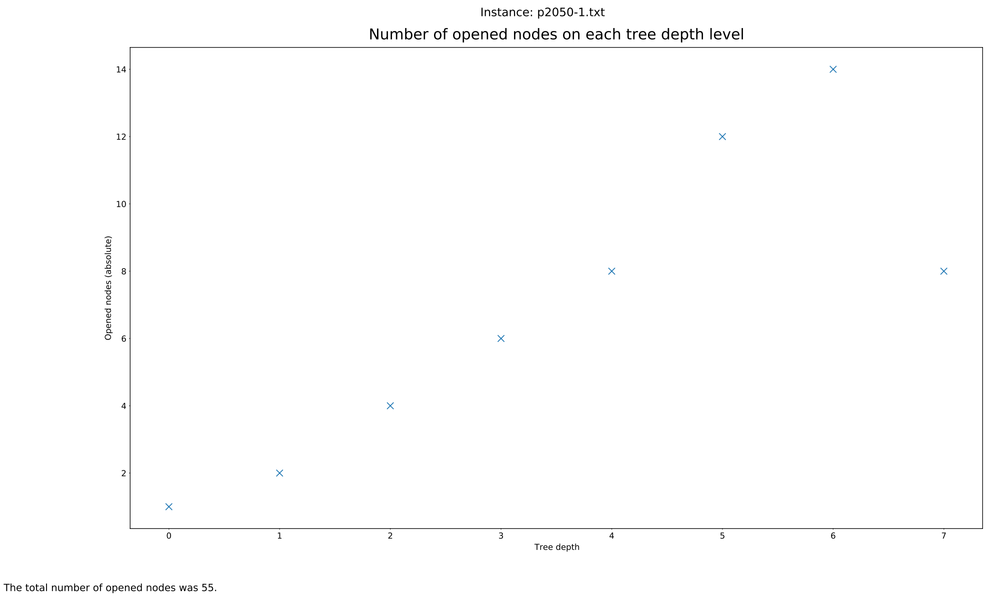


Figure 70: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/p2050-1.txt.default.tree.plot.pdf`

1.8 Instance: TEST0055

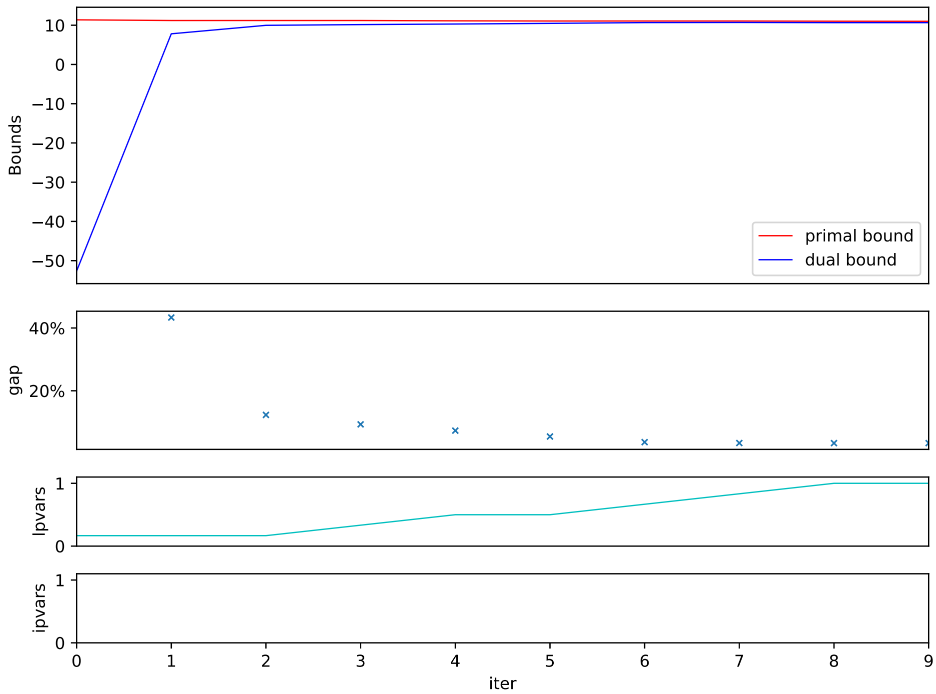


Figure 71: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/TEST0055.default.bounds.iter.pdf`

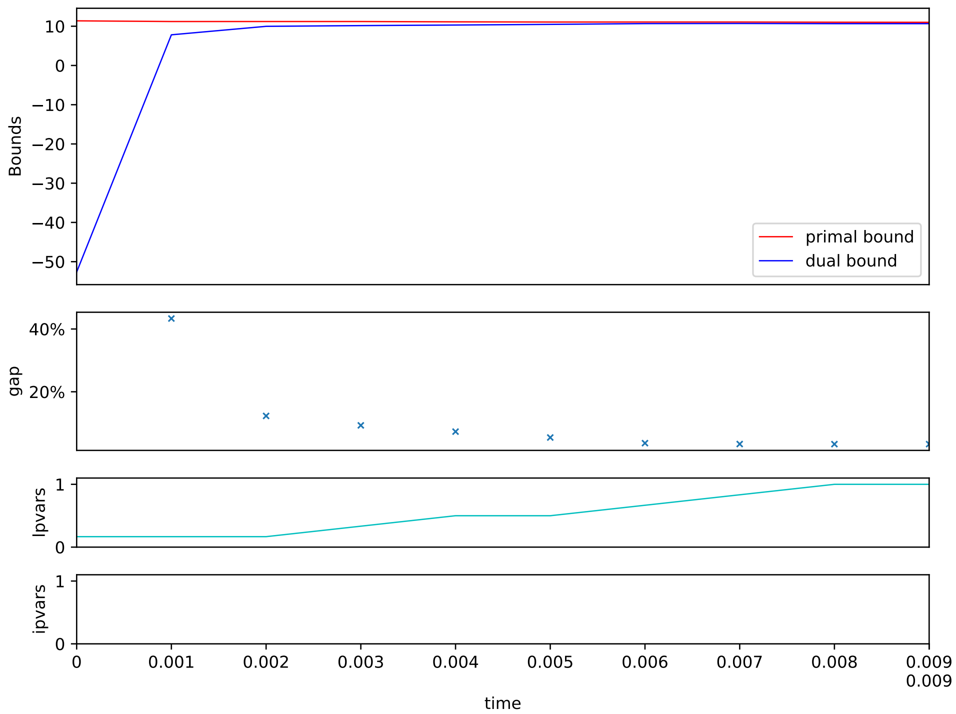


Figure 72: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/TEST0055.default.bounds.time.pdf`

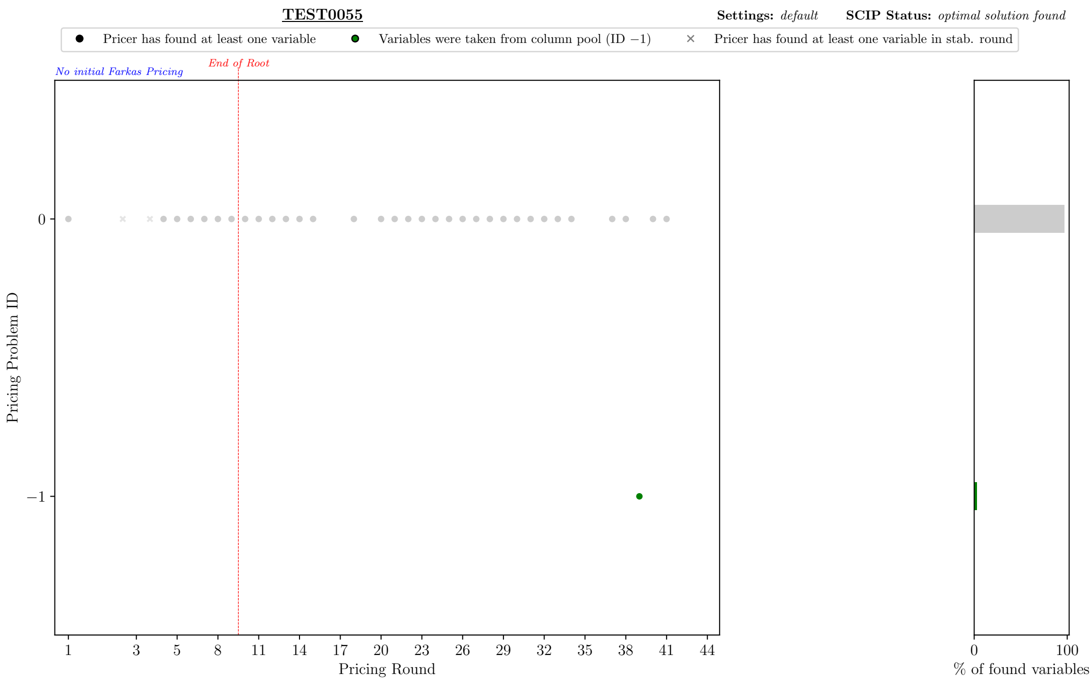


Figure 73: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/TEST0055.default.bubble.pdf`

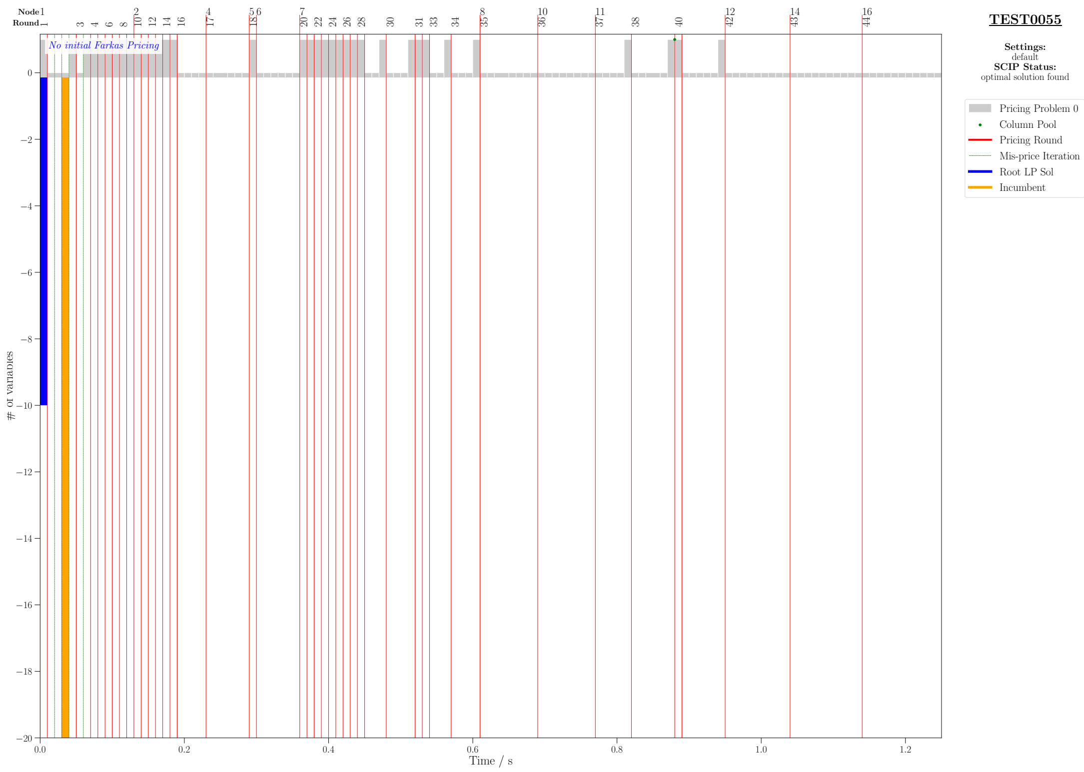


Figure 74: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/TEST0055.default.complete.pdf`

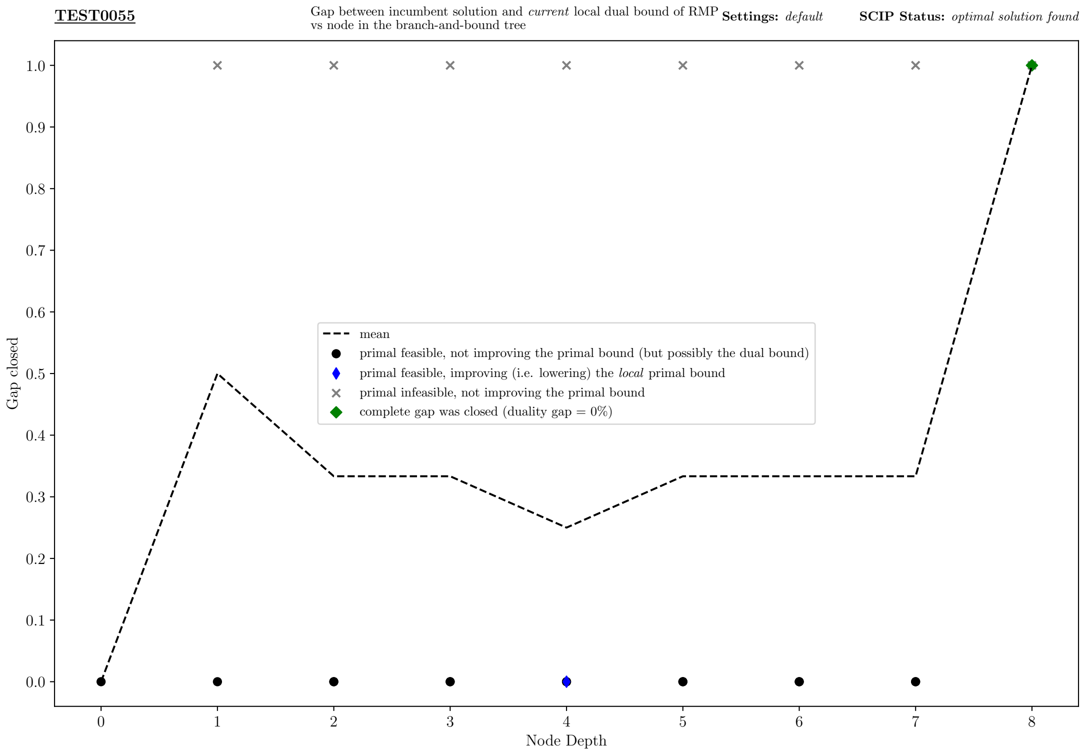


Figure 75: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/TEST0055.default.depth.pdf`

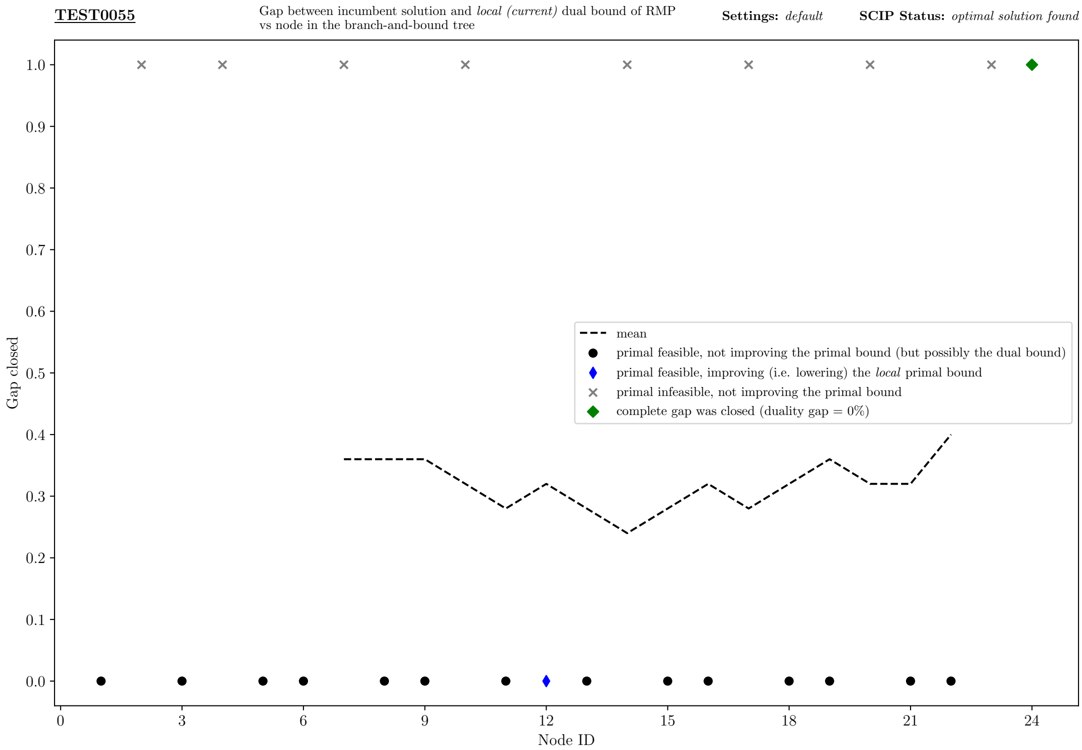


Figure 76: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/TEST0055.default.nodeID.pdf`

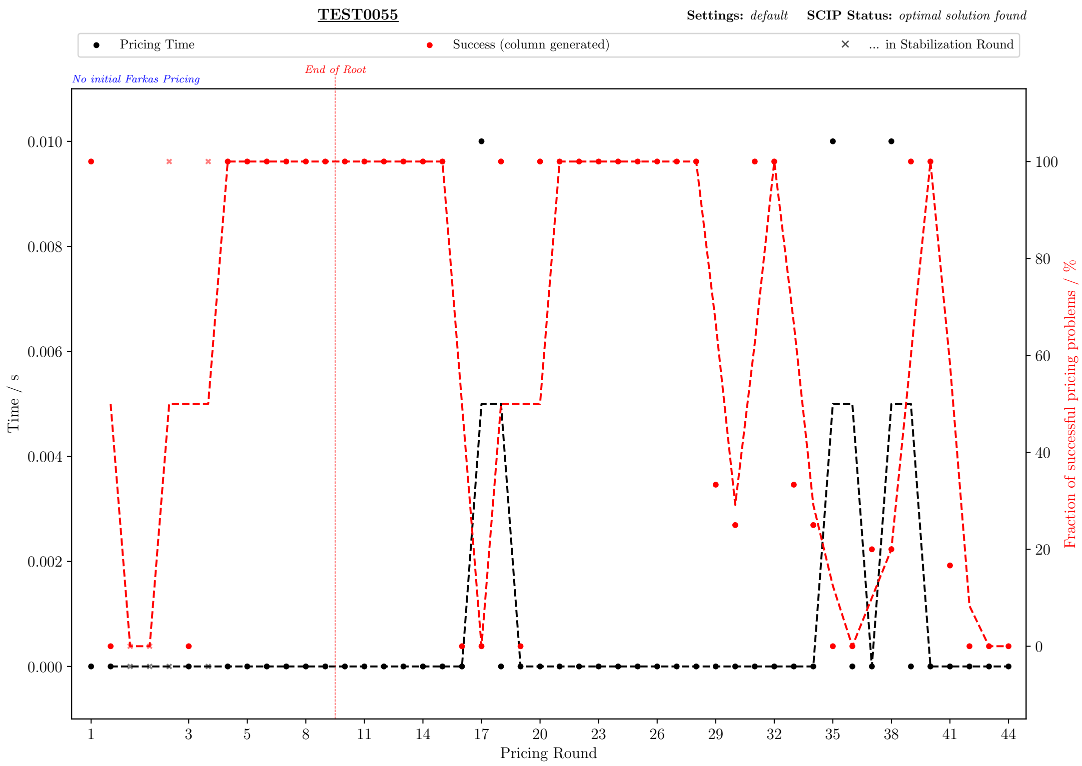


Figure 77: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/TEST0055.default.summary.pdf`

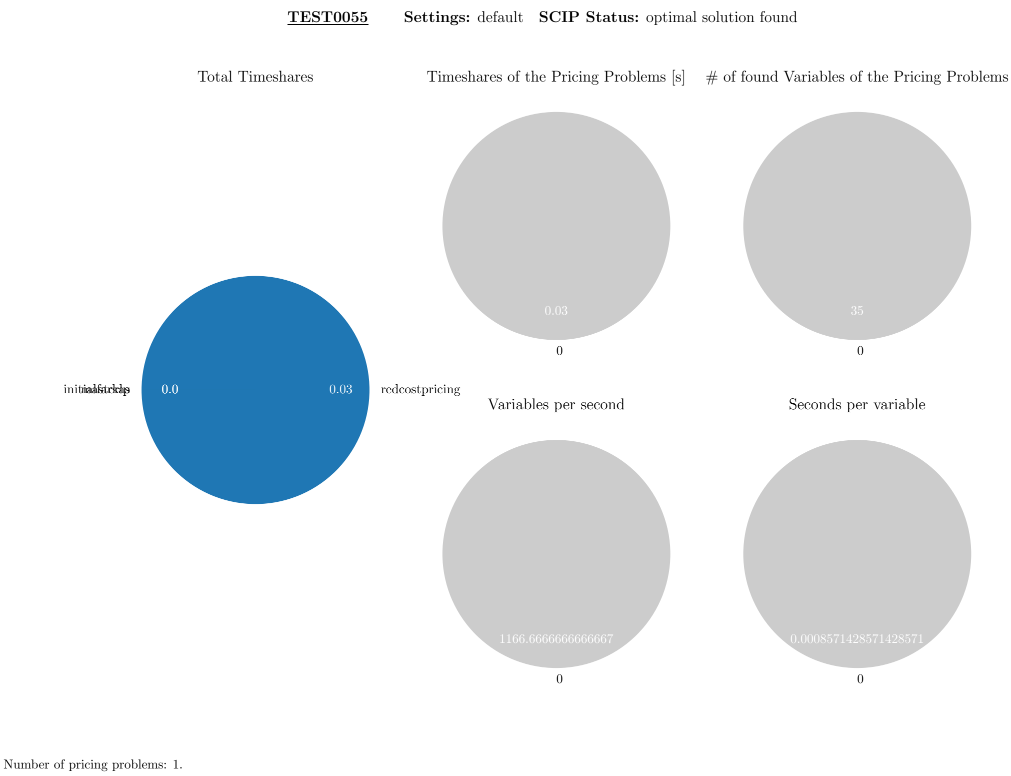
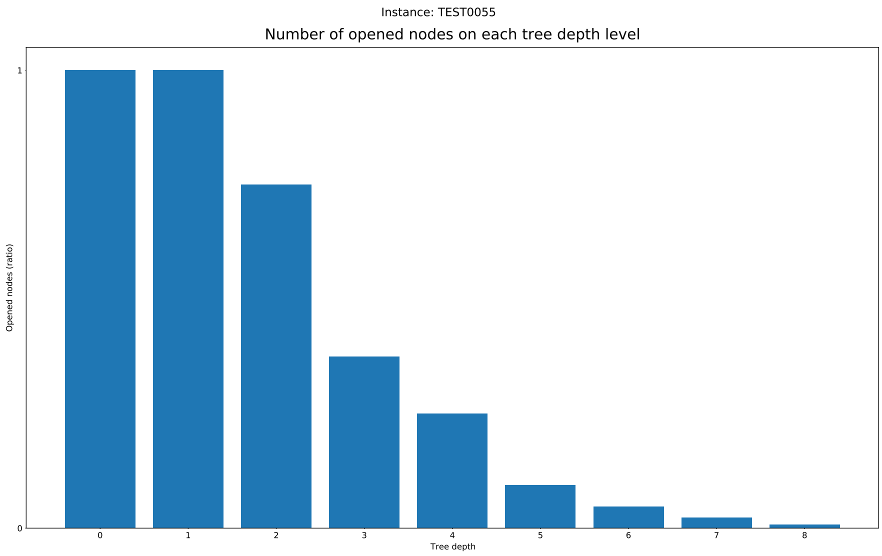
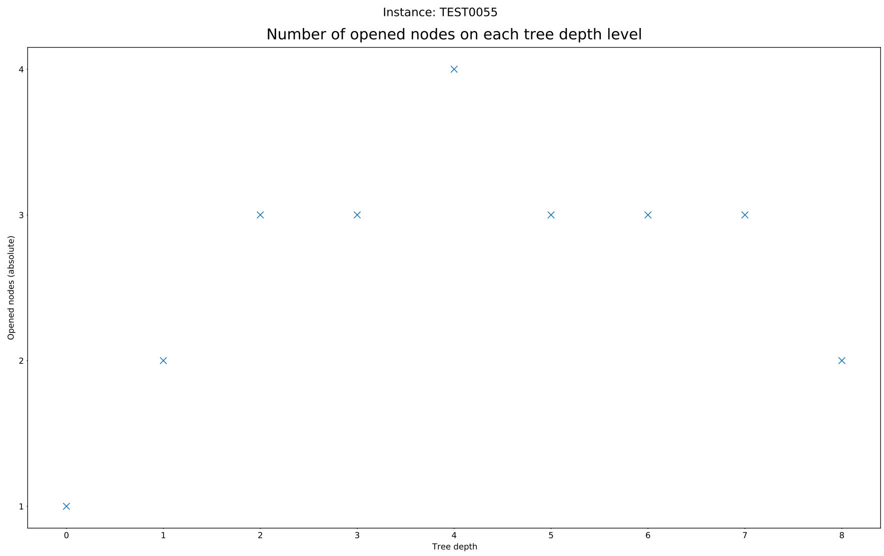


Figure 78: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/TEST0055.default.time.pdf`



The total number of opened nodes was 24.

Figure 79: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/TEST0055.default.tree.bar.pdf`



The total number of opened nodes was 24.

Figure 80: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/TEST0055.default.tree.plot.pdf`

1.9 Instance: TEST0059

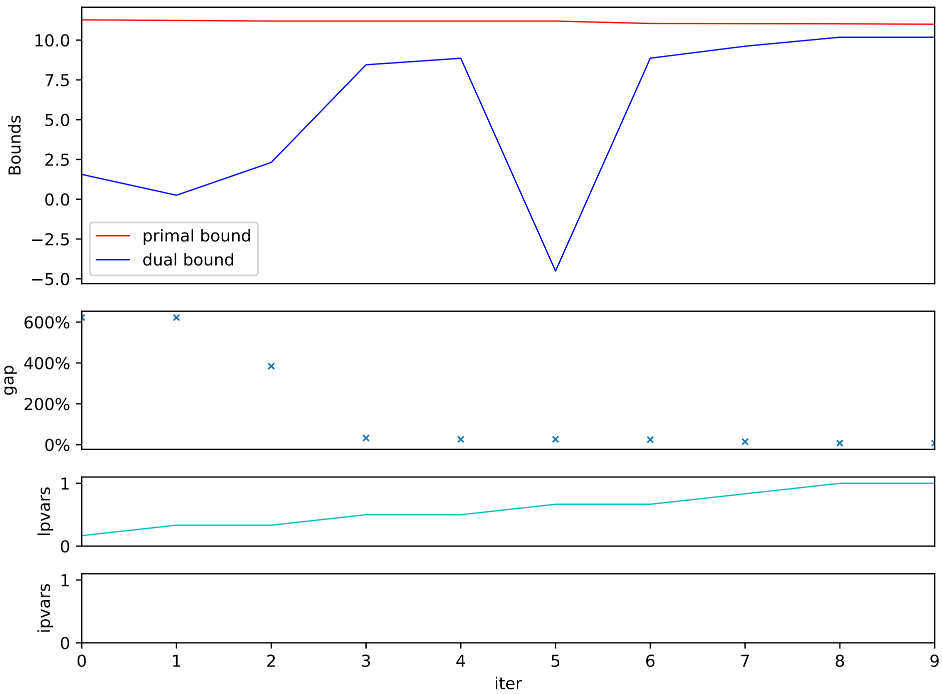


Figure 81: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/TEST0059.default.bounds.iter.pdf`

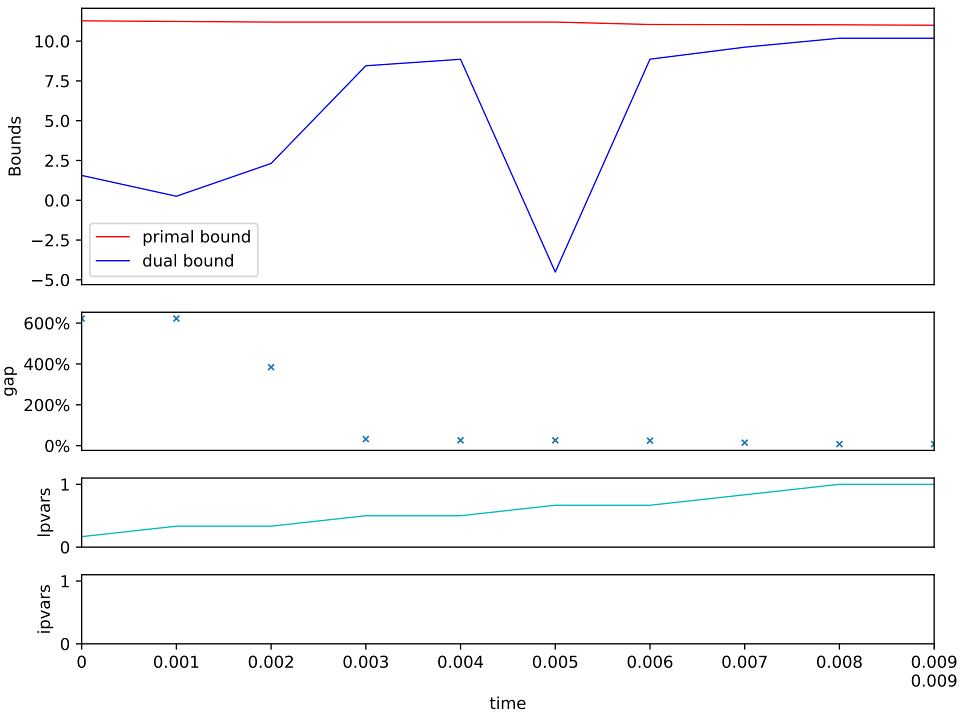


Figure 82: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/TEST0059.default.bounds.time.pdf`

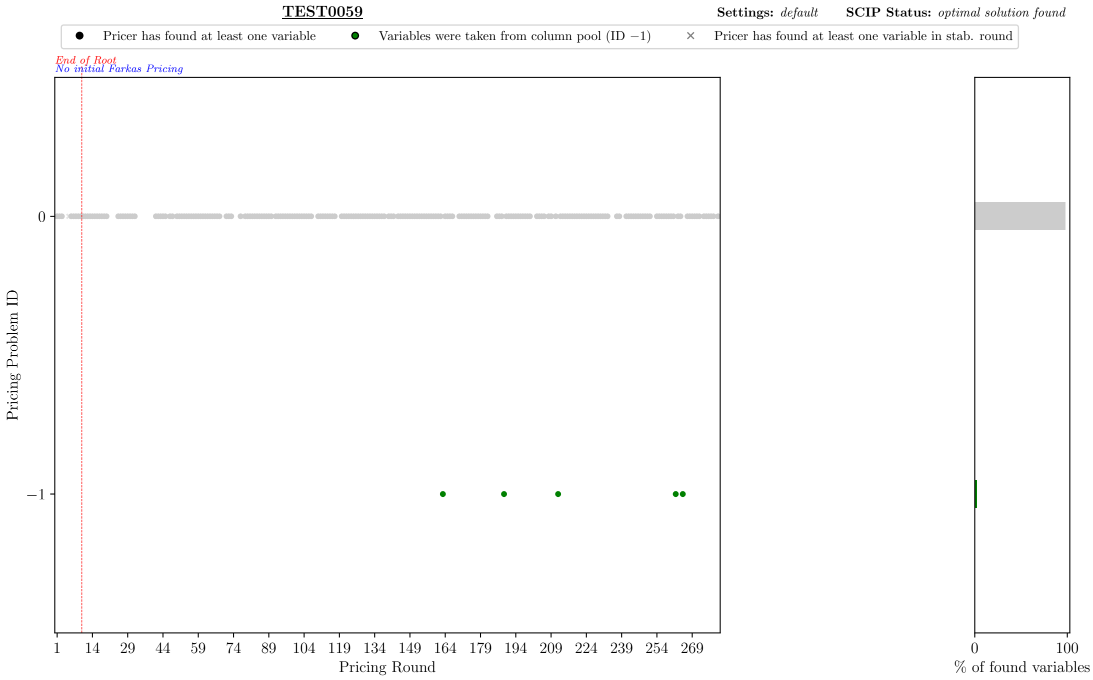


Figure 83: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/TEST0059.default.bubble.pdf`

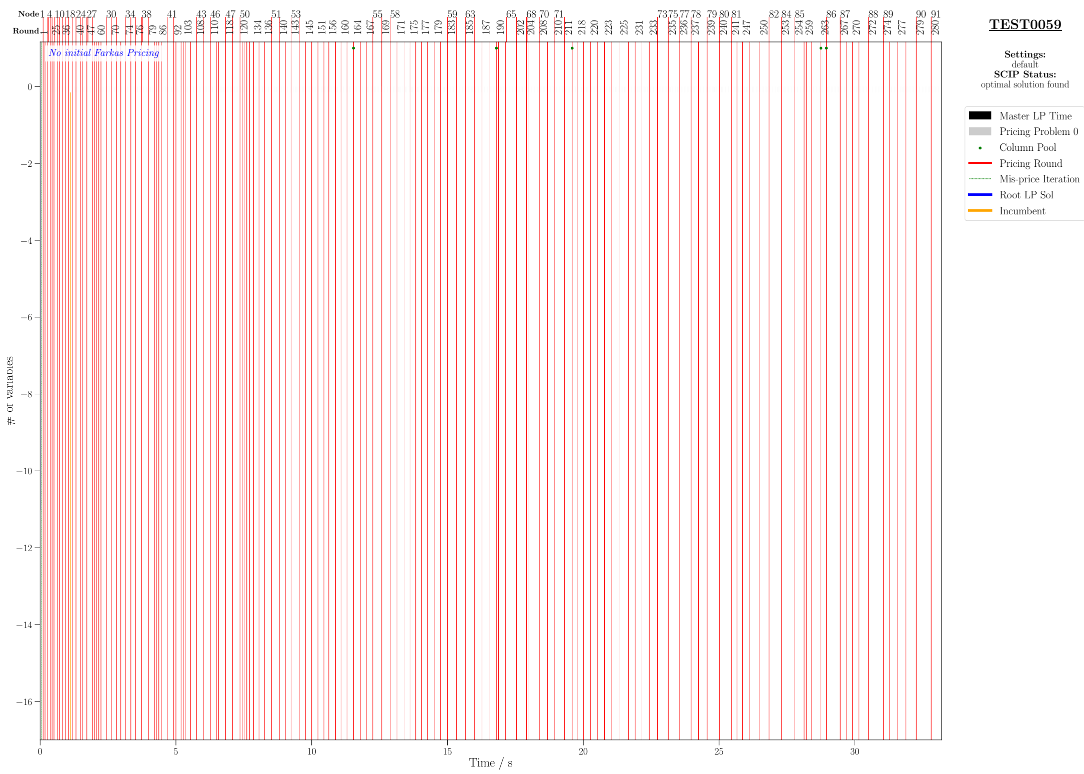


Figure 84: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/TEST0059.default.complete.pdf`

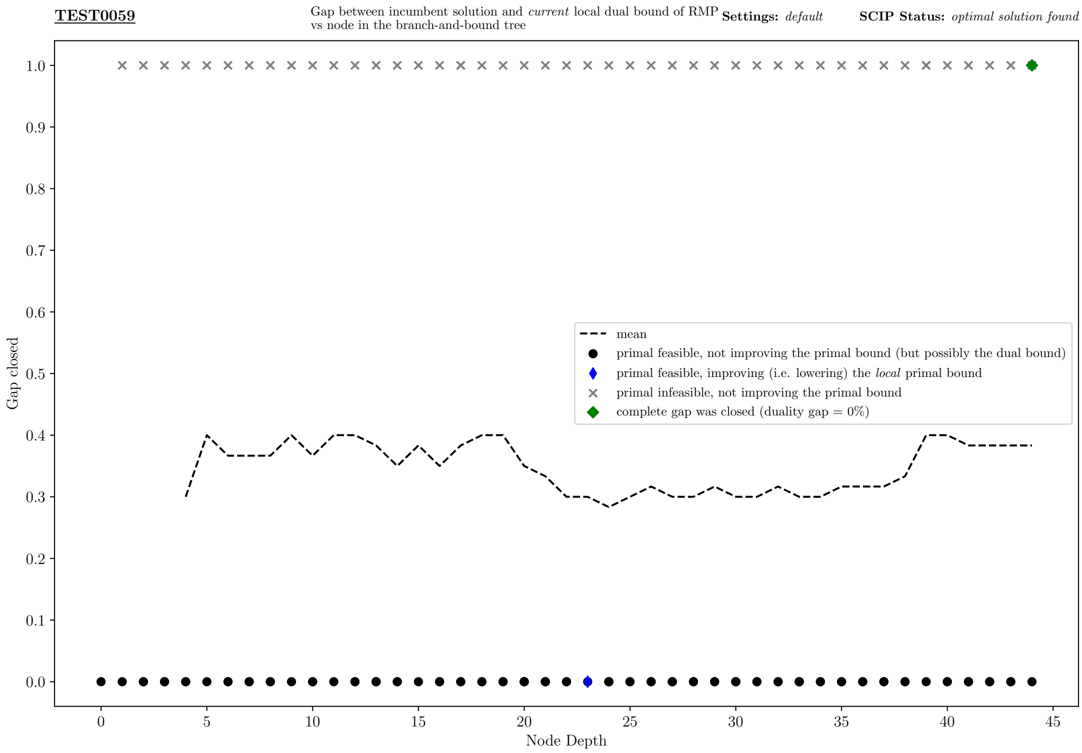


Figure 85: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/TEST0059.default.depth.pdf`

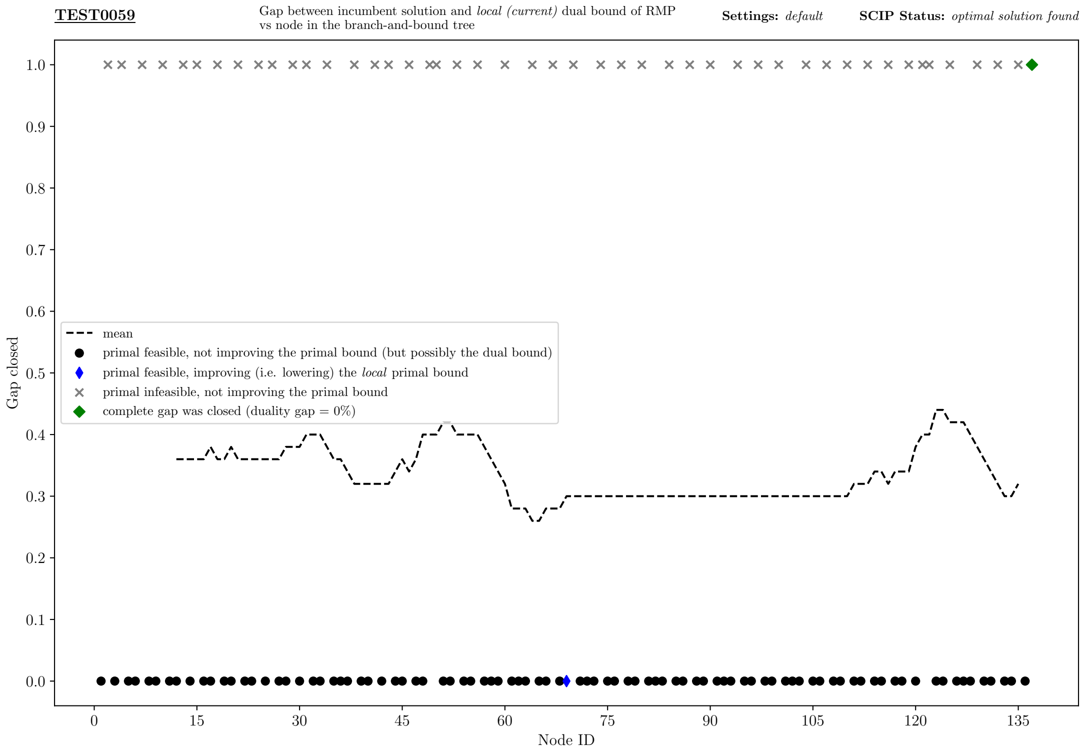


Figure 86: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/TEST0059.default.nodeID.pdf`

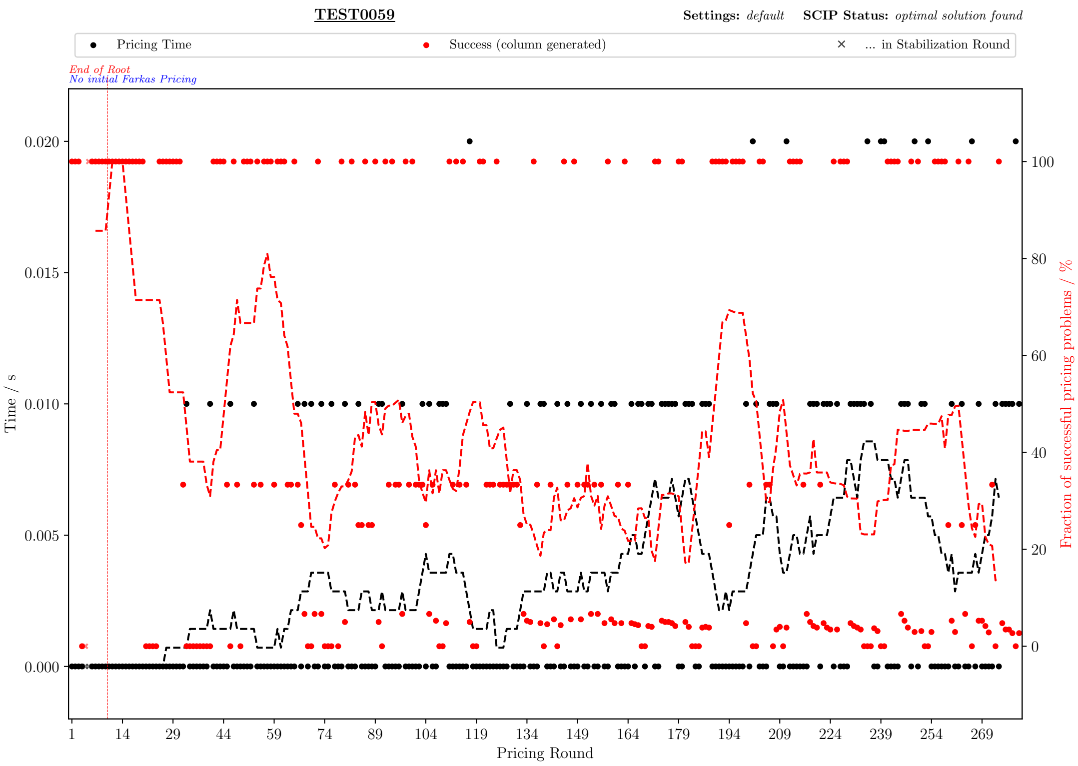


Figure 87: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/TEST0059.default.summary.pdf`

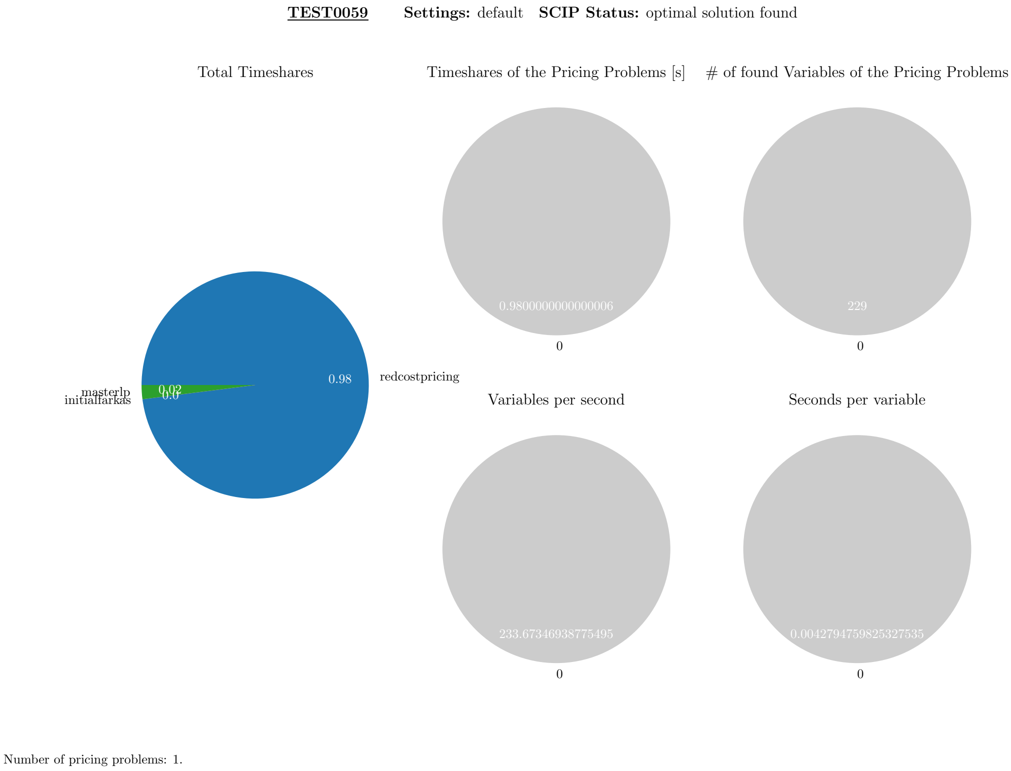
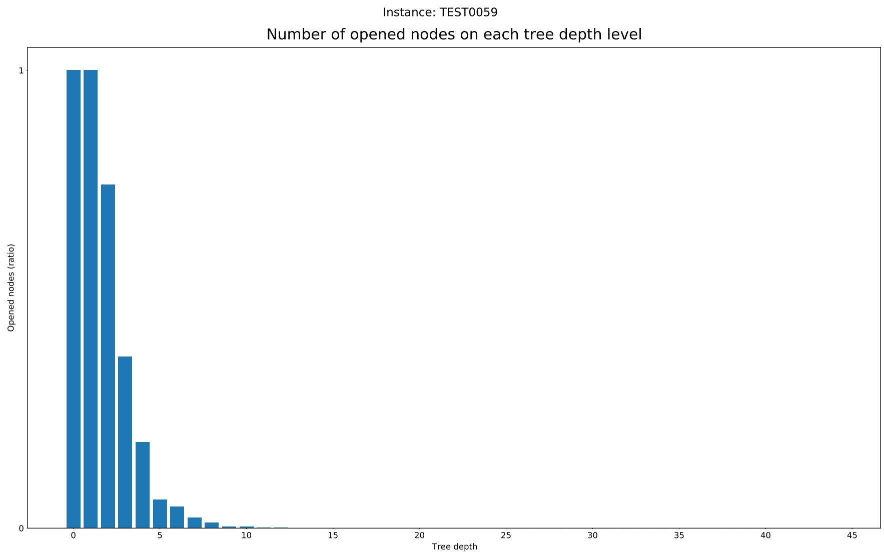


Figure 88: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/TEST0059.default.time.pdf`



The total number of opened nodes was 137.

Figure 89: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/TEST0059.default.tree.bar.pdf`

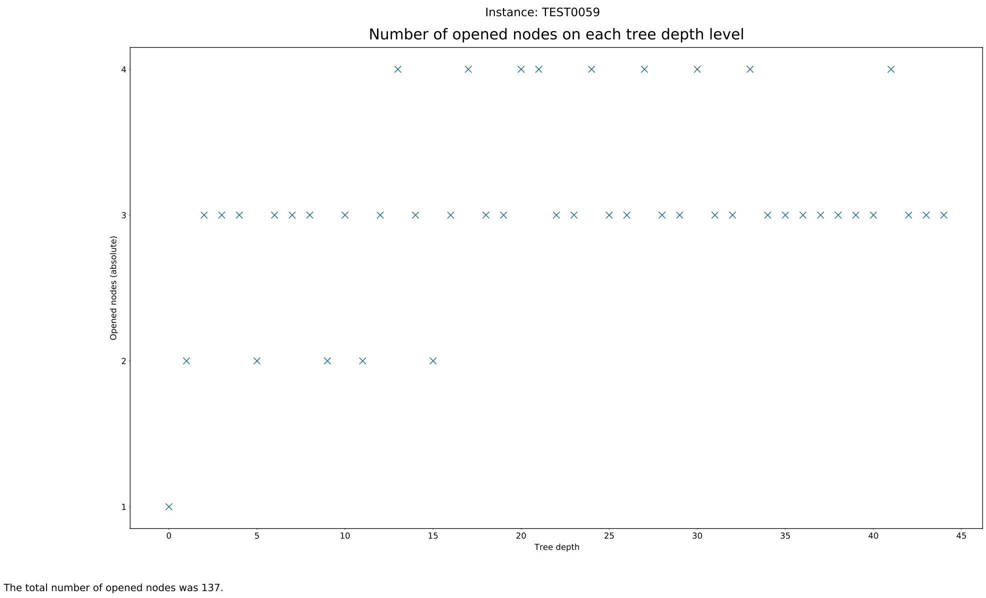


Figure 90: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/TEST0059.default.tree.plot.pdf`

1.10 Instance: gap4_2.txt

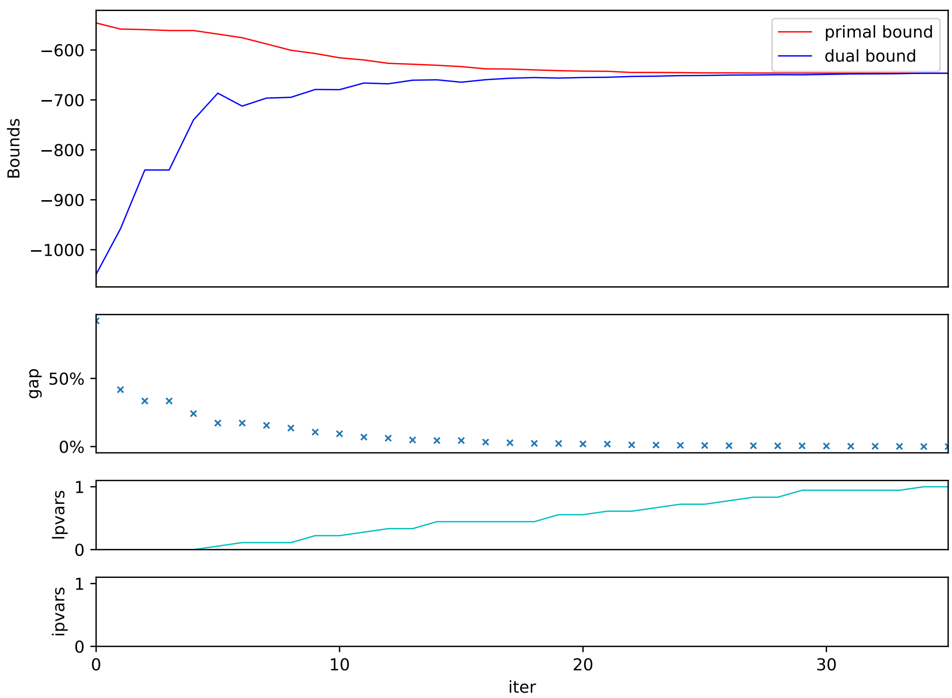


Figure 91: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/gap4_2.txt.default.bounds.iter.pdf`

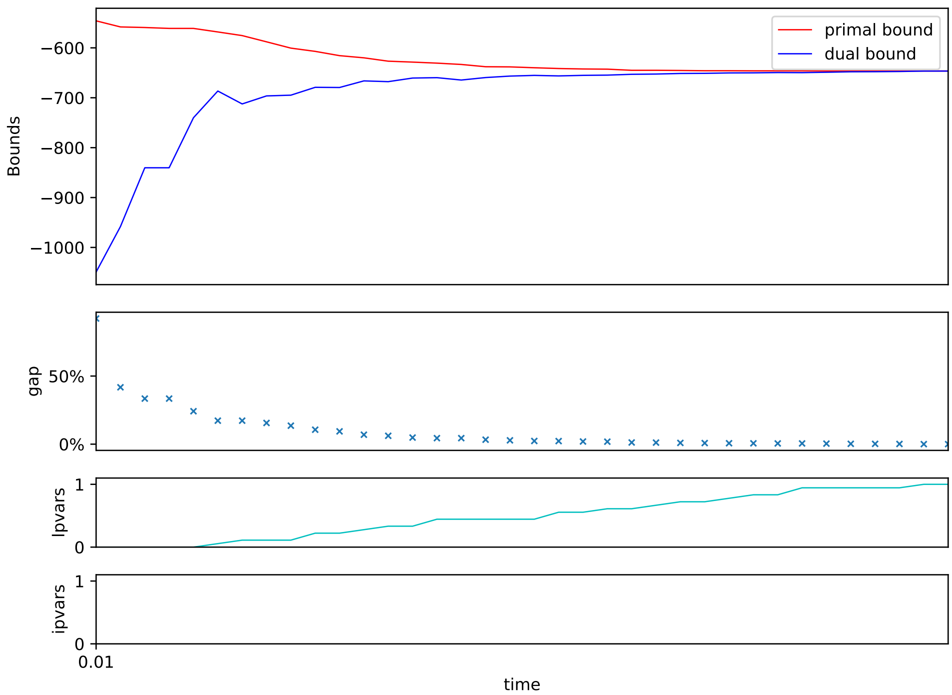


Figure 92: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/gap4_2.txt.default.bounds.time.pdf`

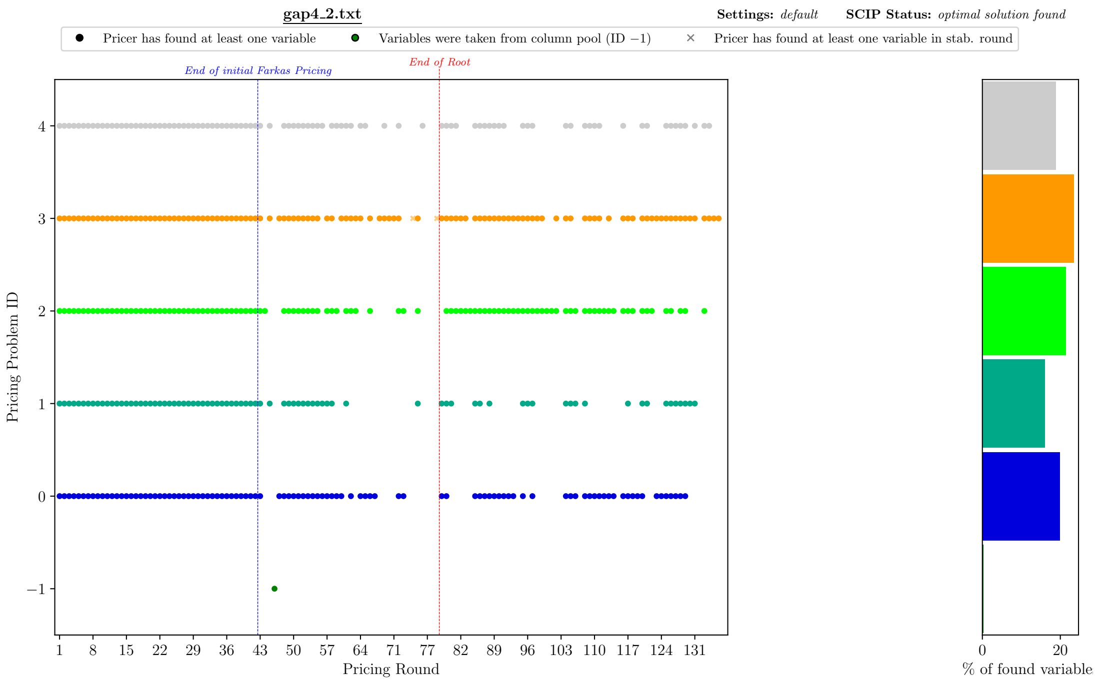


Figure 93: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/gap4_2.txt.default.bubble.pdf`

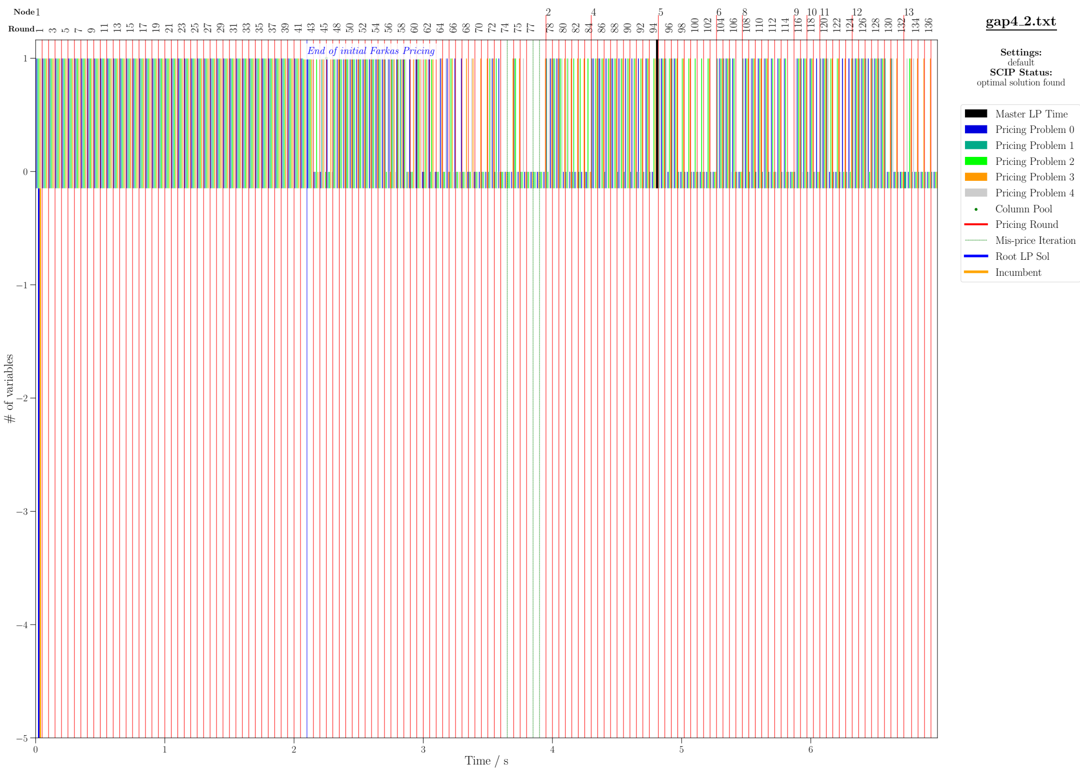


Figure 94: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/gap4_2.txt.default.complete.pdf`

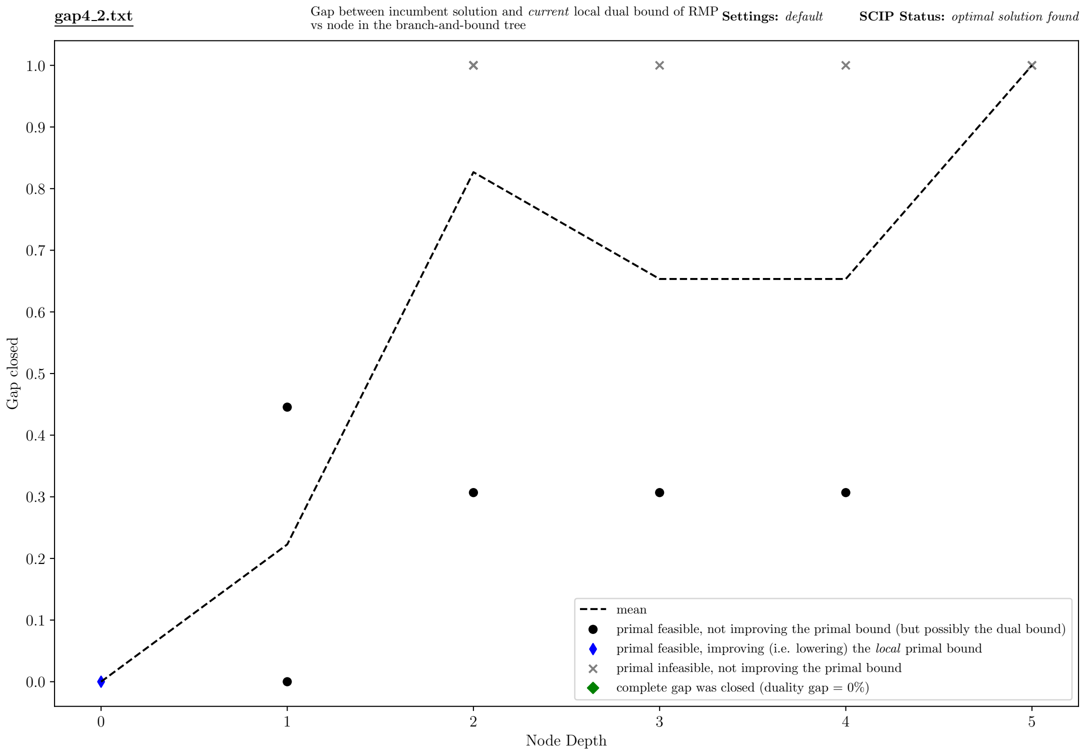


Figure 95: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/gap4_2.txt.default.depth.pdf`

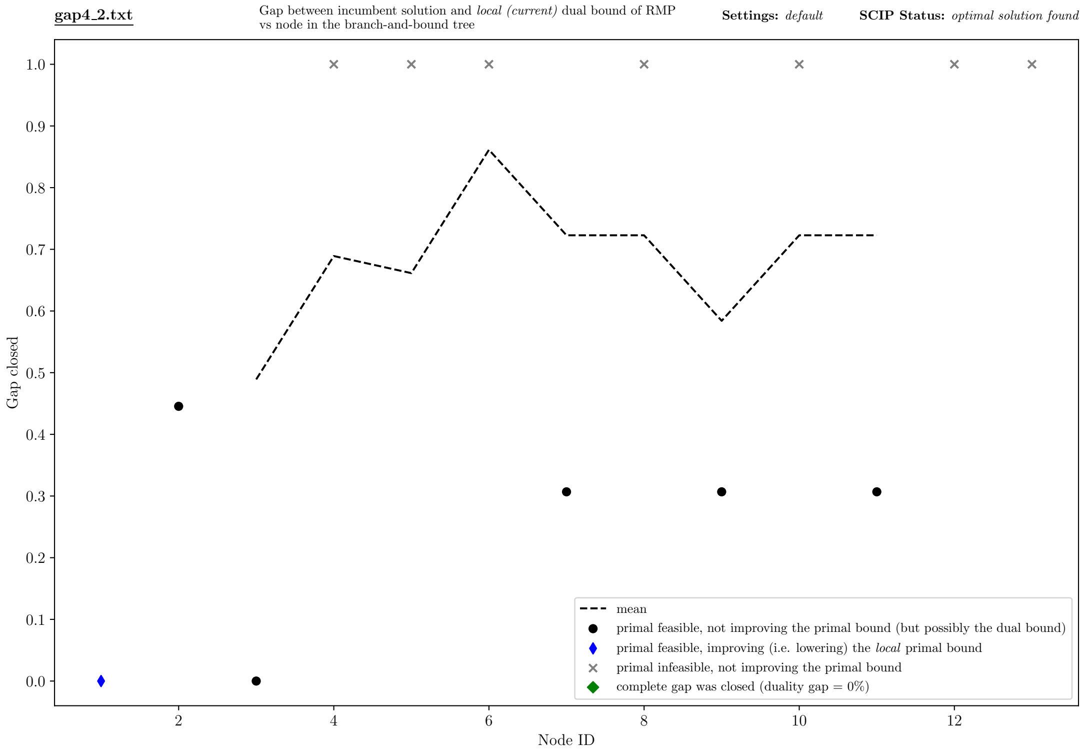


Figure 96: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/gap4_2.txt.default.nodeID.pdf`

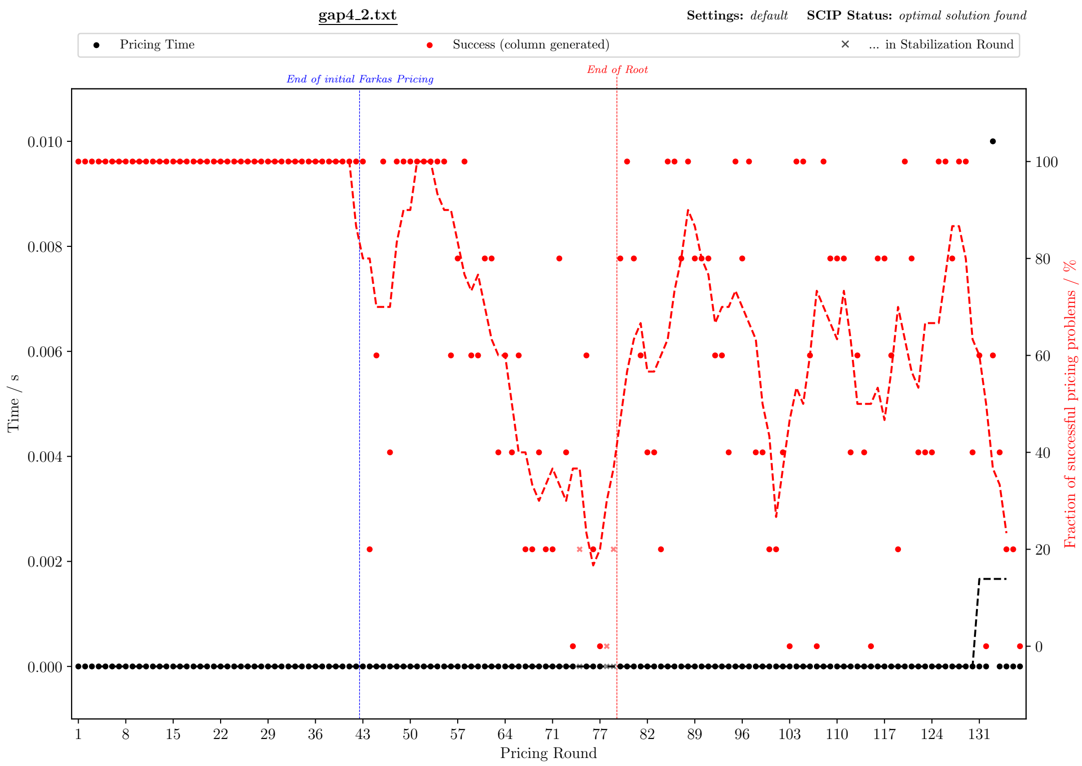


Figure 97: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/gap4_2.txt.default.summary.pdf`

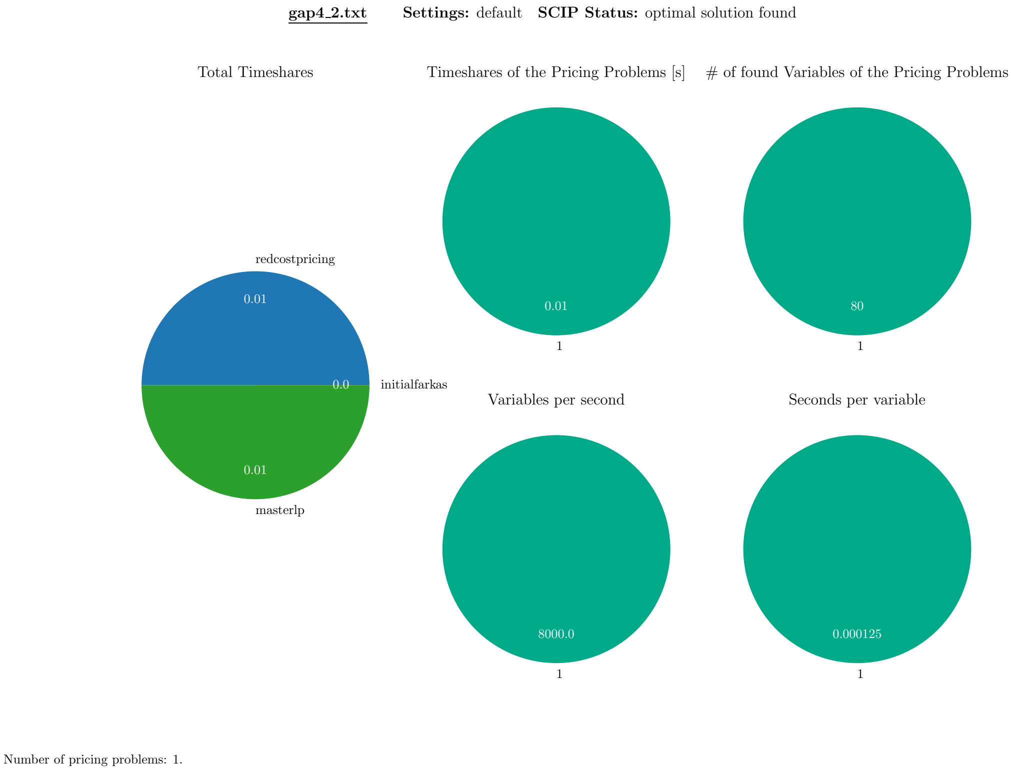
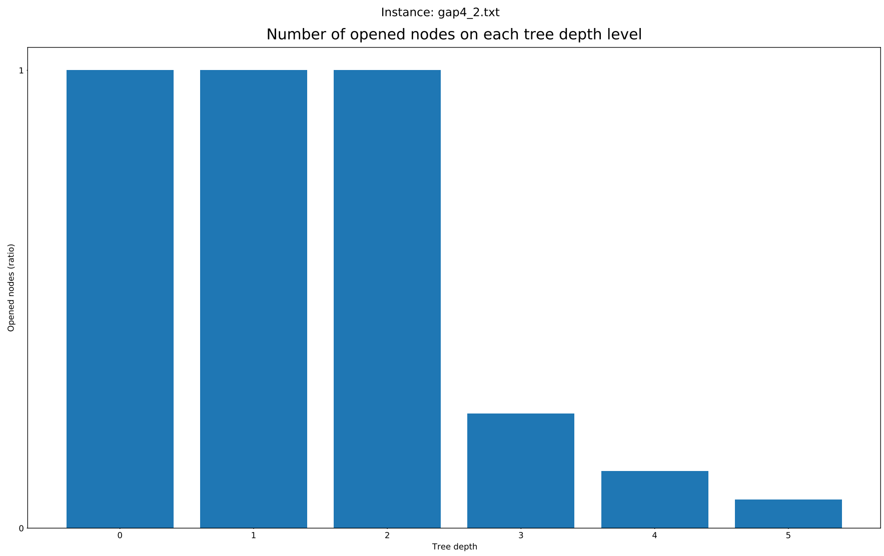


Figure 98: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time (11° of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/gap4_2.txt.default.time.pdf`



The total number of opened nodes was 13.

Figure 99: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/gap4_2.txt.default.tree.bar.pdf`

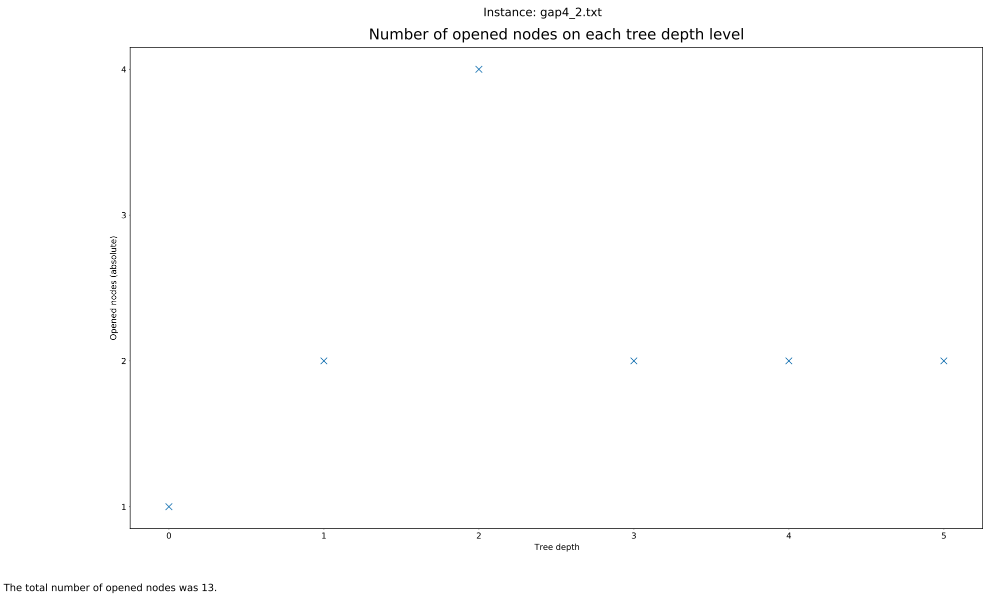


Figure 100: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/gap4_2.txt.default.tree.plot.pdf`

1.11 Instance: gap8_4.txt

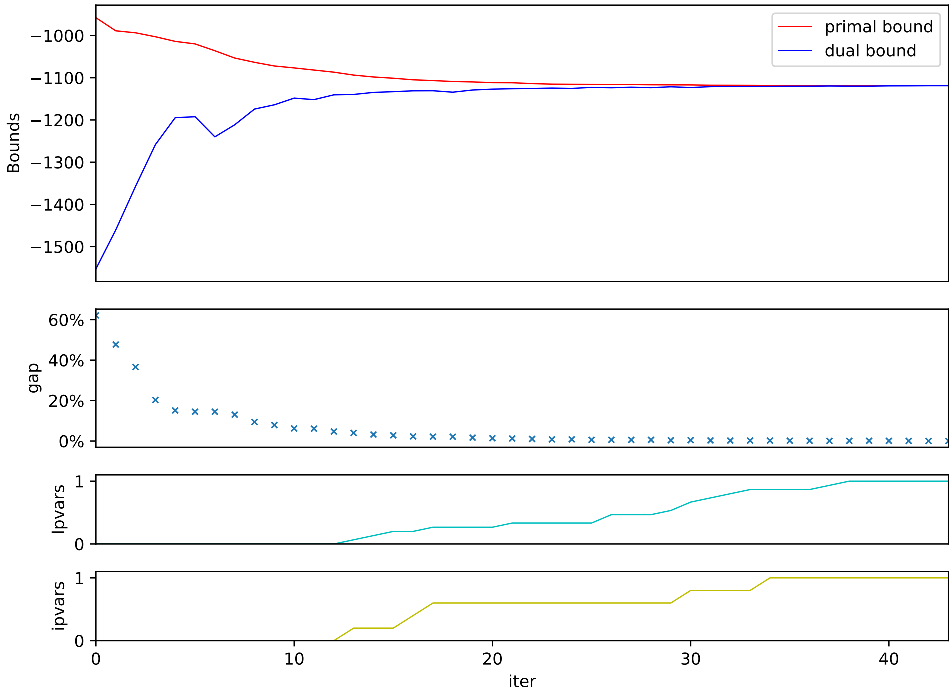


Figure 101: The same plot as the bounds time plot (see below), but with pricing iterations in the root node instead of the time spent there on the x-axis.
Visualization Path: `plots/bounds/gap8_4.txt.default.bounds.iter.pdf`

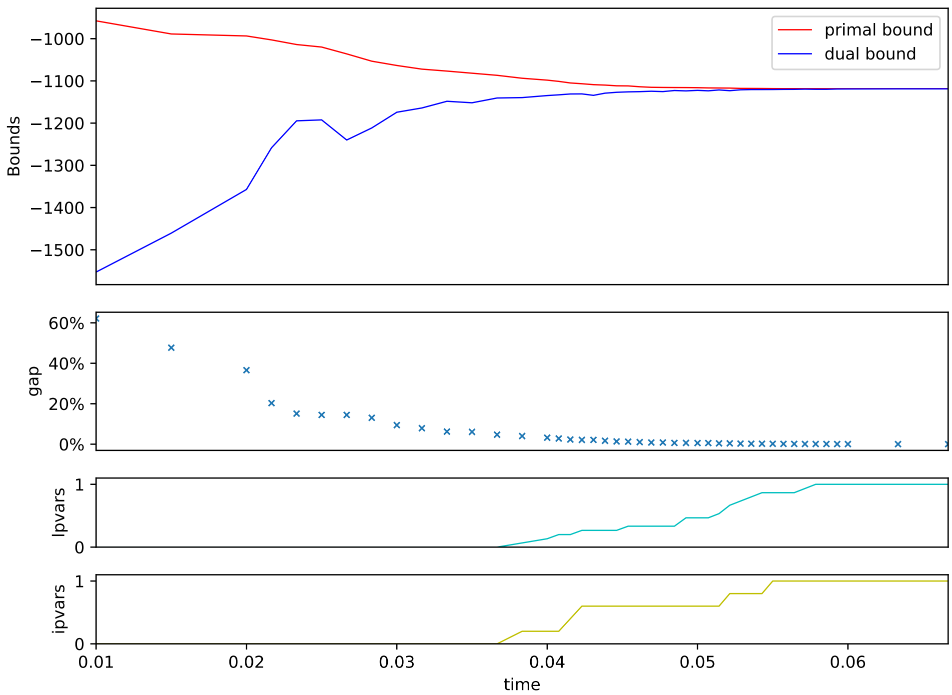


Figure 102: The top subplot shows the development of the primal and dual bounds in the RMP during the pricing in the root node as given by the table root bounds" printed by GCG. Every change represents a pricing iteration and the resulting changes to the bounds. The bounds are complemented by a newly created gap plot, which will be explained in Section sec:tgpp. The other two subplots illustrate the point in time in the pricing at which the columns that are finally in the basis are generated.
Visualization Path: `plots/bounds/gap8_4.txt.default.bounds.time.pdf`

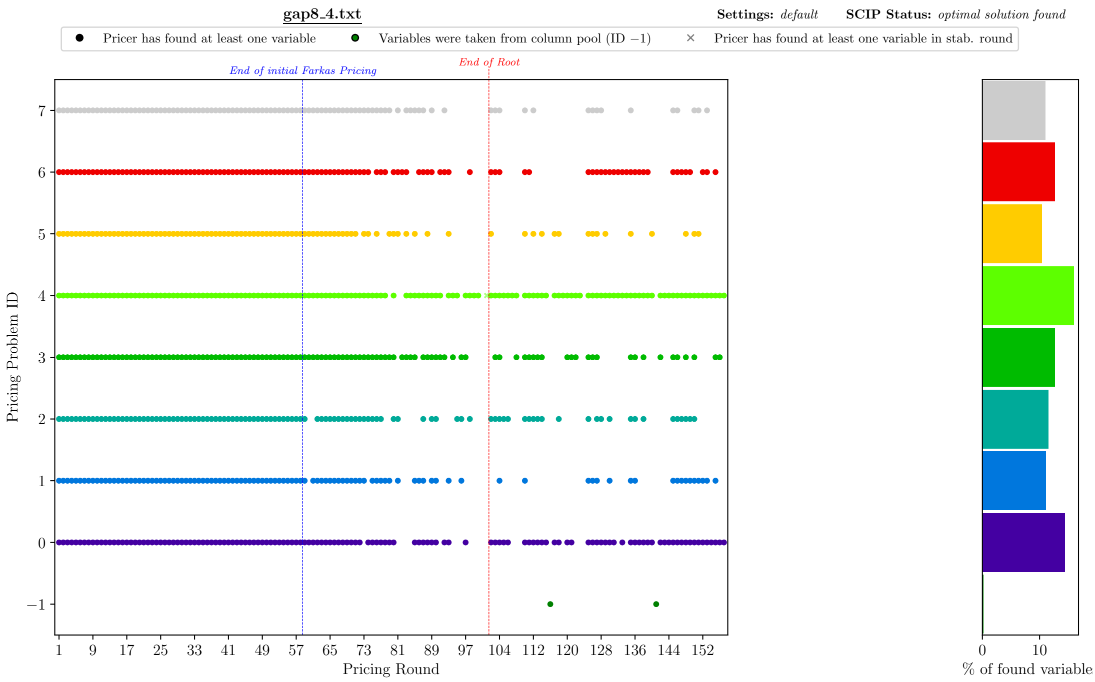


Figure 103: In this visualization, one can see all pricing problems listed vertically along the y-axis. Then, in the left subfigure, they are shown against the pricing rounds on the x-axis. Every time the pricer yielded at least one variable resulting from a pricing problem, a dot is printed in the round where it was generated. This results in the ability to not only see the sensibility of each pricing problem, but also in which rounds what pricing problem performed best. The subplot on the right-hand side shows how many percent of the variables were generated by which problem.
Visualization Path: `plots/pricing/gap8_4.txt.default.bubble.pdf`

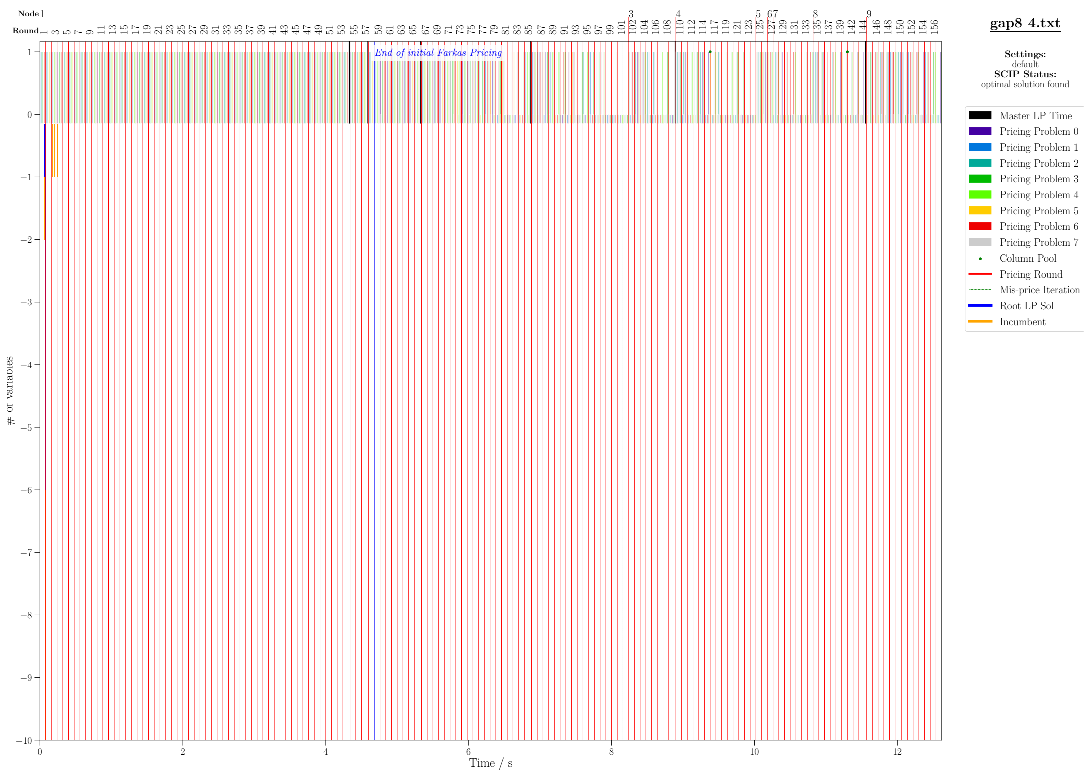


Figure 104: This Plot shows how many variables were generated in a certain pricing round in which time for all nodes of the Branch and Bound tree. The node numbers are shown above the plot and the rounds are in the line below that. Each bar represents the iteration of one pricing problem. Note that the numbers of the pricing problems can have gaps in between, since they could have been aggregated prior to the pricing. Whether those variables are useful is shown by all bars that are below zero, as they mean that the variables of that pricing iteration are in the optimal solution of the Root LP (Root LP Sol) or IP (Incumbent). Finally, the dots show how many columns are taken from the column pool.
Visualization Path: `plots/pricing/gap8_4.txt.default.complete.pdf`

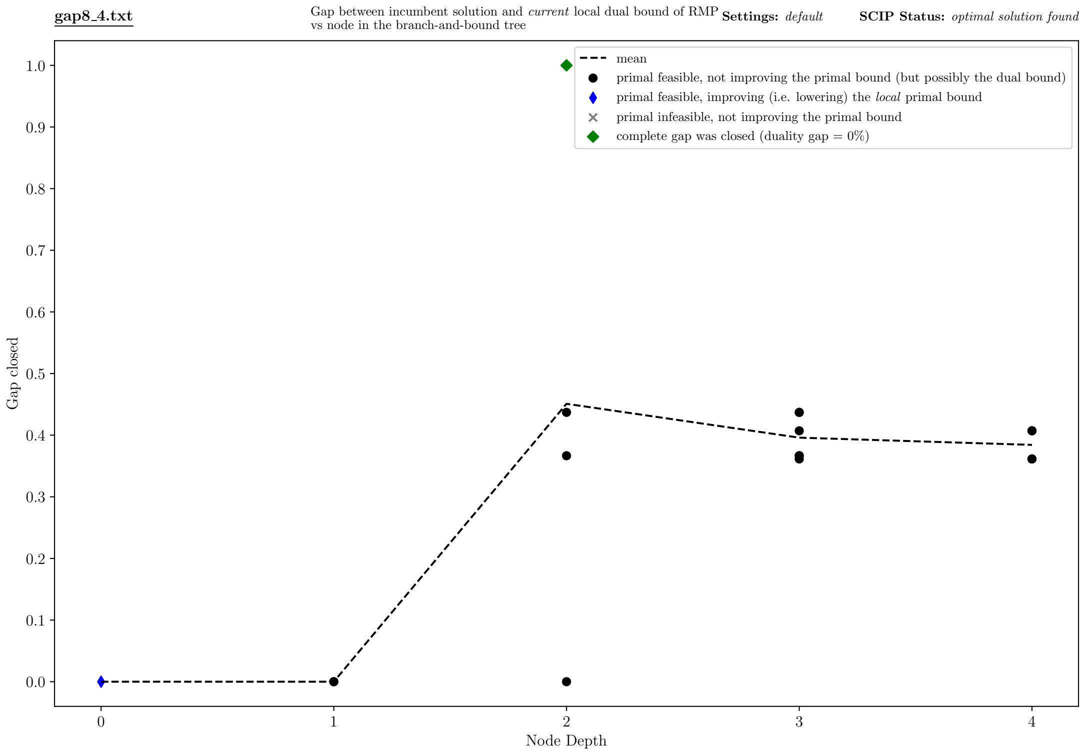


Figure 105: This figure illustrates how the gap develops along the depth of the branching tree. Each dot represents the gap as given by the primal and dual bounds in this specific node as given by the GCG root bounds" table (just like in the bounds plotter). This node is located on the tree depth that can be read on the x-axis, such that for each x-coordinate, at most 2^x points can exist. Furthermore, a plot of the mean is given.
Visualization Path: `plots/pricing/gap8_4.txt.default.depth.pdf`

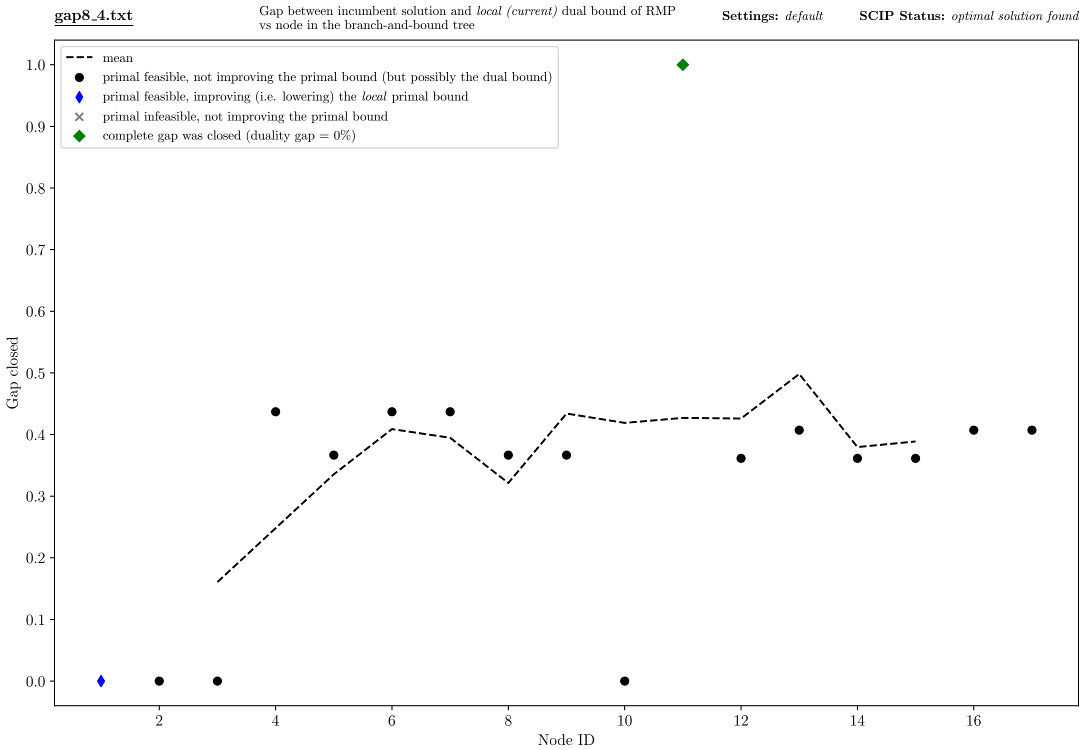


Figure 106: The Node ID plot is similar to the Depth Plot. Instead of the depth in the branch-and-bound tree, we now have the node ID. This leads to the fact that one can see behavior that is not dependent of the depth, but of the time progression during the branching.
Visualization Path: `plots/pricing/gap8_4.txt.default.nodeID.pdf`

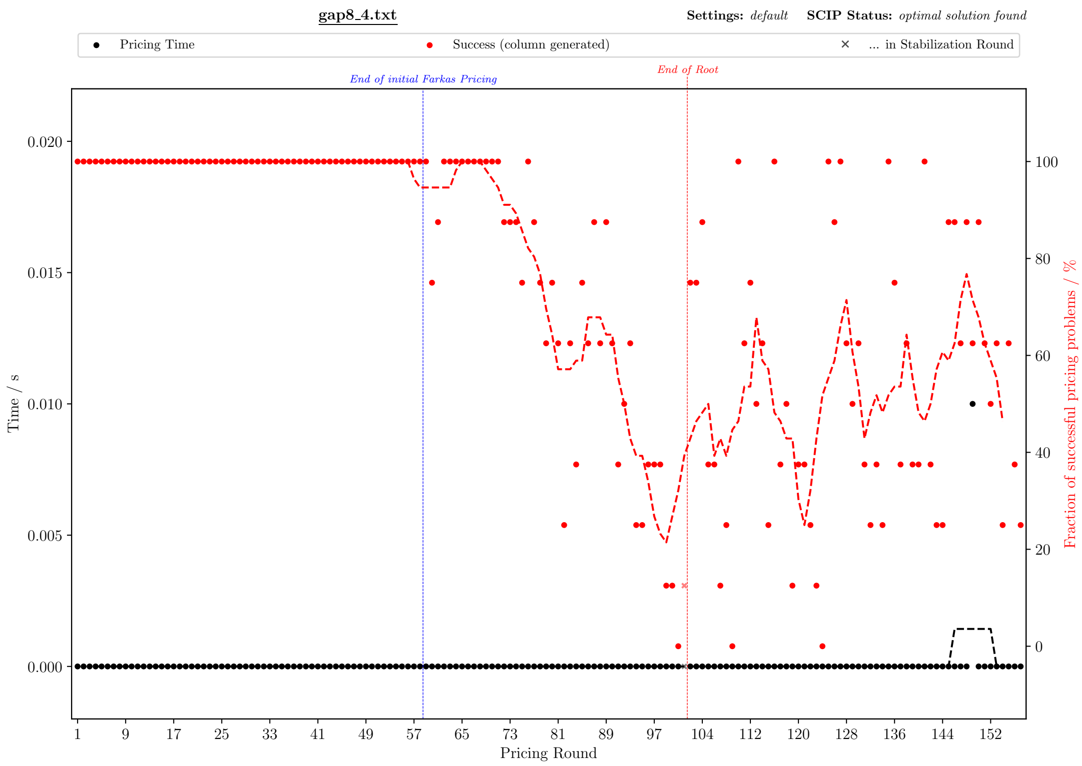


Figure 107: The summary plot aims to illustrate the same thing as the complete plot". The end of the root node, which is treated in deeper detail in the Bounds Plot, is marked by a red line. The plot consists of two different y-axes, one representing the time (in seconds) needed for the pricing and the other the fraction of pricing problems that generated variables. This leads to the ability to identify pricing rounds that ran for a long time and see when and how many pricing problems were successful.
Visualization Path: `plots/pricing/gap8_4.txt.default.summary.pdf`

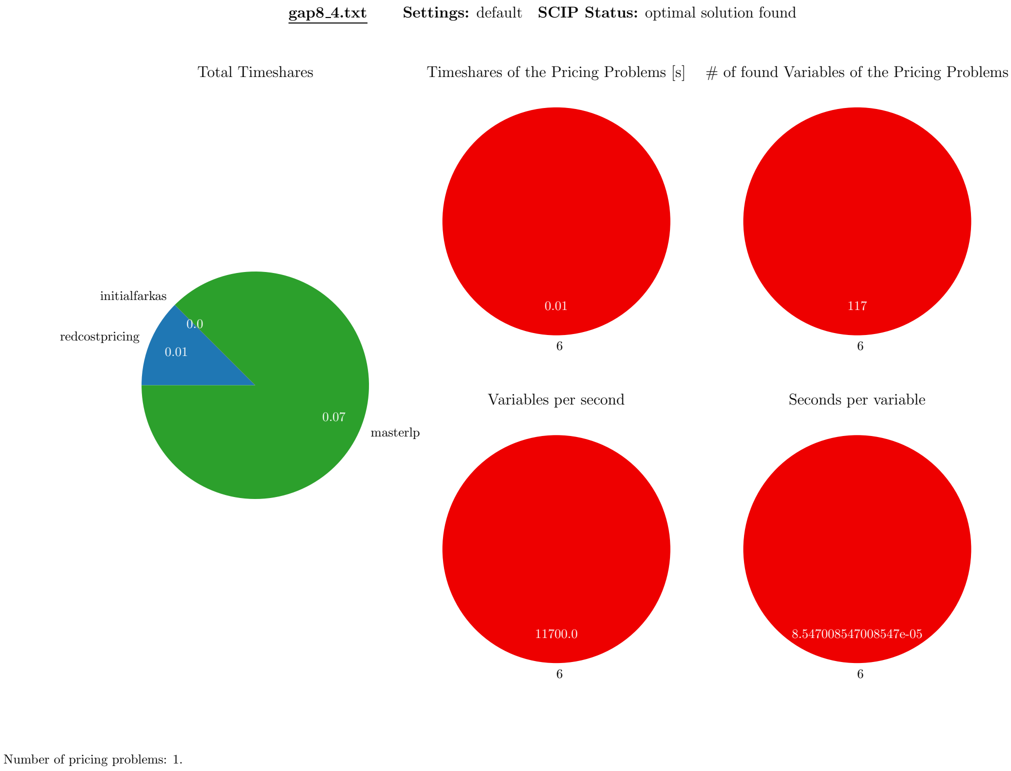


Figure 108: The Pricing Time Statistics include four pie charts. The first one shows how much of the runtime was needed in the reduced cost pricing, the master LP and during the initial Farkas. The upper center one shows the relative (and, inside the slices, absolute) time needed by each pricing problem that took at least $\frac{11}{360}$ of the total pricing time ($11°$ of the pie, the last degree where the absolute numbers inside the slices are still readable). Note that if no absolute numbers are needed, but only the highest possible amount of slices (pricing problems) should be shown, the `--short-times` argument can be set. The pie chart to the upper right shows how many columns were generated by each pricing problem and the ratio between the upper right and the upper center, i.e. the variables per second, is shown in the lower left, illustrating which pricing problem yielded the most variables for the RMP. Finally, in the course of this thesis, an additional subplot that illustrates the seconds needed by each pricing problem to generate a variable was added.
Visualization Path: `plots/pricing/gap8_4.txt.default.time.pdf`

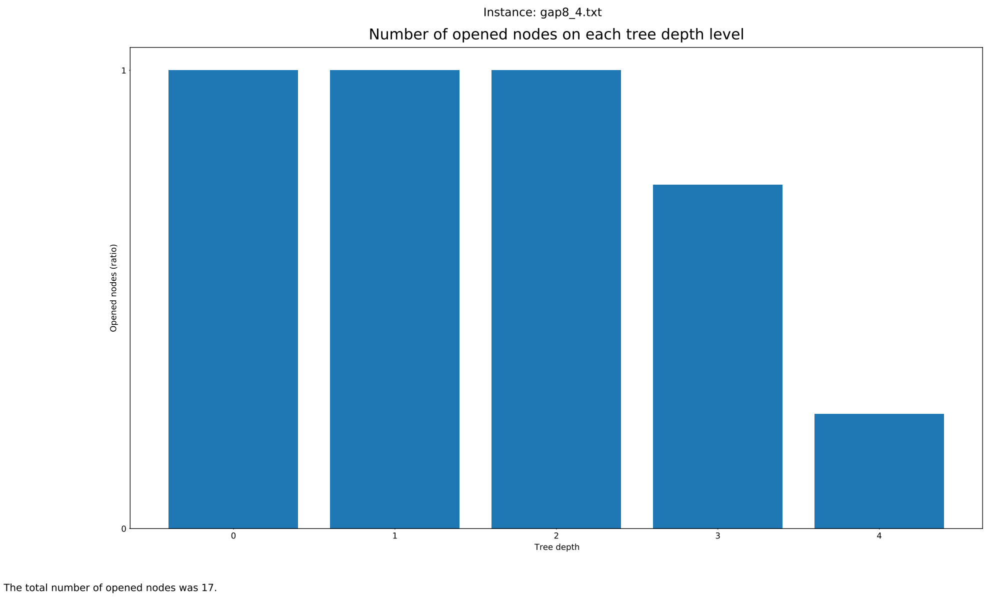
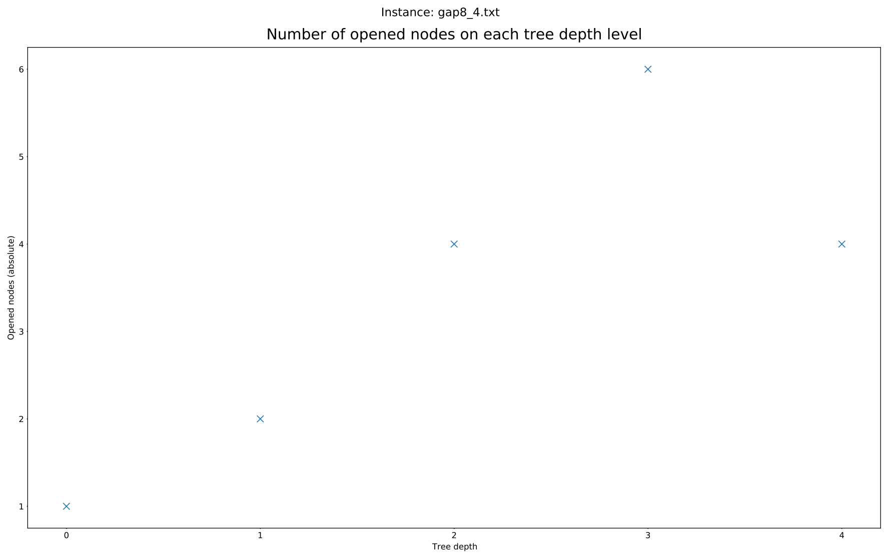


Figure 109: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/gap8_4.txt.default.tree.bar.pdf`



The total number of opened nodes was 17.

Figure 110: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/gap8_4.txt.default.tree.plot.pdf`

1.12 Instance: strong_d_45_15_12

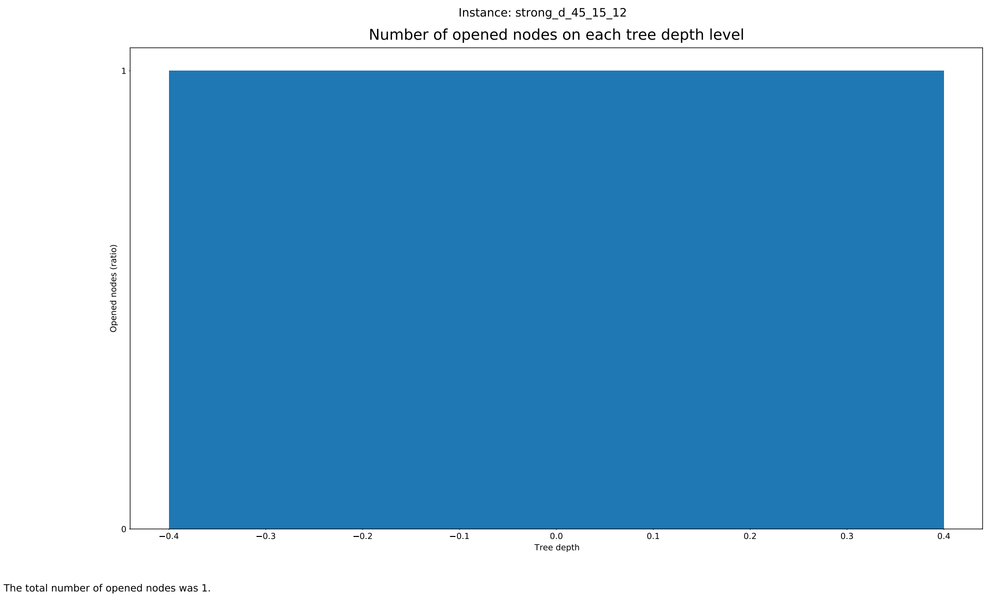


Figure 111: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/strong_d_45_15_12.default.tree.bar.pdf`

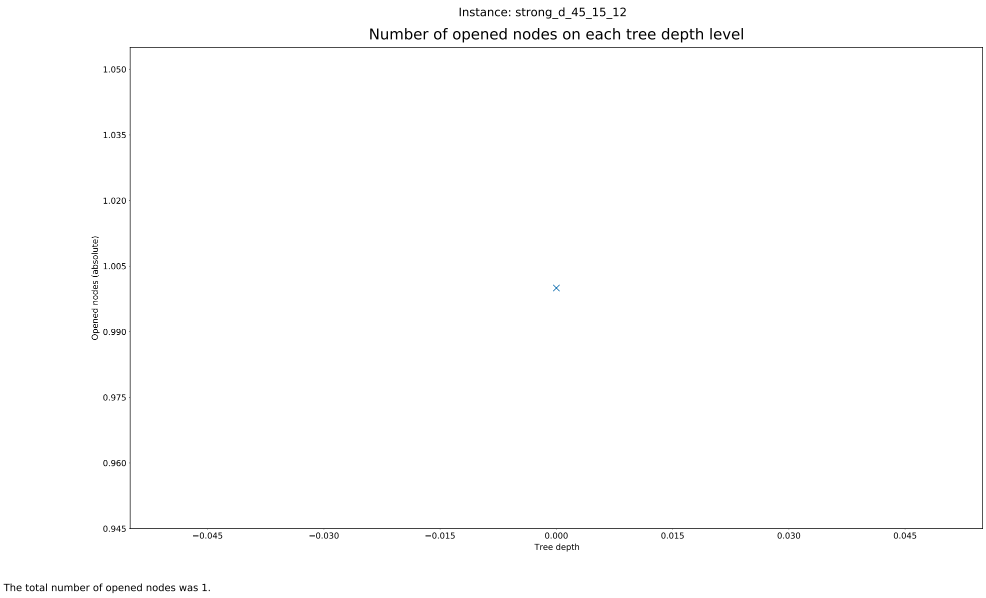


Figure 112: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/strong_d_45_15_12.default.tree.plot.pdf`

1.13 Instance: strong_s_75_15_14

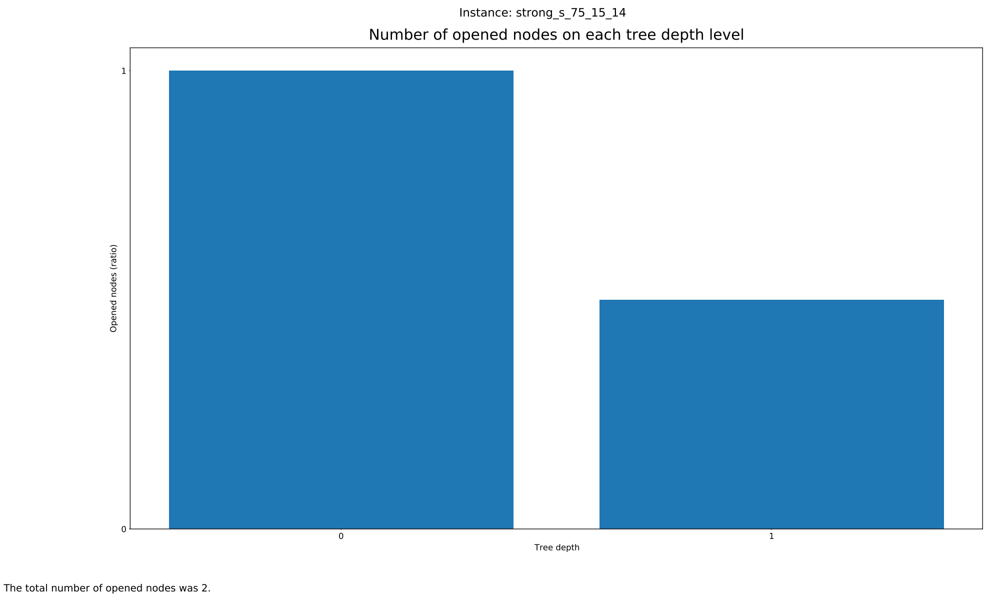


Figure 113: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/strong_s_75_15_14.default.tree.bar.pdf`

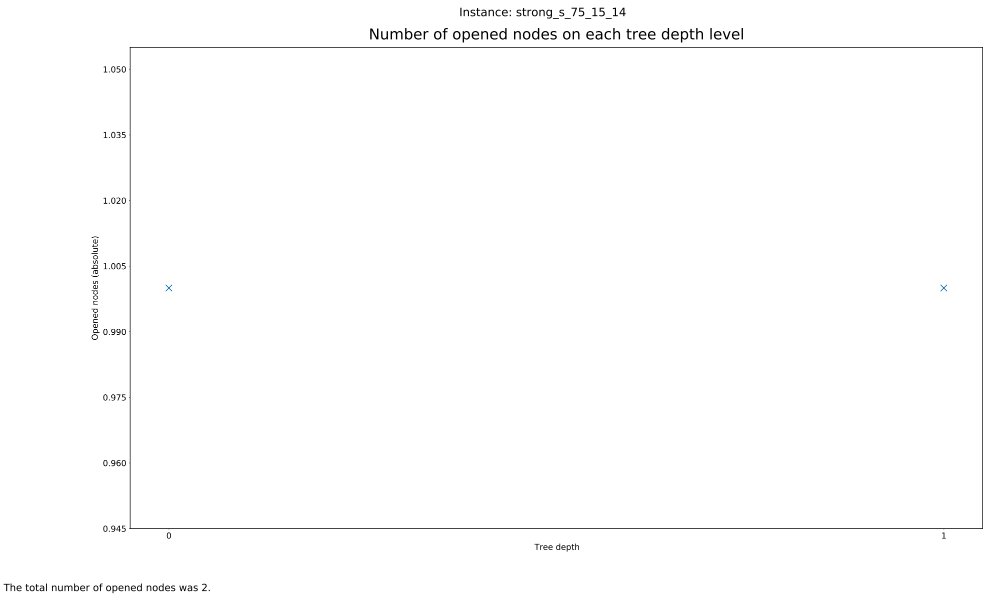


Figure 114: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/strong_s_75_15_14.default.tree.plot.pdf`

1.14 Instance: strong_s_75_15_18

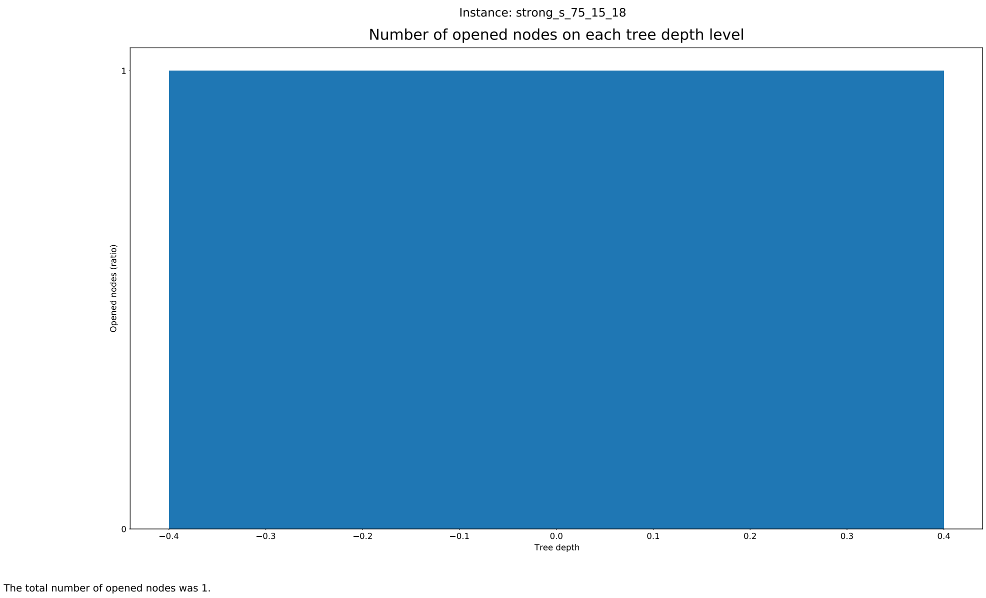


Figure 115: This plot shows the percentage of nodes in the Branch-and-Bound tree opened on each level against how many exist on this level (2^x).
Visualization Path: `plots/tree/strong_s_75_15_18.default.tree.bar.pdf`

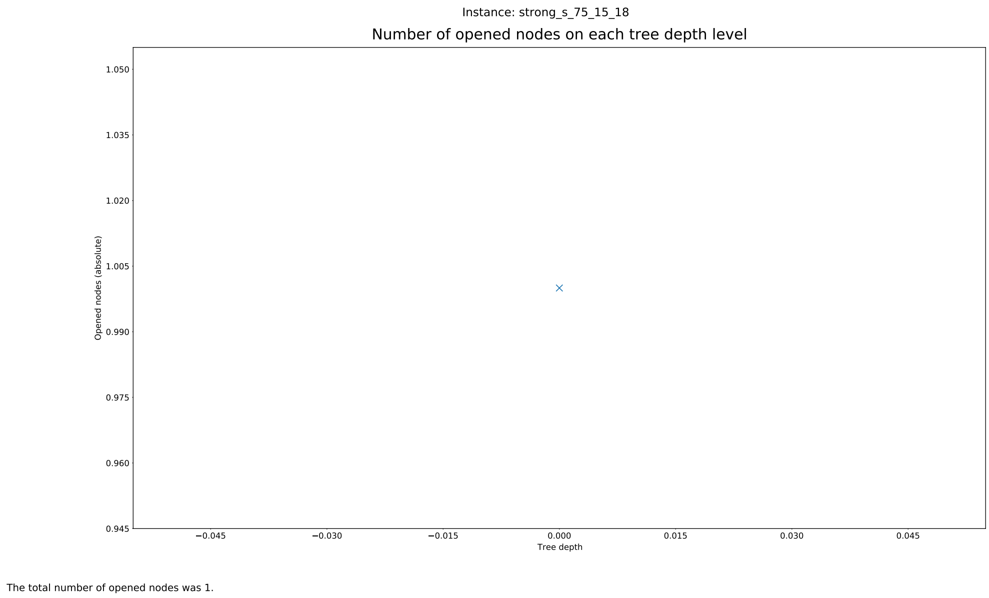


Figure 116: This plot shows the distribution of nodes in the Branch-and-Bound tree opened in absolute terms.
Visualization Path: `plots/tree/strong_s_75_15_18.default.tree.plot.pdf`

2 Aggregated Information

2.1 Time Distribution Plot

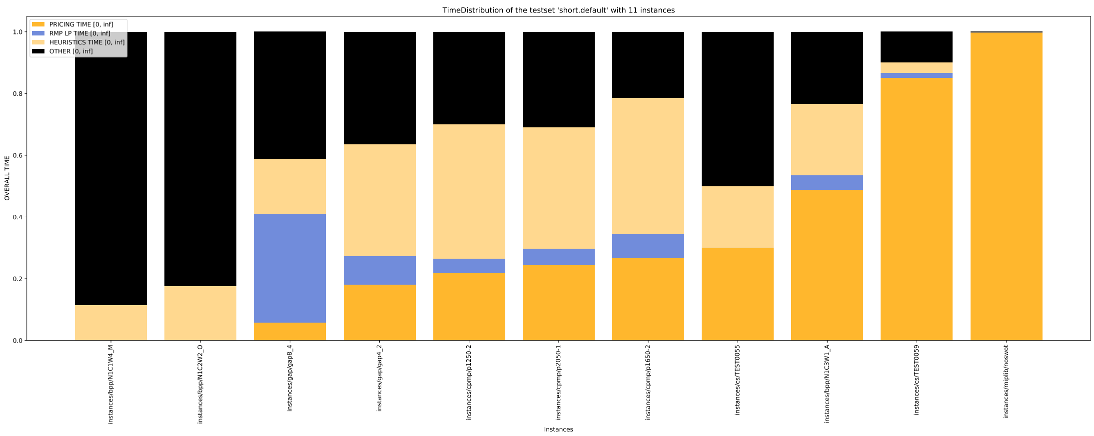


Figure 117: A stacked bar chart of the normalized time distribution of all instances in the test set.
Visualization Path: `plots/timedist/short.default.timedist.bar.pdf`

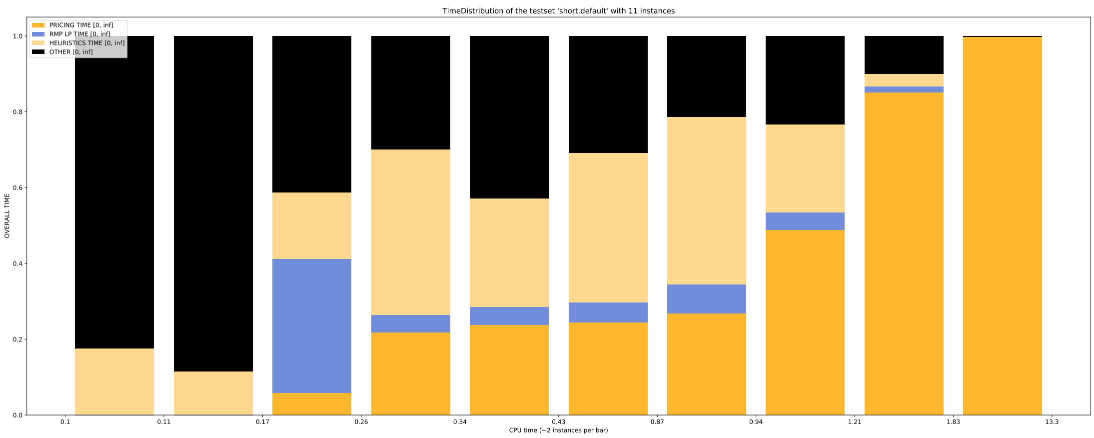


Figure 118: A normalized grouped bar chart of the testset, showing the distribution of the distribution of times.
Visualization Path: `plots/timedist/short.default.timedist.grouped_bar.pdf`

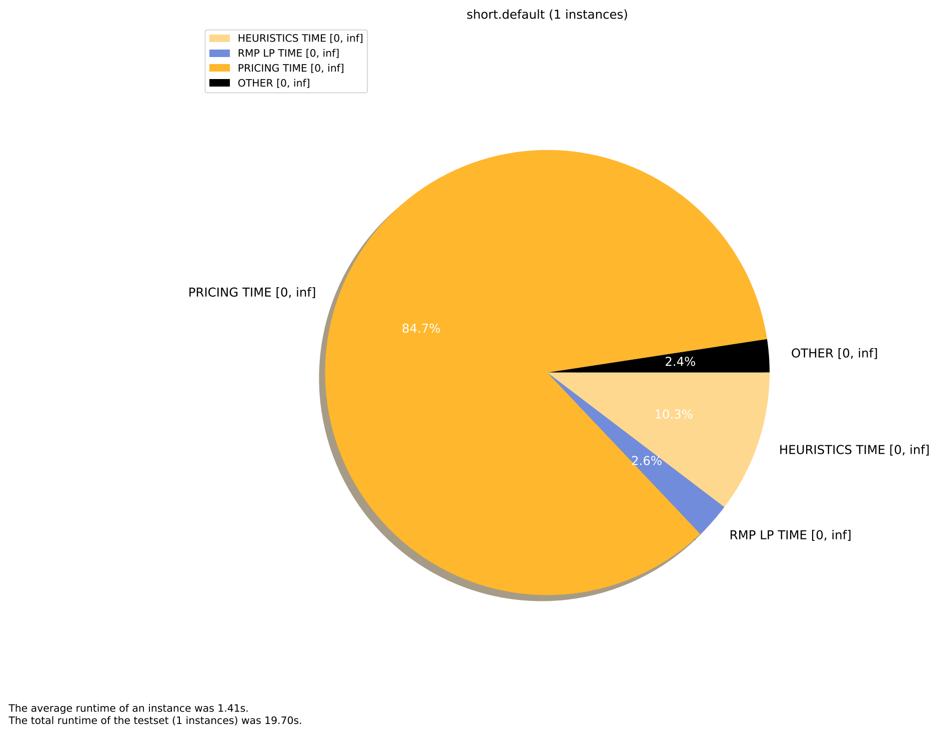


Figure 119: An averaged pie chart for the whole testset.
Visualization Path: `plots/timedist/short.default.timedist.pie.pdf`

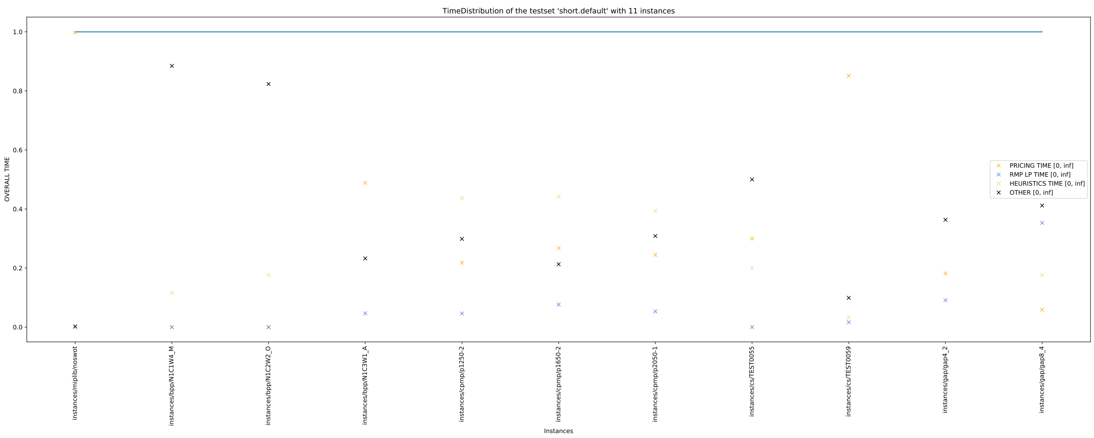


Figure 120: A simple plot of normalized times used for each instance of the testset.
Visualization Path: `plots/timedist/short.default.timedist.plot.pdf`

2.2 Detection Visualizations

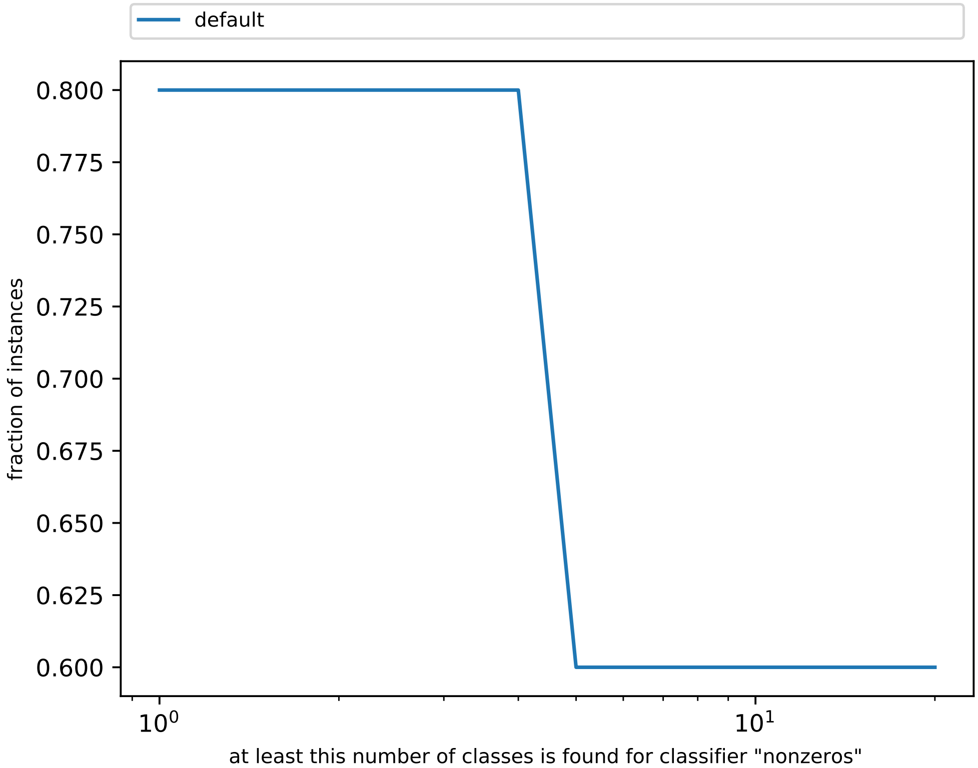


Figure 121: With this visualization, one can see how many classes a classifier determined for the variables. It can be chosen for which classifier to plot this, here ” was chosen.
Visualization Path: `plots/detection/short.detection.classification_classes_nonzeros.pdf`

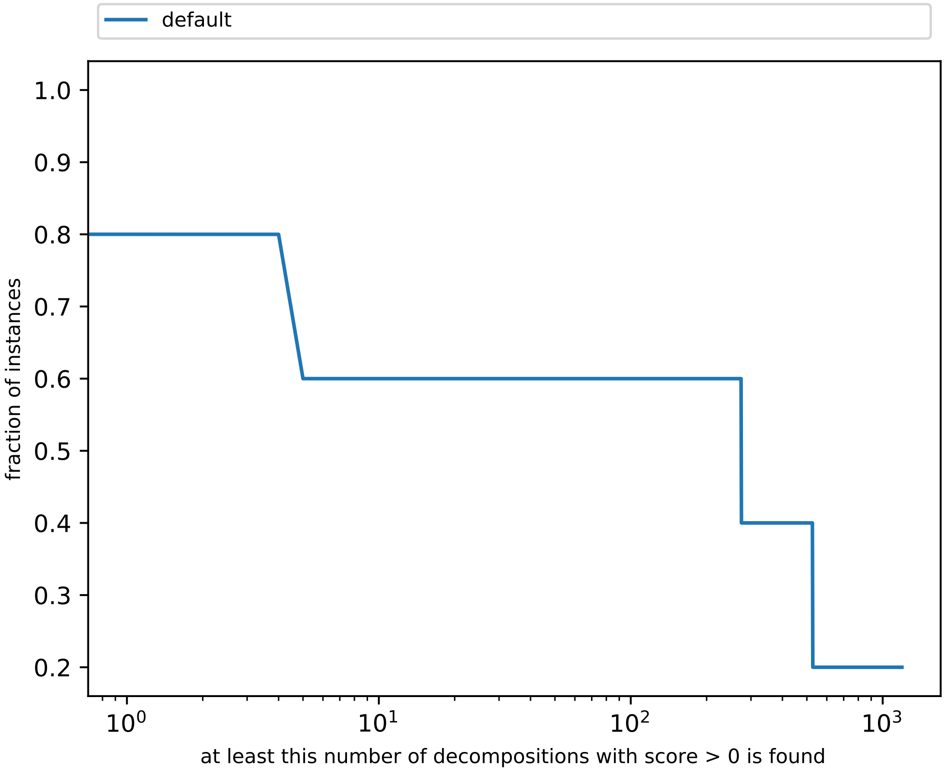


Figure 122: The above plot shows the number of decompositions that were found for what fraction of instances in the given test set on a logarithmic scale. Only those decompositions are shown that have a score that is strictly greater than 0.
Visualization Path: `plots/detection/short.detection.decomps.pdf`

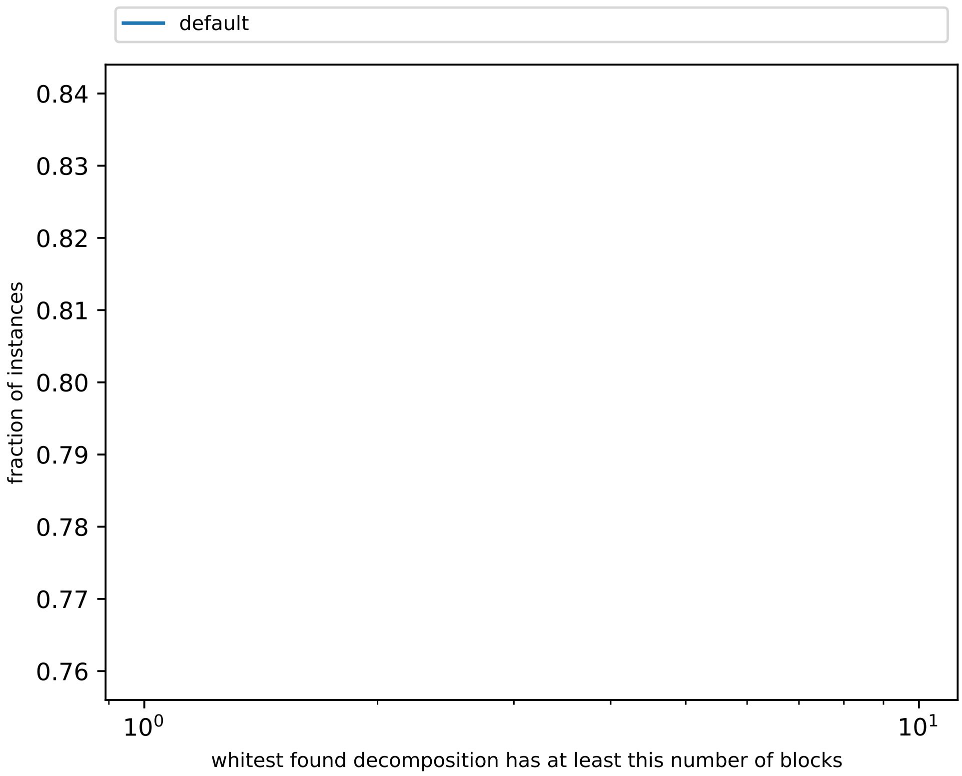


Figure 123: This plot shows how many blocks are used in the (according to the max white score) best decomposition, allowing to make statements about the sensibility of different numbers of blocks.
Visualization Path: `plots/detection/short.detection.nBlocksOfBest.pdf`

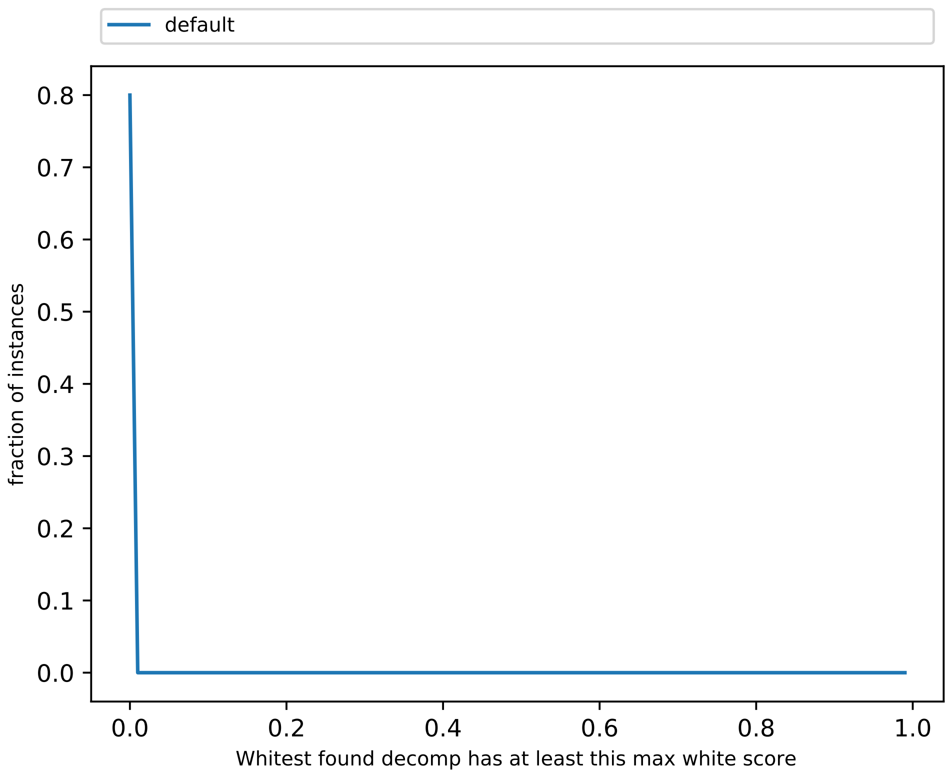


Figure 124: This plot illustrates the detection goodness of the test set (according to the max white score"). If for all instances in the test set, the best (whitest") decomposition was completely white, this plot would show a line with $y = 1$ for all x.
Visualization Path: `plots/detection/short.detection.quality.pdf`

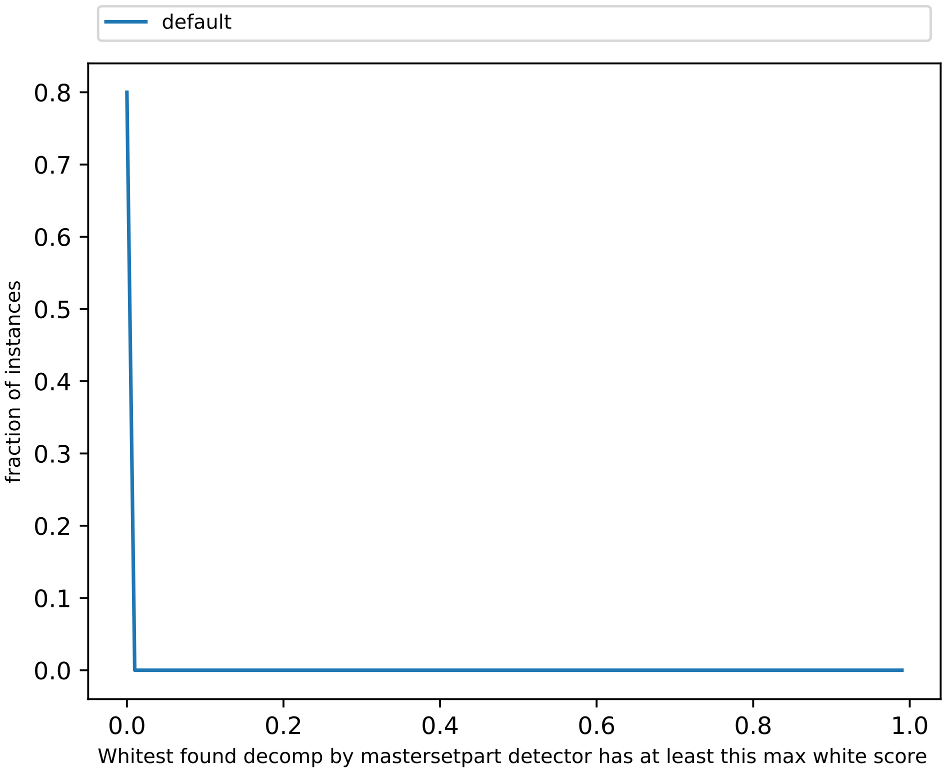


Figure 125: Similarly to the previous plot, this one shows the detection goodness of the whole test set, but for a specific detector, the " detector. For this specific detector, GCG outputs the scores separately in the `detectionstatistics` test mode. With this plot, together with the previous one, one can compare the performance of the Set Partitioning Master detector with the overall performance.
Visualization Path: `plots/detection/short.detection.quality_SetPartMaster.pdf`

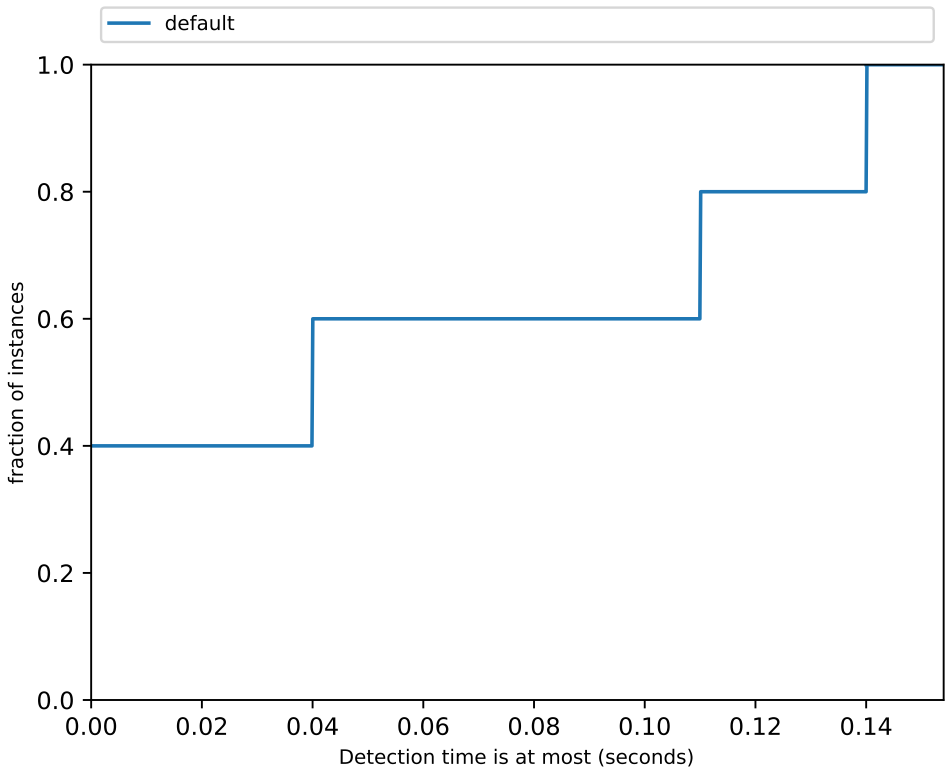


Figure 126: This plot visualizes what fraction of instances against the time used for the whole detection process, including classification and score computation, on a logarithmic scale.
Visualization Path: `plots/detection/short.detection.times.pdf`