Measuring progress in branch-and-bound MILP algorithms

Brady Hunsaker   Osman Özaltın

University of Pittsburgh

2006 MIP Workshop

Overview

- Reasons to measure progress of branch-and-bound
- Current measures
- Some graphical representations
- A weighted sum measure of progress
- Conclusions and future work

Reasons to measure the progress of branch-and-bound

- How good is the best solution so far?

Reasons to measure the progress of branch-and-bound

- How good is the best solution so far?
- How much longer until we have a proven optimal solution?

Reasons to measure the progress of branch-and-bound

- How good is the best solution so far?
- How much longer until we have a proven optimal solution?
- How likely is it that a better solution will be found, and how much better will it be?

Reasons to measure the progress of branch-and-bound

- How good is the best solution so far?
- How much longer until we have a proven optimal solution?
- How likely is it that a better solution will be found, and how much better will it be?
- Should we change any algorithm strategies? (branching, node selection, cuts, ...)

Current measures

- Optimality gap

Current measures

- Optimality gap
- Number of active nodes

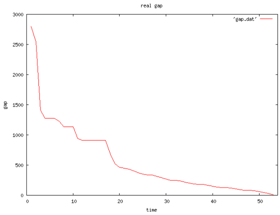
Current measures

- Optimality gap
- Number of active nodes
- Predicted tree size

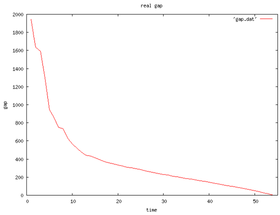
Current measures

- Optimality gap
- Number of active nodes
- Predicted tree size
- Some internal measures used for guiding the algorithm

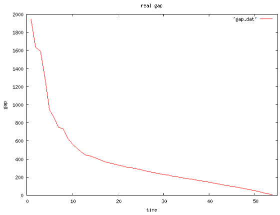
Optimality gap: strengths and weaknesses

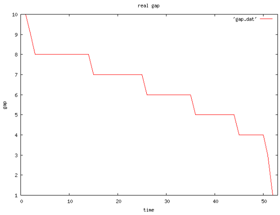


Optimality gap: strengths and weaknesses

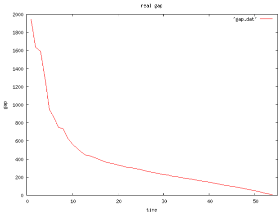


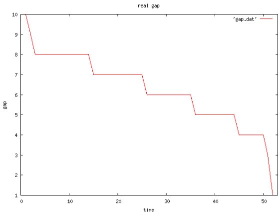
Optimality gap: strengths and weaknesses

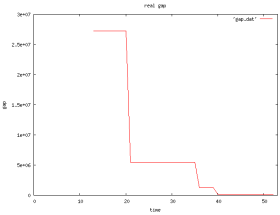




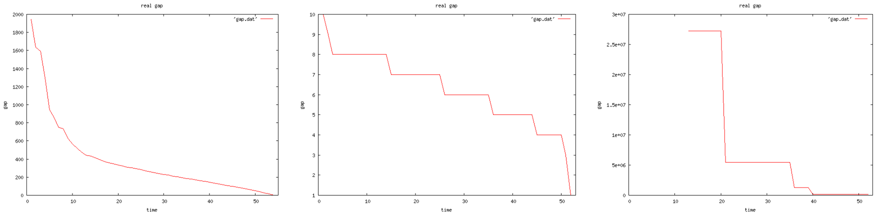
Optimality gap: strengths and weaknesses





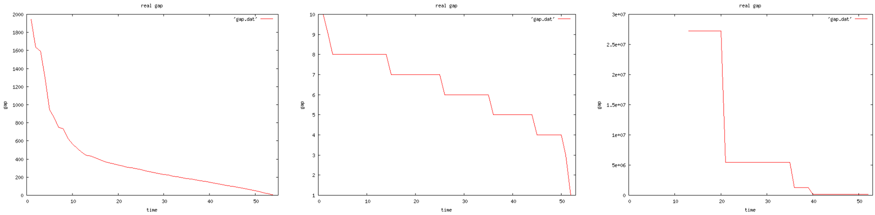


Optimality gap: strengths and weaknesses



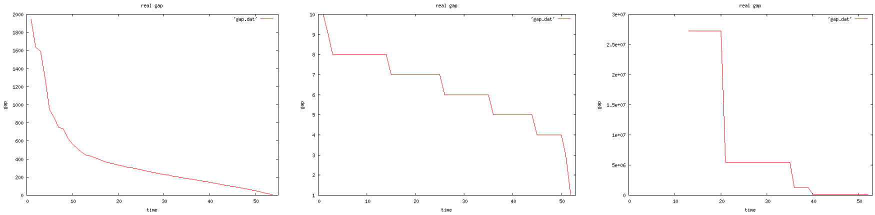
- Strength: guarantee on quality of solution

Optimality gap: strengths and weaknesses



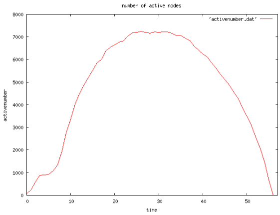
- Strength: guarantee on quality of solution
- Strength: nonincreasing

Optimality gap: strengths and weaknesses

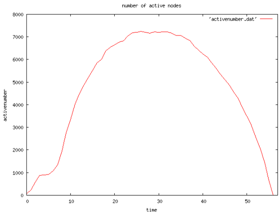


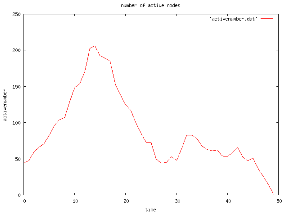
- Strength: guarantee on quality of solution
- Strength: nonincreasing
- Weakness: may remain constant for long periods, then drop suddenly

Number of active nodes: strengths and weaknesses

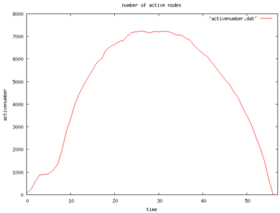


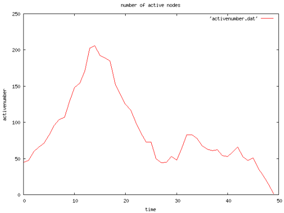
Number of active nodes: strengths and weaknesses

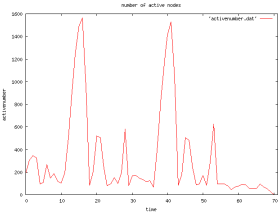




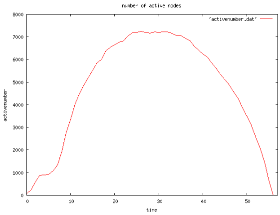
Number of active nodes: strengths and weaknesses

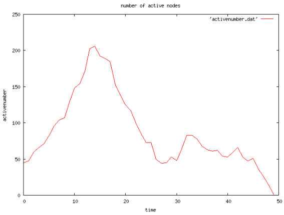


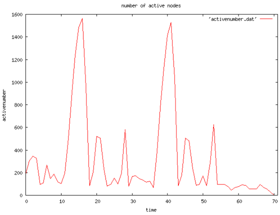




Number of active nodes: strengths and weaknesses

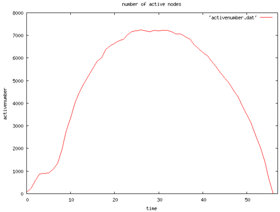


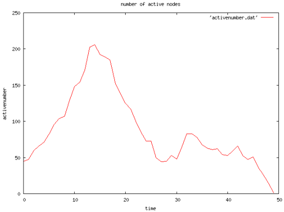


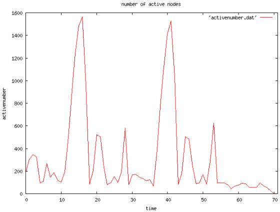


- Strength: some sense of "work remaining"

Number of active nodes: strengths and weaknesses

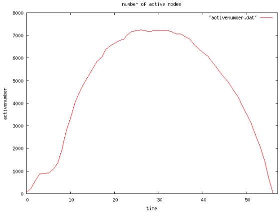


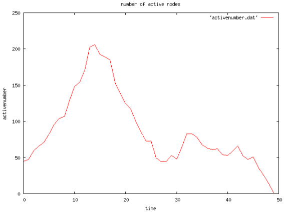


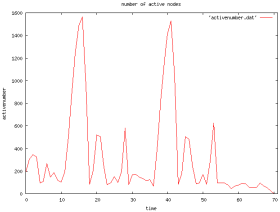


- Strength: some sense of "work remaining"
- Weakness: may go up and down

Number of active nodes: strengths and weaknesses







- Strength: some sense of “work remaining”
- Weakness: may go up and down
- Weakness: not all active nodes are equal

Predicting the eventual tree size

- Approach adopted by Cornuéjols, Karamanov, and Li (2006)

Predicting the eventual tree size

- Approach adopted by Cornuéjols, Karamanov, and Li (2006)
- Goal: make an early prediction of the solution time within an order of magnitude

Predicting the eventual tree size

- Approach adopted by Cornuéjols, Karamanov, and Li (2006)
- Goal: make an early prediction of the solution time within an order of magnitude
- The total number of nodes that will be explored is estimated early in the process

Predicting the eventual tree size

- Approach adopted by Cornuéjols, Karamanov, and Li (2006)
- Goal: make an early prediction of the solution time within an order of magnitude
- The total number of nodes that will be explored is estimated early in the process
- Strength: Addresses a key question

Predicting the eventual tree size

- Approach adopted by Cornuéjols, Karamanov, and Li (2006)
- Goal: make an early prediction of the solution time within an order of magnitude
- The total number of nodes that will be explored is estimated early in the process
- Strength: Addresses a key question
- Weakness: Estimate is based on a common tree shape, but this tree shape depends on specific algorithm implementation and parameters

Possible goals in analyzing b&b progress

- Predict the time to completion

Possible goals in analyzing b&b progress

- Predict the time to completion
- Predict the optimal objective value

Possible goals in analyzing b&b progress

- Predict the time to completion
- Predict the optimal objective value
- Determine when to change algorithm strategies

Possible goals in analyzing b&b progress

- Predict the time to completion
- Predict the optimal objective value
- Determine when to change algorithm strategies
- Find a good measure of progress

Information available during b&b

- Number of active nodes

Information available during b&b

- Number of active nodes
- For each active node:

Information available during b&b

- Number of active nodes
- For each active node:
    - LP bound

Information available during b&b

- Number of active nodes
- For each active node:
  - LP bound
  - integer infeasibility information

Information available during b&b

- Number of active nodes
- For each active node:
  - LP bound
  - integer infeasibility information
  - history/position in tree (such as depth and parent)

Information available during b&b

- Number of active nodes
- For each active node:
  - LP bound
  - integer infeasibility information
  - history/position in tree (such as depth and parent)
- Similar information for each processed node

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit

- CBC: COIN-OR Branch and Cut

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
- CBC: COIN-OR Branch and Cut

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
    - Primary author: Andrew Makhorin, Moscow Aviation Institute
    - Two branch and bound algorithms
- CBC: COIN-OR Branch and Cut

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut
  - Primary author: John Forrest, IBM

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut
  - Primary author: John Forrest, IBM
  - Many sophisticated options

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut
  - Primary author: John Forrest, IBM
  - Many sophisticated options
  - Stronger in general than GLPK

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut
  - Primary author: John Forrest, IBM
  - Many sophisticated options
  - Stronger in general than GLPK
- We modified the source code to collect desired information

Approach: used two open-source codes

- GLPK: GNU Linear Programming Kit
  - Primary author: Andrew Makhorin, Moscow Aviation Institute
  - Two branch and bound algorithms
  - Newer algorithm has option of basic cuts
- CBC: COIN-OR Branch and Cut
  - Primary author: John Forrest, IBM
  - Many sophisticated options
  - Stronger in general than GLPK
- We modified the source code to collect desired information
- So far, we have considered several instances from MIPLIB 3 that take more than 30 seconds but less than an hour

Graphical analysis

- Initial goal: develop visualization tools

Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns

Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot

Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot
- Graphical representations:

Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot
- Graphical representations:
    - Usual: gap, number of active nodes

Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot
- Graphical representations:
  - Usual: gap, number of active nodes
  - Histogram of active node LP bounds

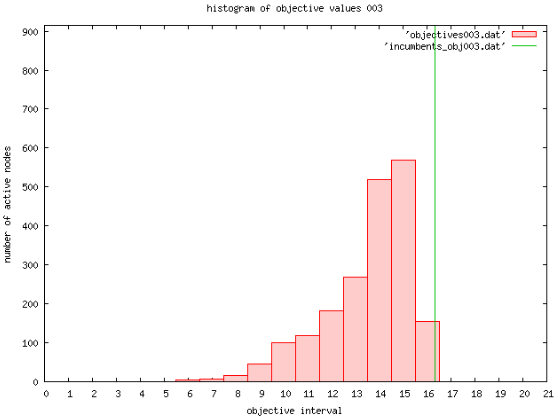
Graphical analysis

- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot
- Graphical representations:
  - Usual: gap, number of active nodes
  - Histogram of active node LP bounds
  - Scatterplot of active node LP bounds and integer infeasibility

Graphical analysis

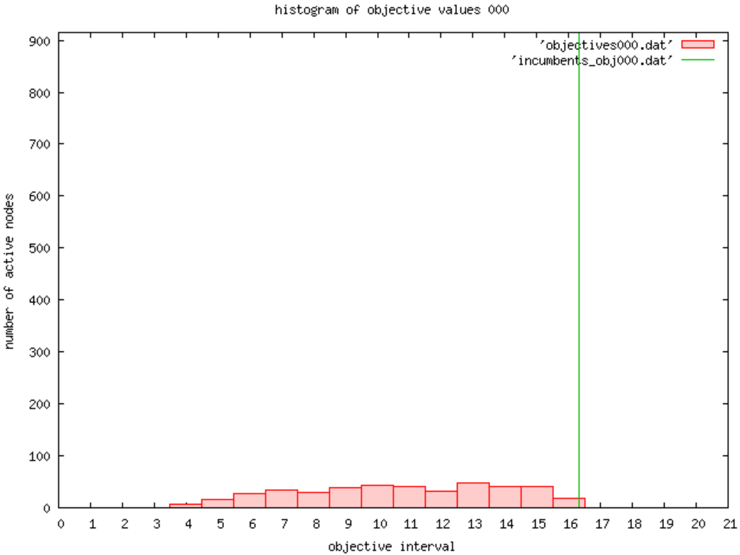
- Initial goal: develop visualization tools
- Let our brains (and yours!) search for patterns
- We made use of Perl, shell scripts, and Gnuplot
- Graphical representations:
  - Usual: gap, number of active nodes
  - Histogram of active node LP bounds
  - Scatterplot of active node LP bounds and integer infeasibility
  - Node history in scatterplot

Histogram of active node LP bounds

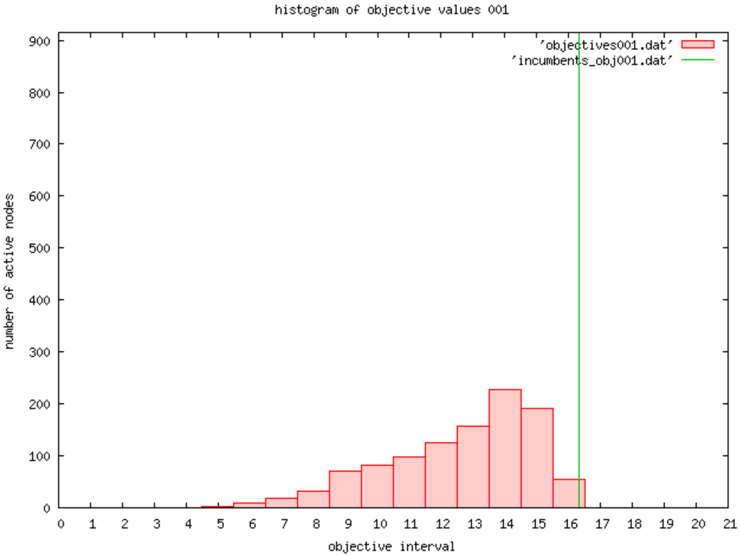


- Horizontal axis is the LP bound bins
- Vertical axis is number of active nodes
- Green vertical line is the current incumbent value

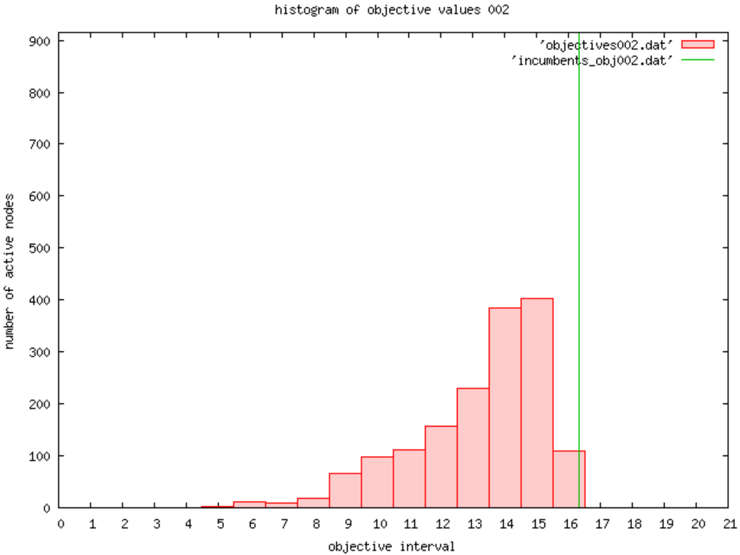
Example histogram series 1



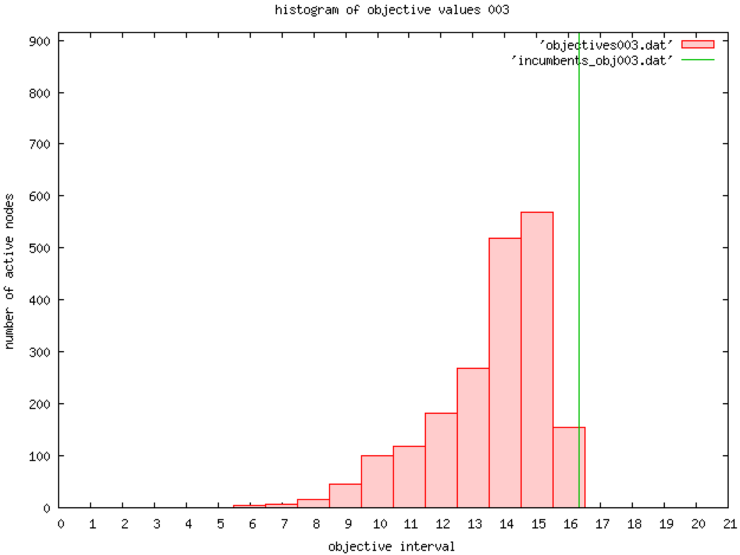
Example histogram series 1



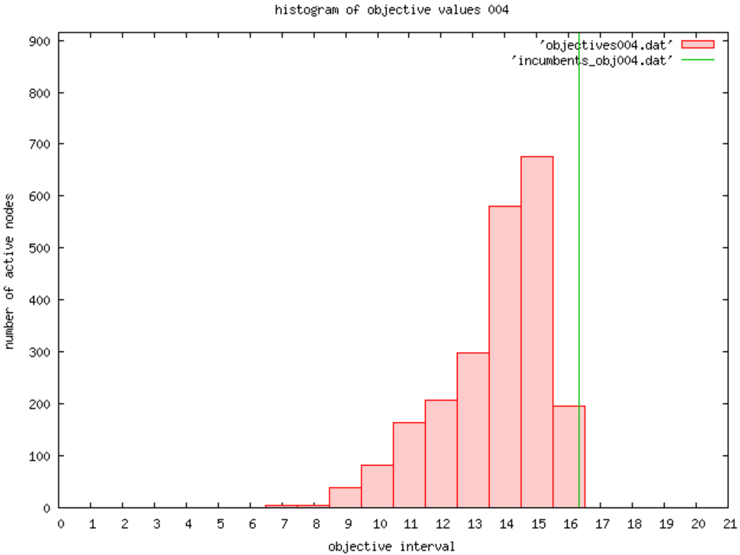
Example histogram series 1



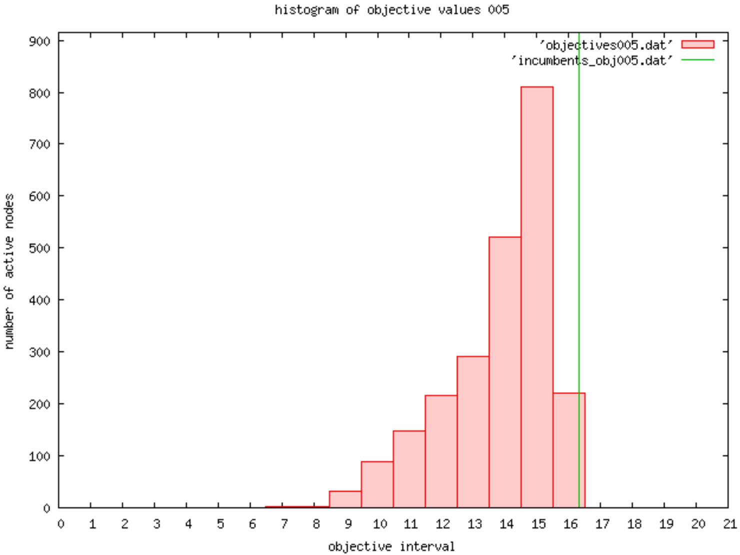
Example histogram series 1



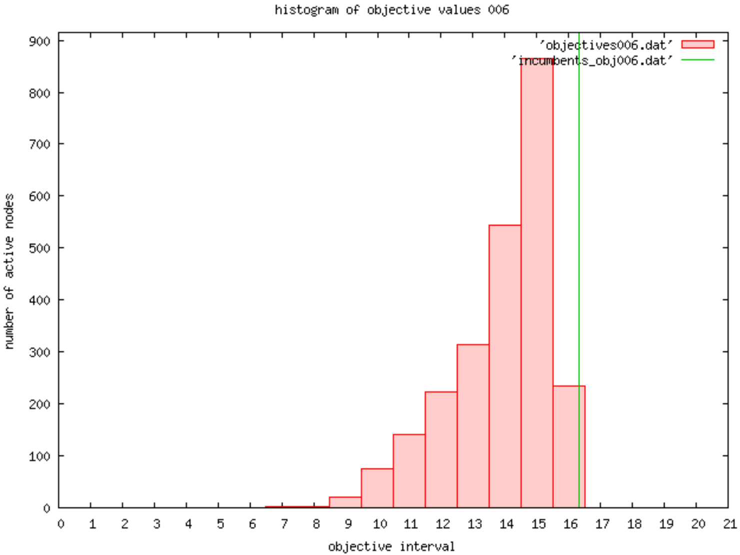
Example histogram series 1



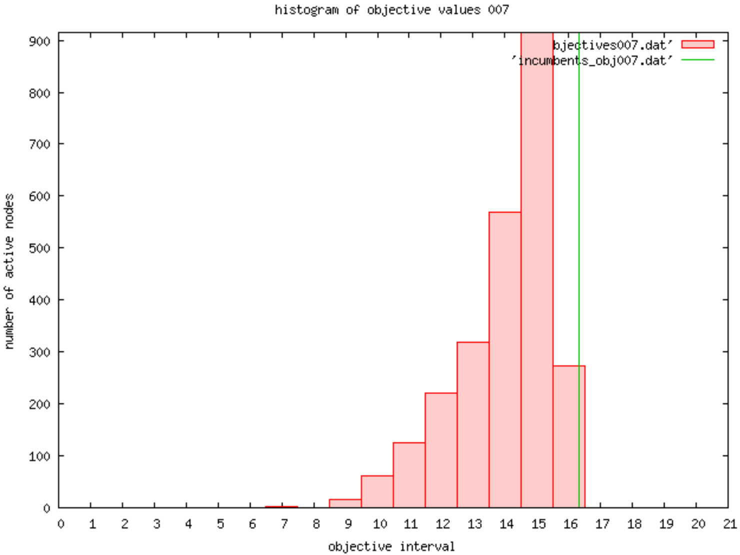
Example histogram series 1



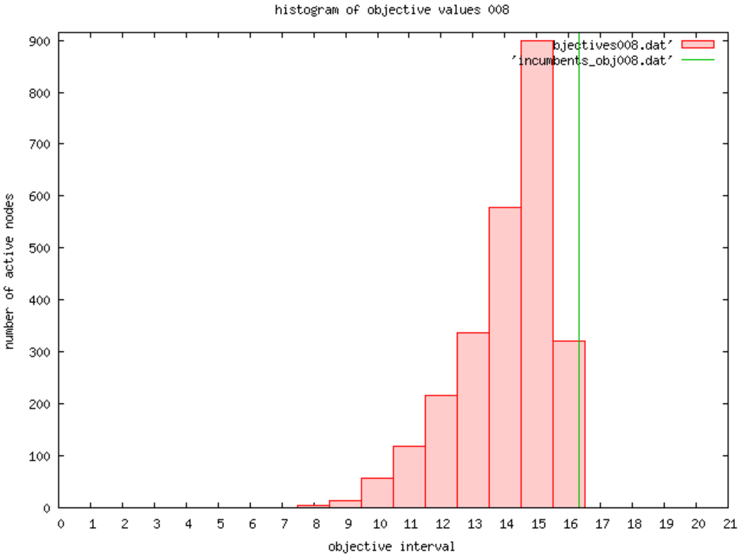
Example histogram series 1



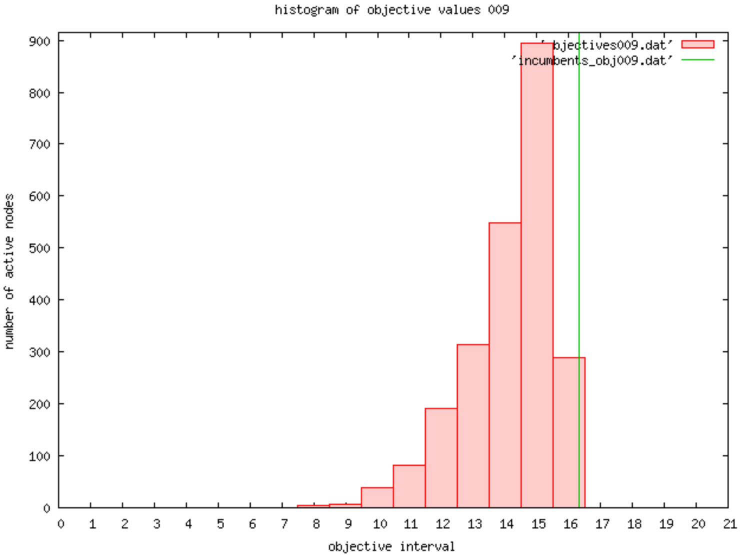
Example histogram series 1



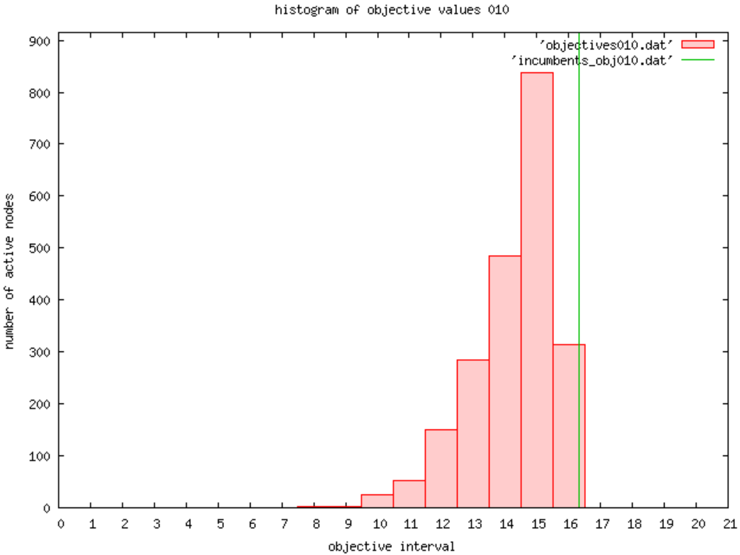
Example histogram series 1



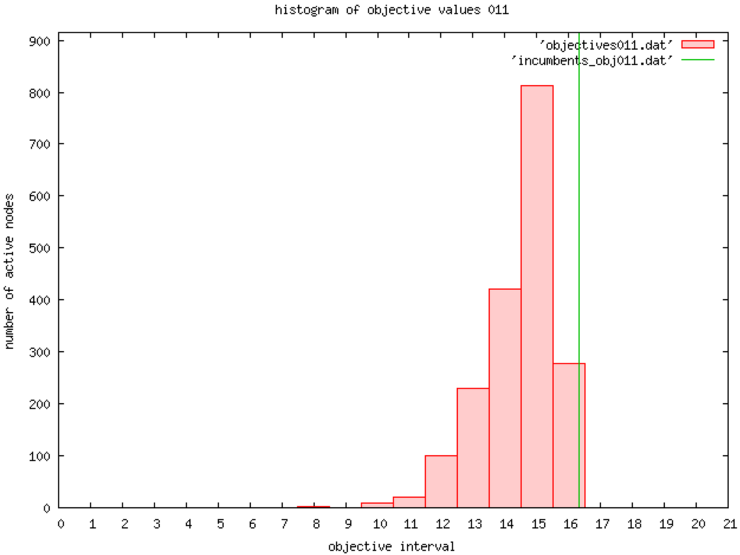
Example histogram series 1



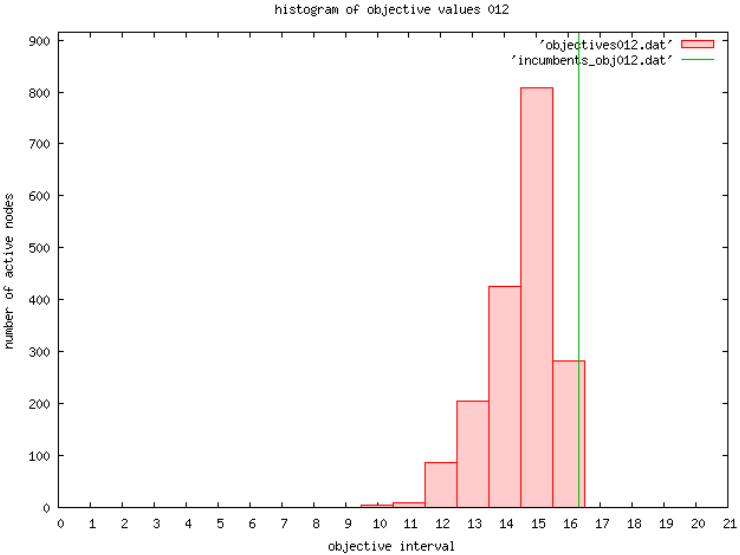
Example histogram series 1



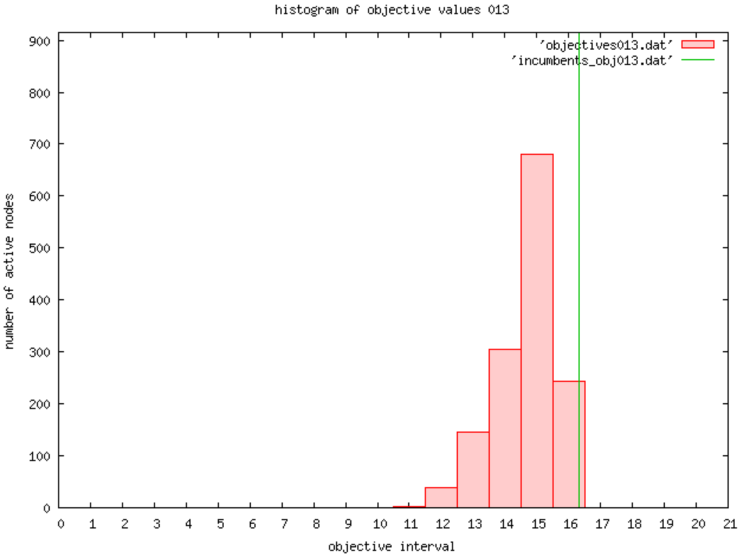
Example histogram series 1



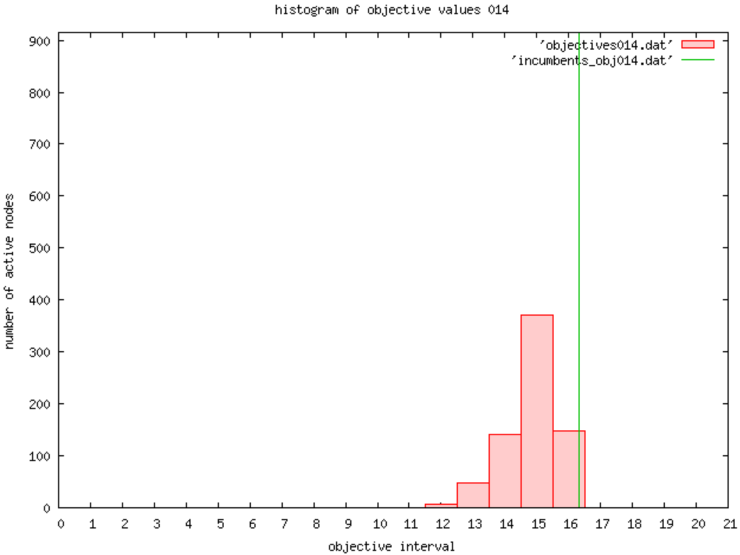
Example histogram series 1



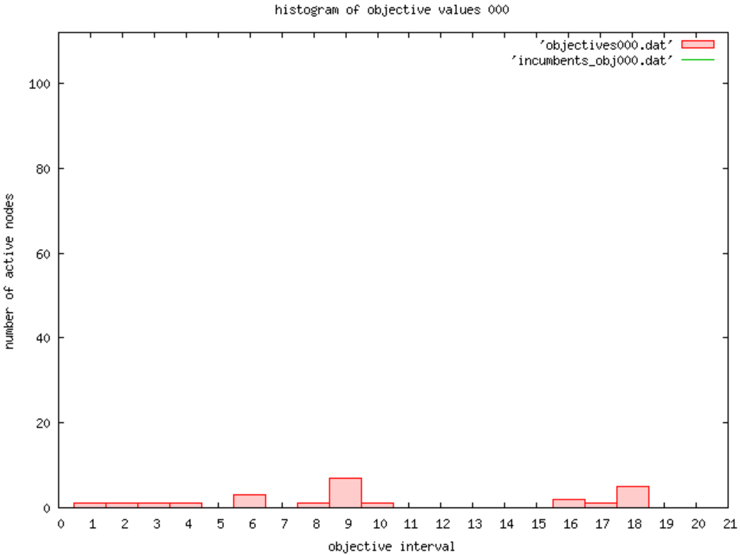
Example histogram series 1



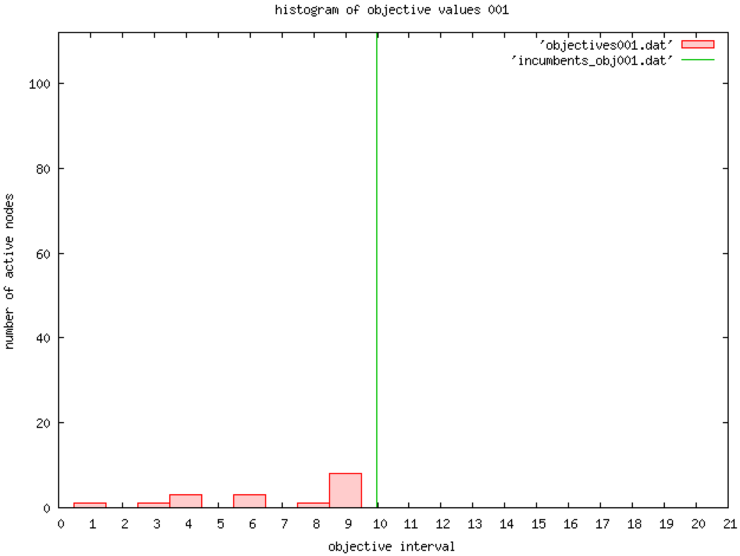
Example histogram series 1



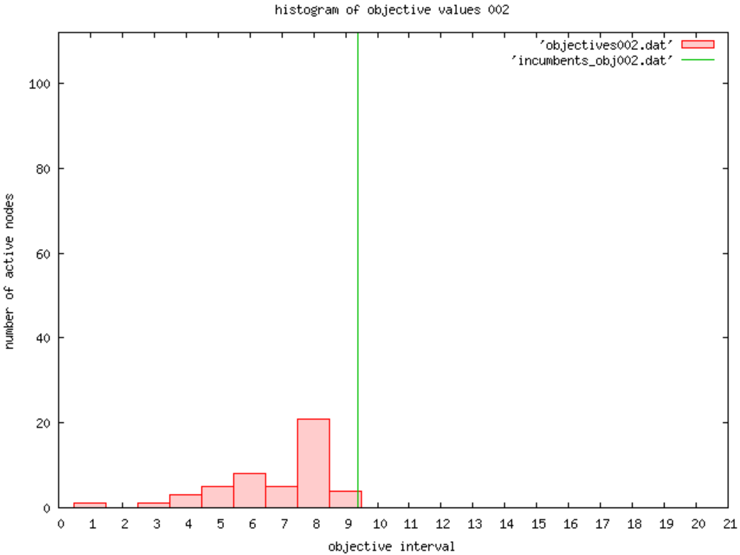
Example histogram series 2



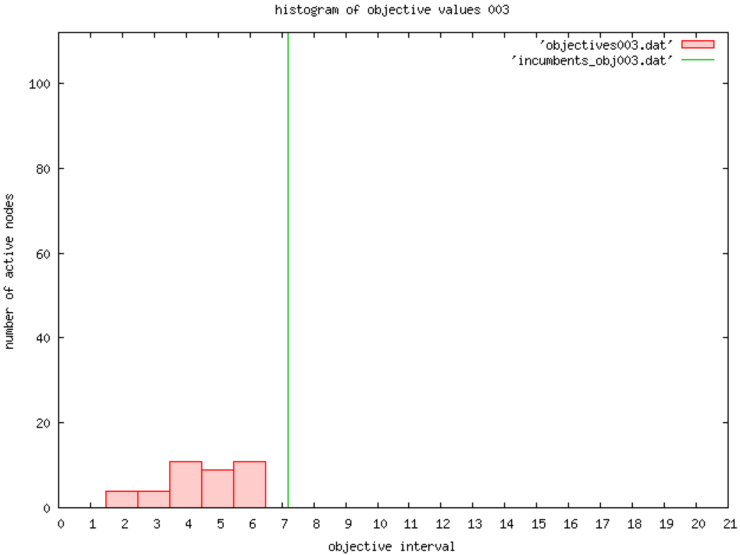
Example histogram series 2



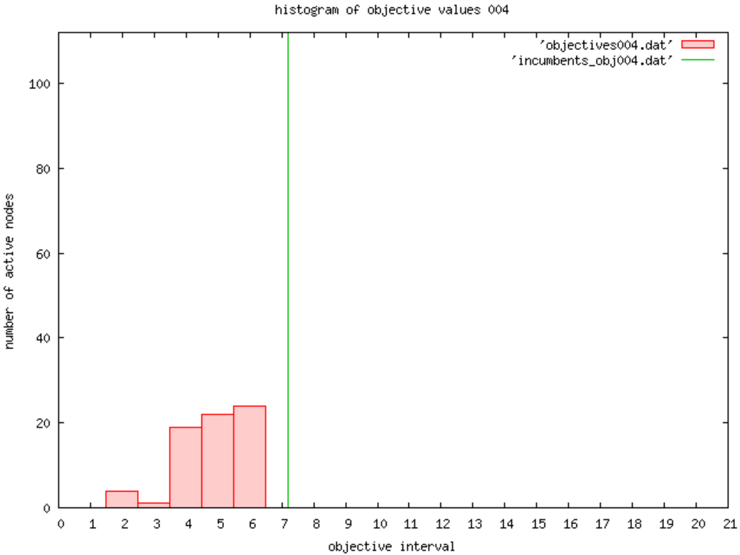
Example histogram series 2



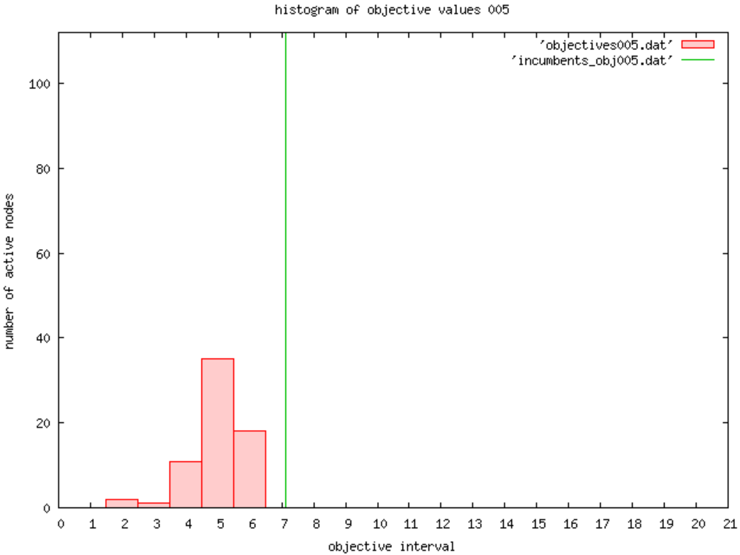
Example histogram series 2



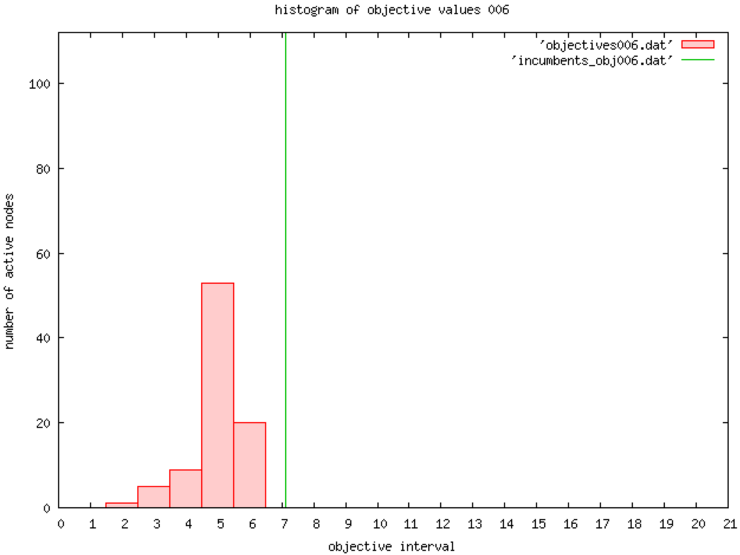
Example histogram series 2



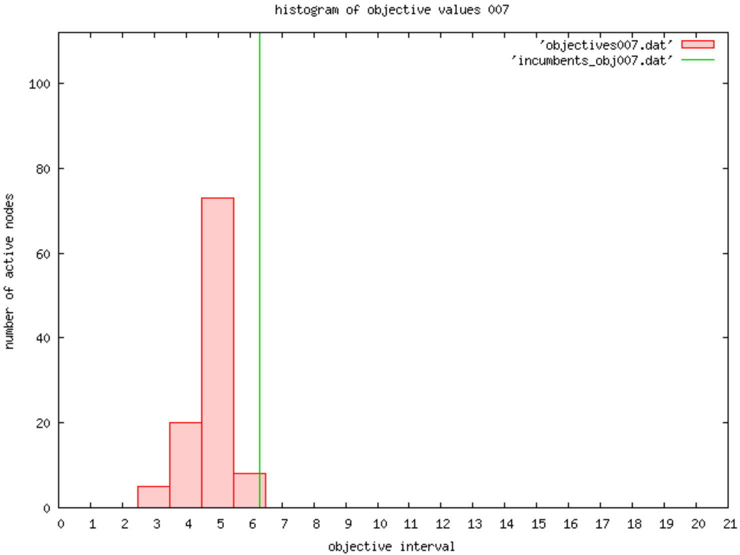
Example histogram series 2



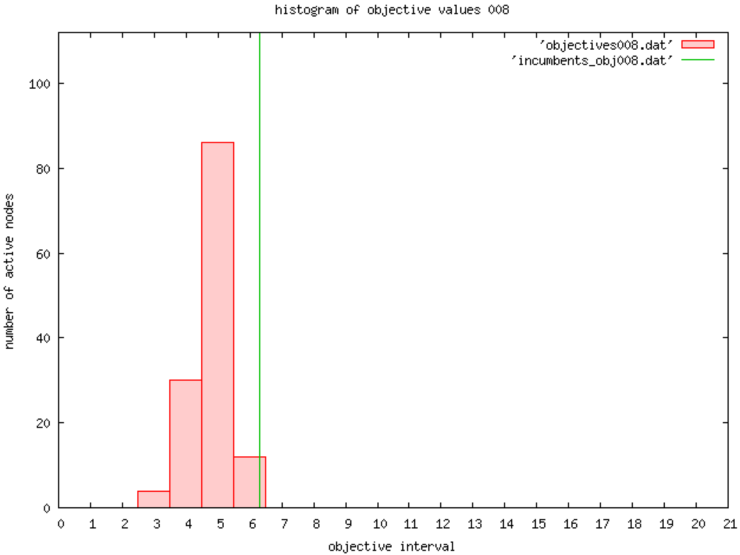
Example histogram series 2



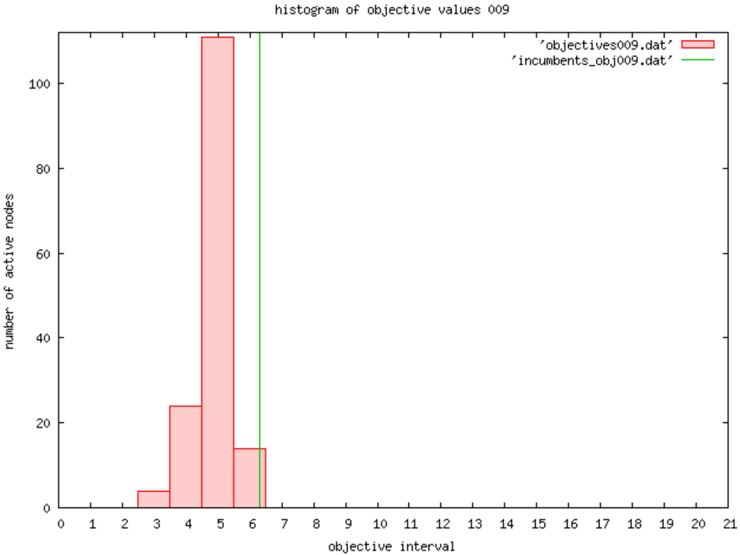
Example histogram series 2



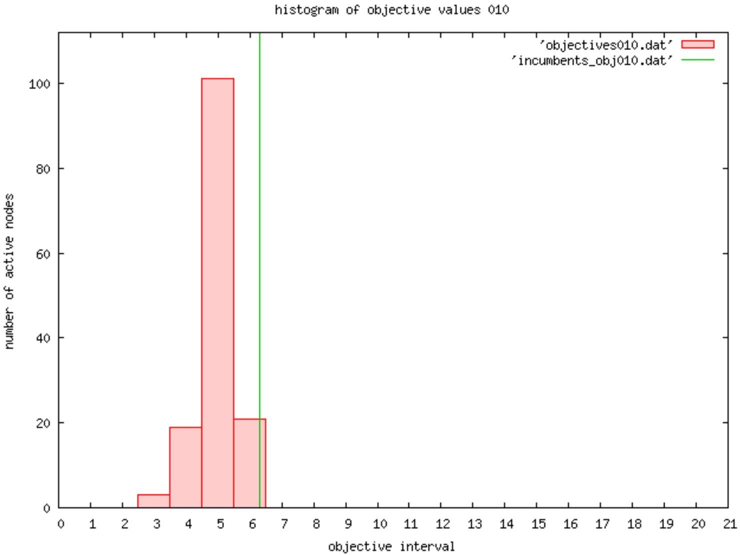
Example histogram series 2



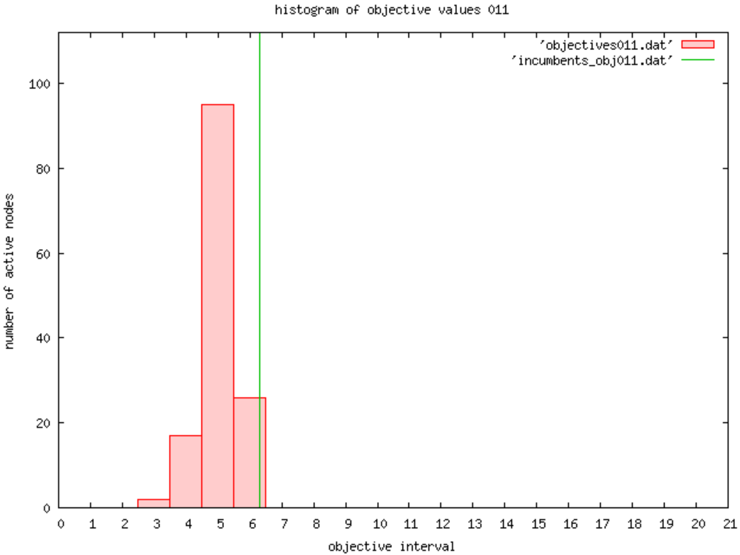
Example histogram series 2



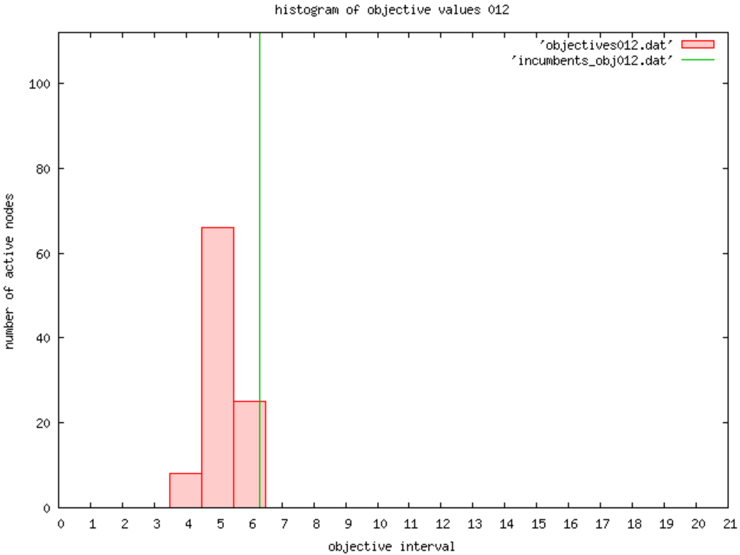
Example histogram series 2



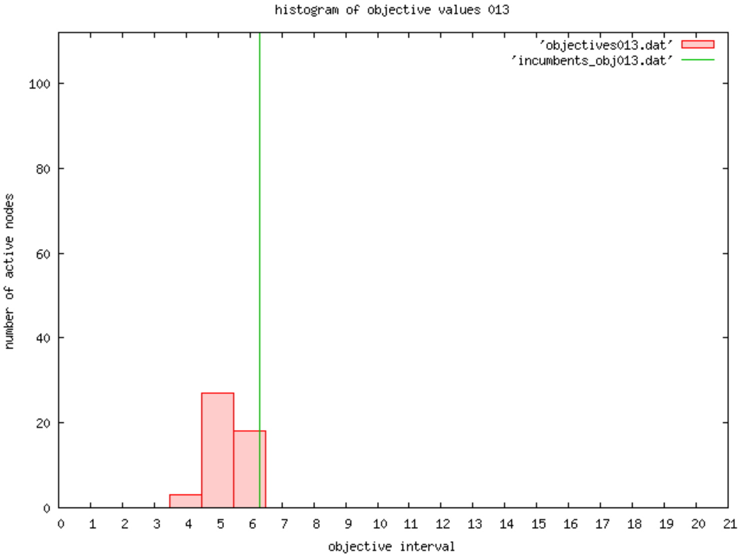
Example histogram series 2



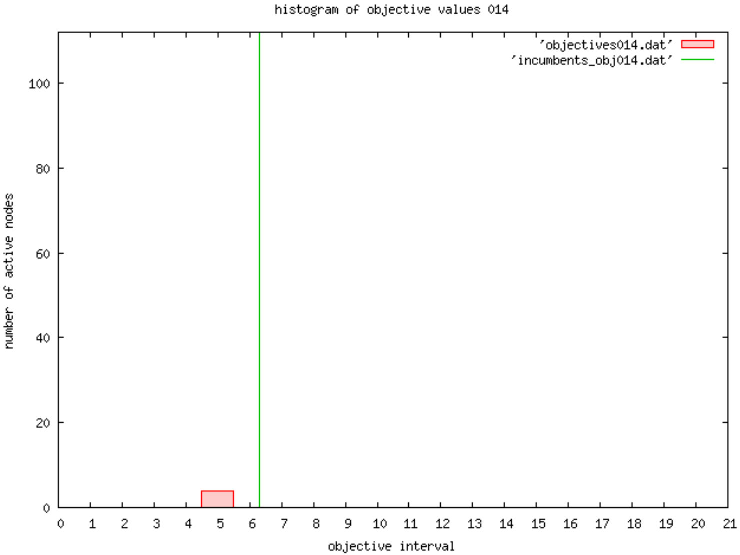
Example histogram series 2



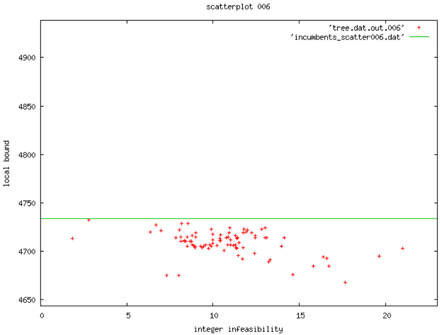
Example histogram series 2



Example histogram series 2

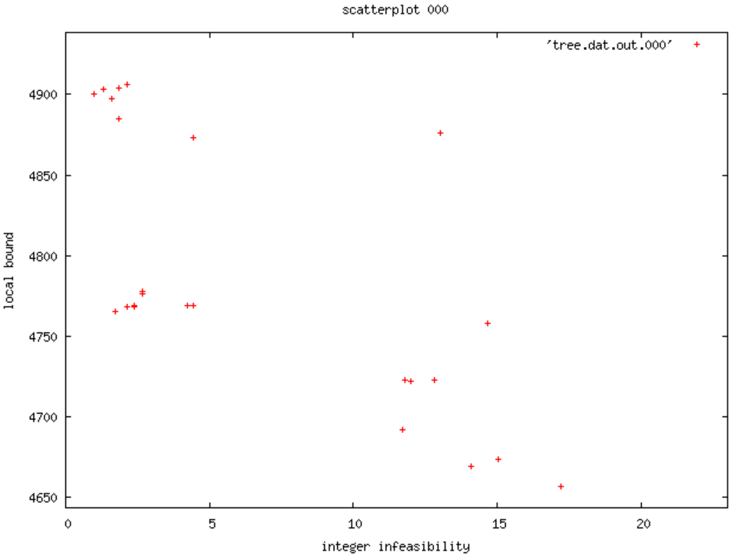


Scatterplot of active node LP bounds and integer infeasibility

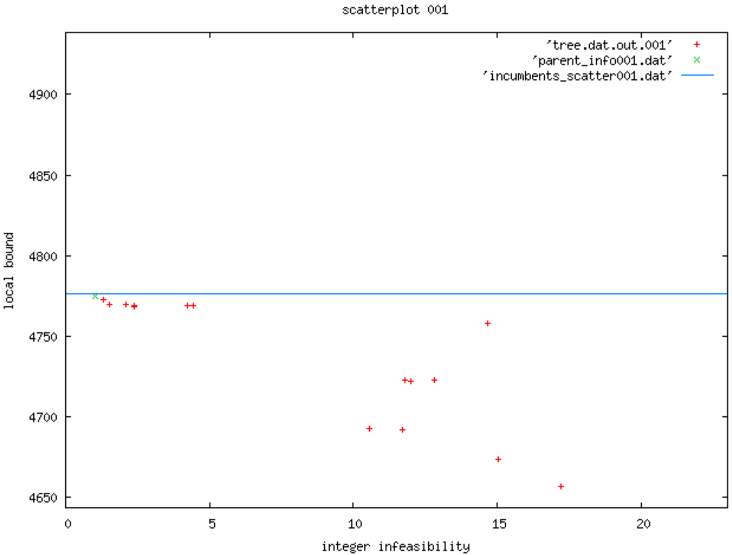


- Points represent active nodes
- Vertical axis is the LP bound
- Horizontal axis is the sum of integer infeasibilities
- Green horizontal line is the current incumbent value

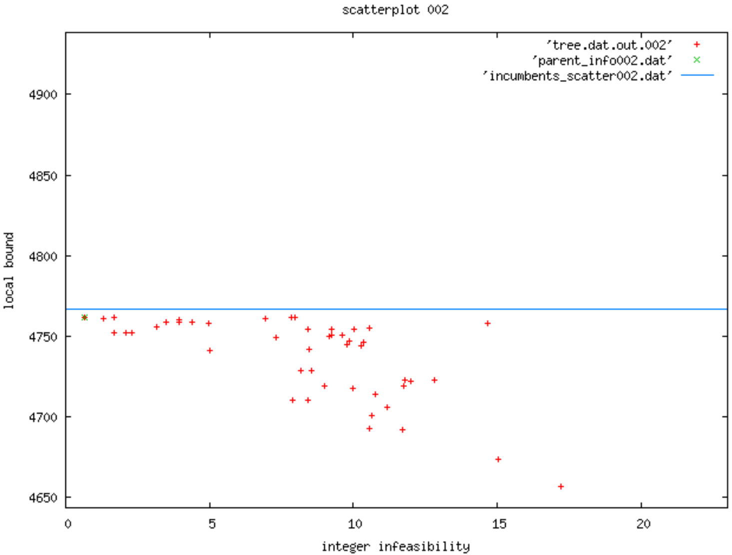
Example scatterplot series



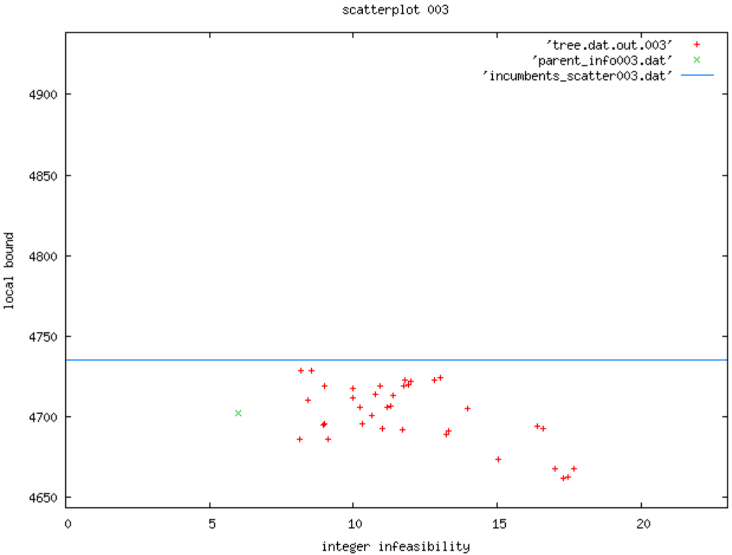
Example scatterplot series



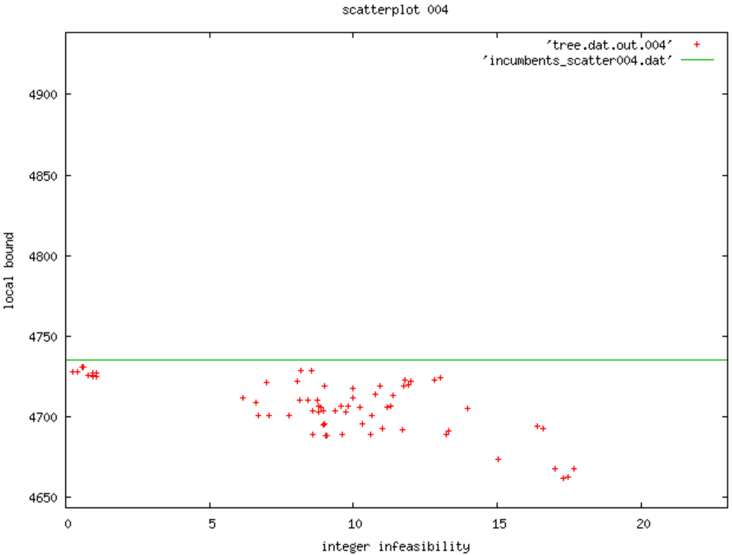
Example scatterplot series



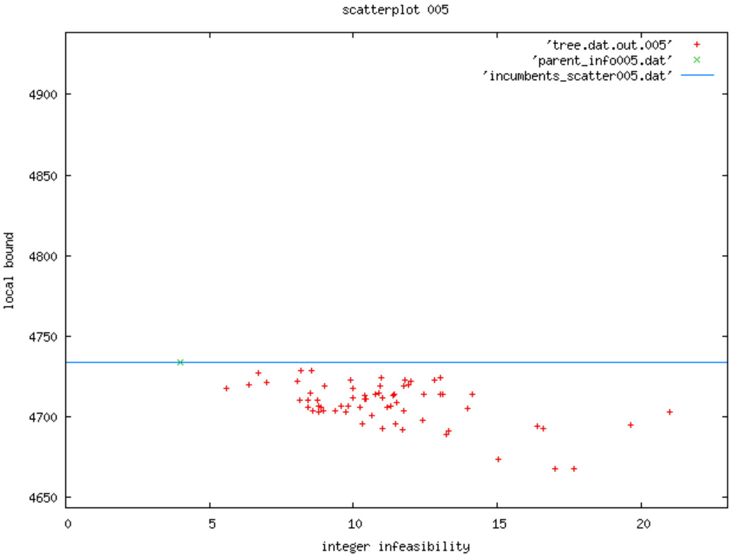
Example scatterplot series



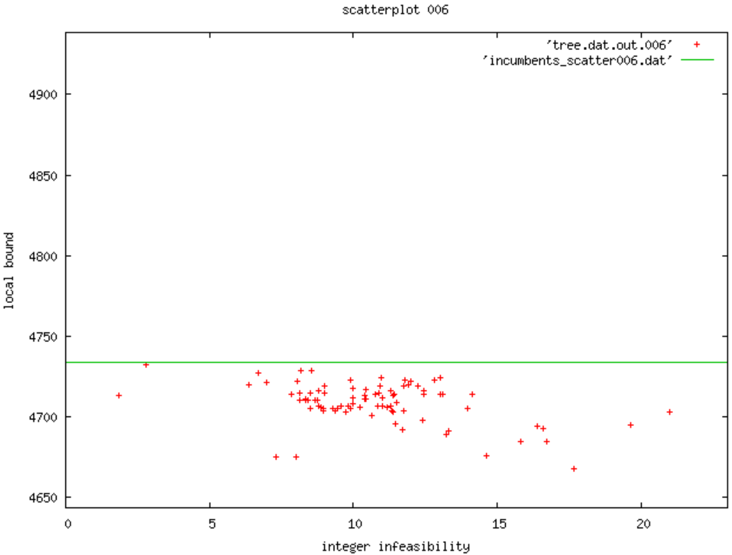
Example scatterplot series



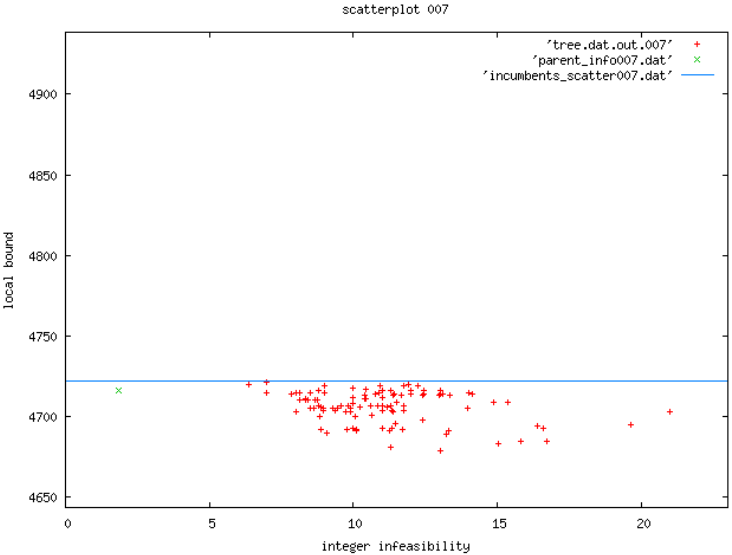
Example scatterplot series



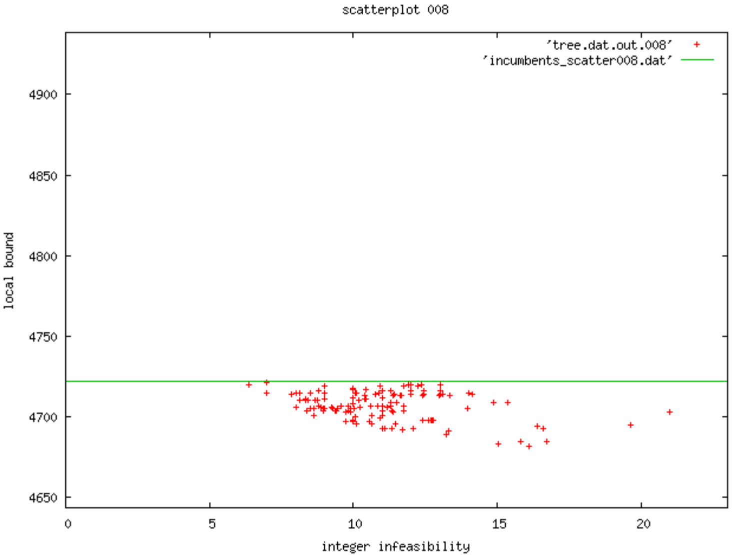
Example scatterplot series



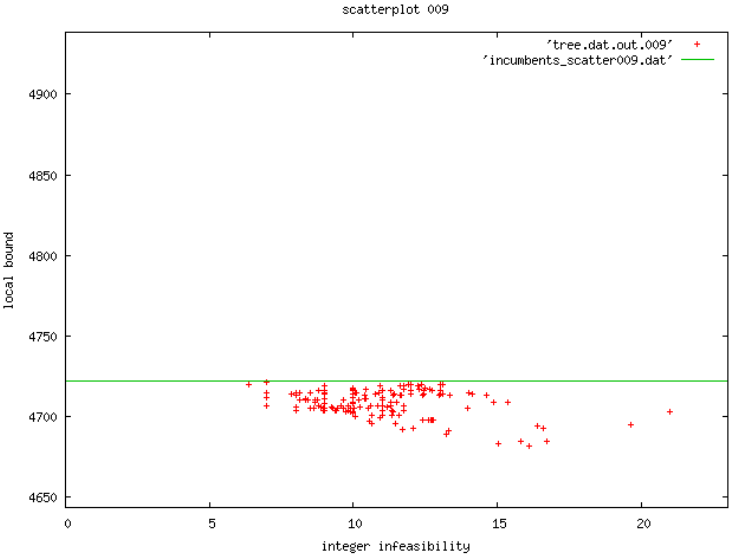
Example scatterplot series



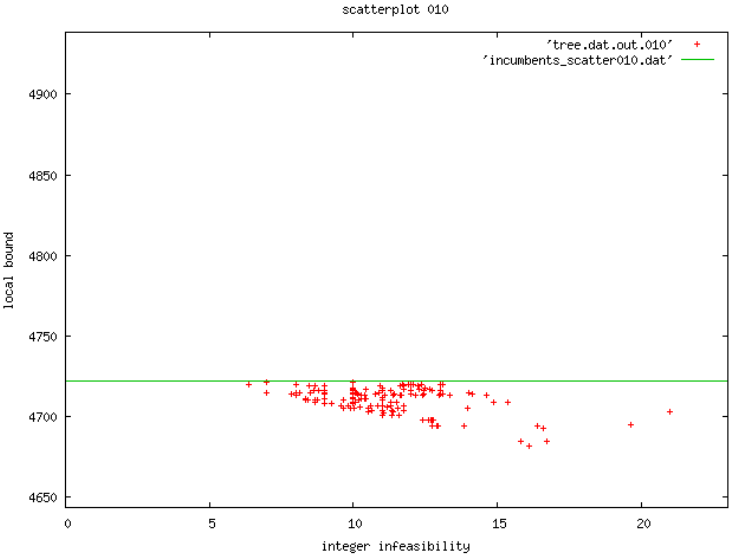
Example scatterplot series



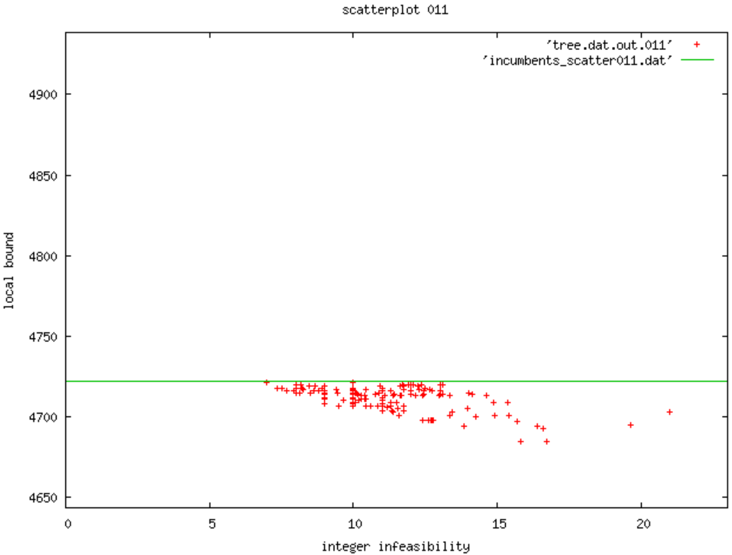
Example scatterplot series



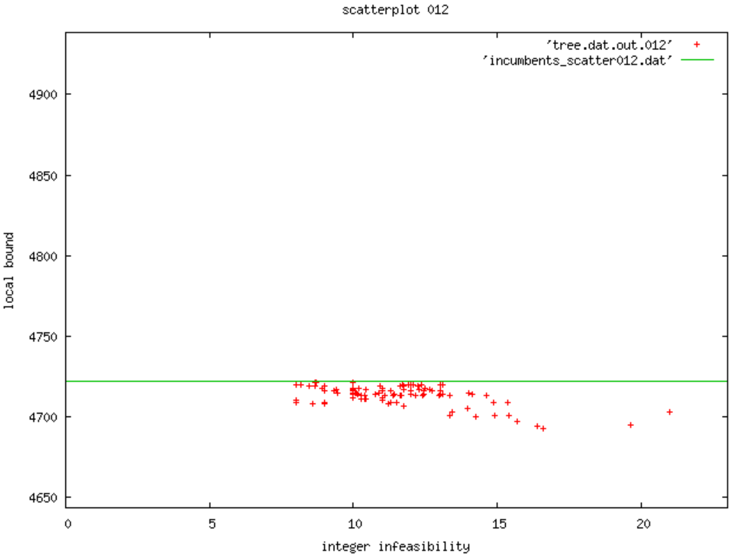
Example scatterplot series



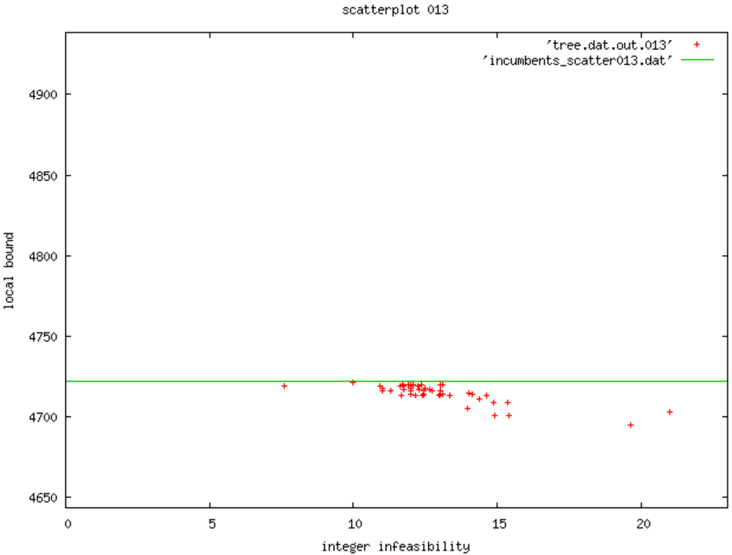
Example scatterplot series



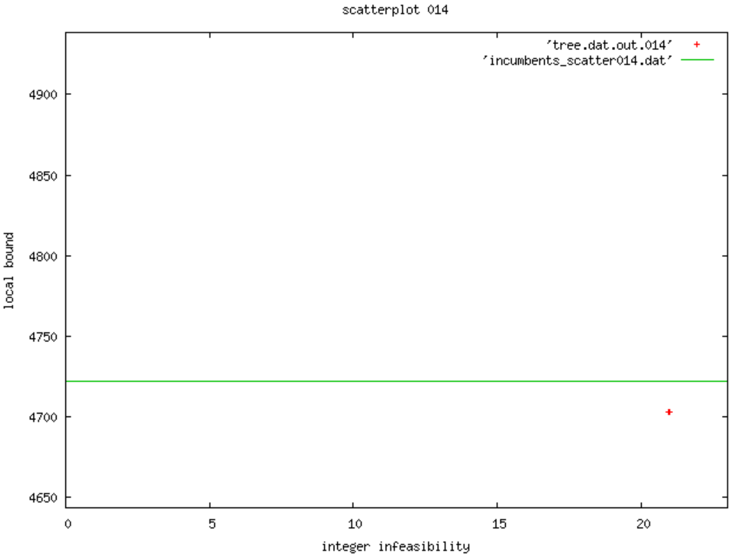
Example scatterplot series



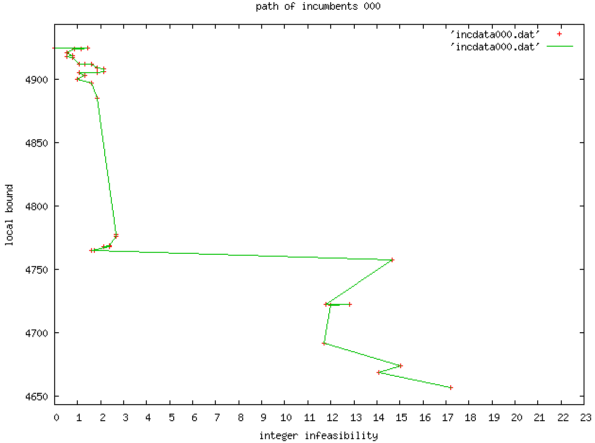
Example scatterplot series



Example scatterplot series

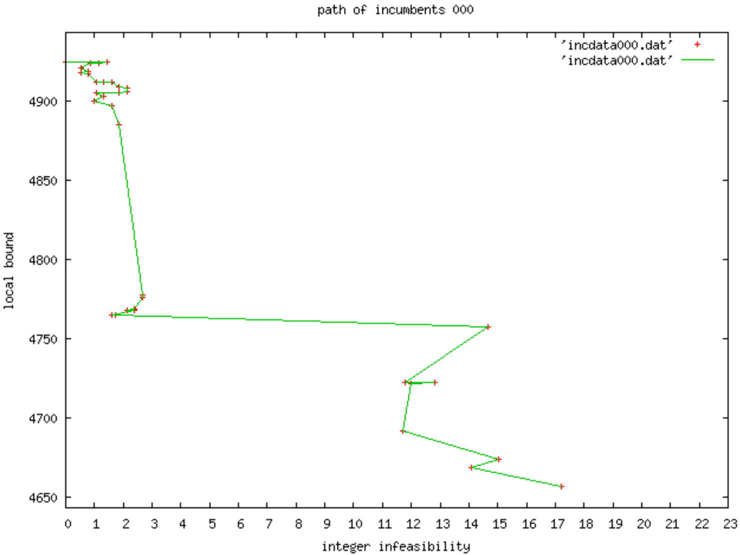


History of active nodes

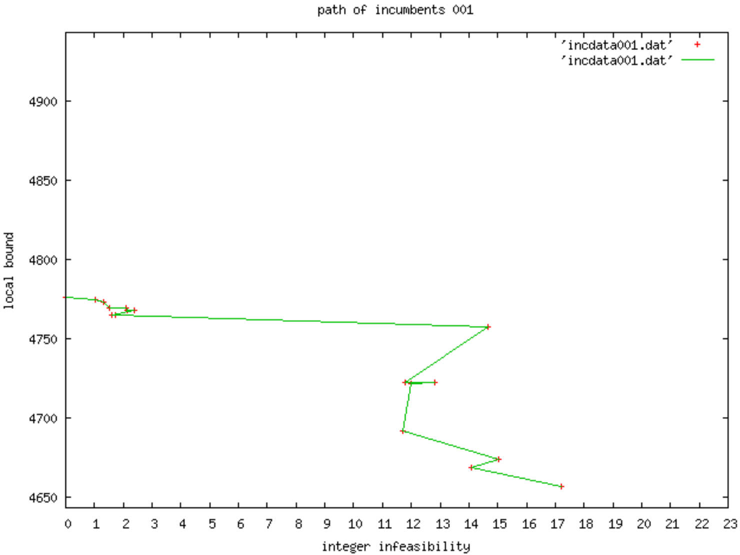


- Shows the ancestors of the node

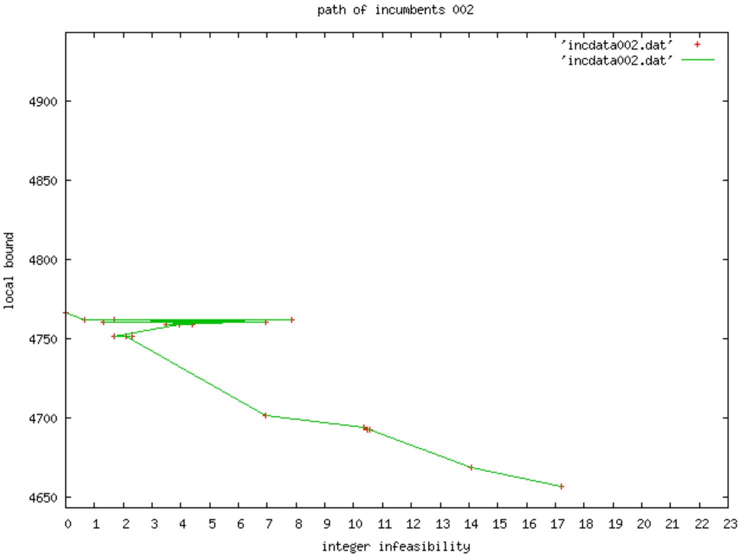
Example histories of active nodes



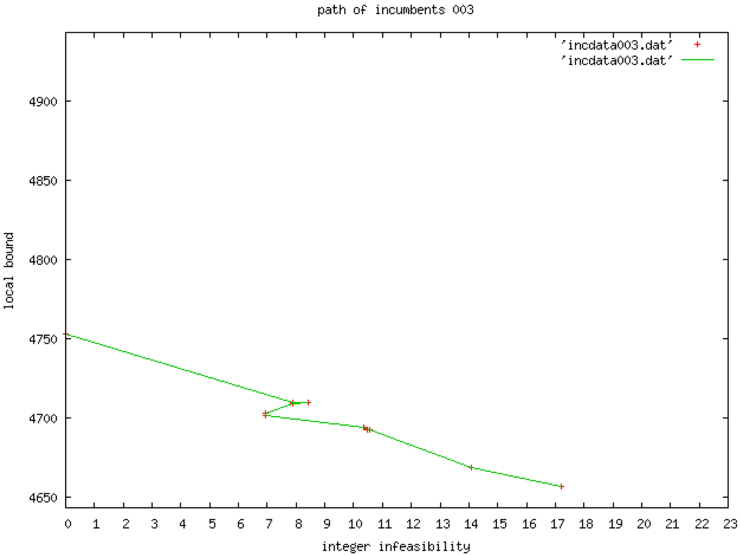
Example histories of active nodes



Example histories of active nodes



Example histories of active nodes



Measure of progress

- The user would like a good measure of progress

Measure of progress

- The user would like a good measure of progress
- Gap and number of active nodes don't work well

Measure of progress

- The user would like a good measure of progress
- Gap and number of active nodes don't work well
- Histograms give good information, but we want a single value

Measure of progress

- The user would like a good measure of progress
- Gap and number of active nodes don't work well
- Histograms give good information, but we want a single value
- One idea: sum of gaps

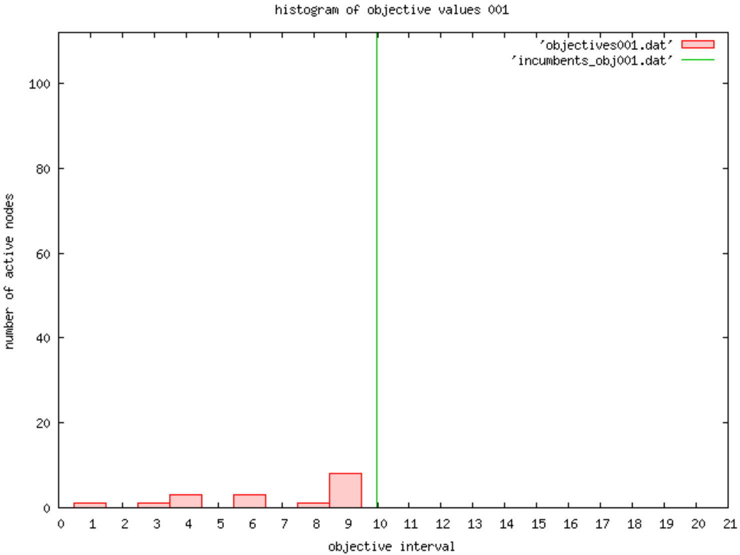
Measure of progress

- The user would like a good measure of progress
- Gap and number of active nodes don't work well
- Histograms give good information, but we want a single value
- One idea: sum of gaps
- But this fluctuates a great deal (with the number of active nodes)

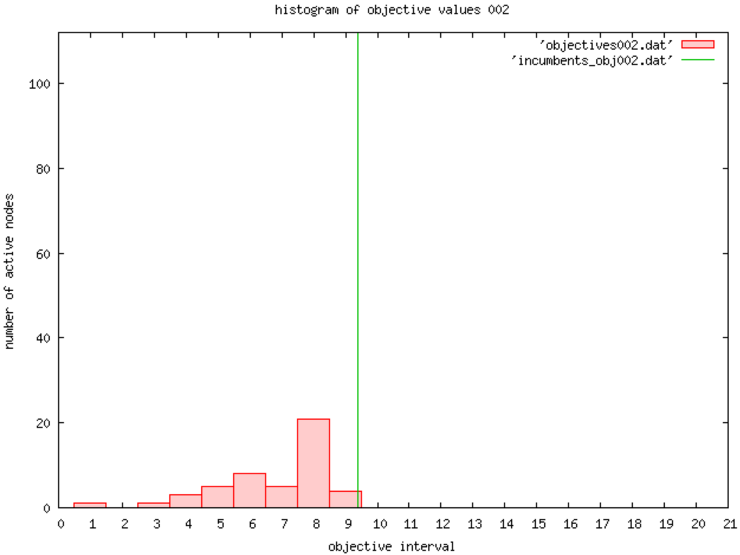
Measure of progress

- The user would like a good measure of progress
- Gap and number of active nodes don't work well
- Histograms give good information, but we want a single value
- One idea: sum of gaps
- But this fluctuates a great deal (with the number of active nodes)
- Another idea: average gap

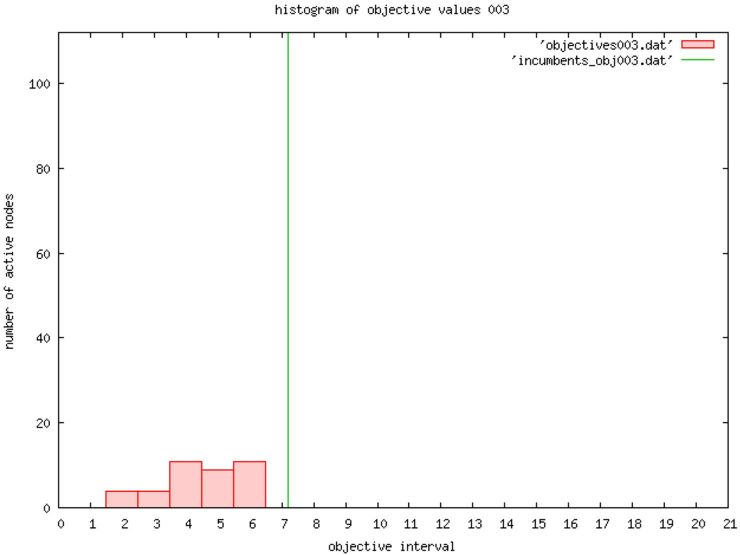
Measure of progress: Average gap doesn't work well



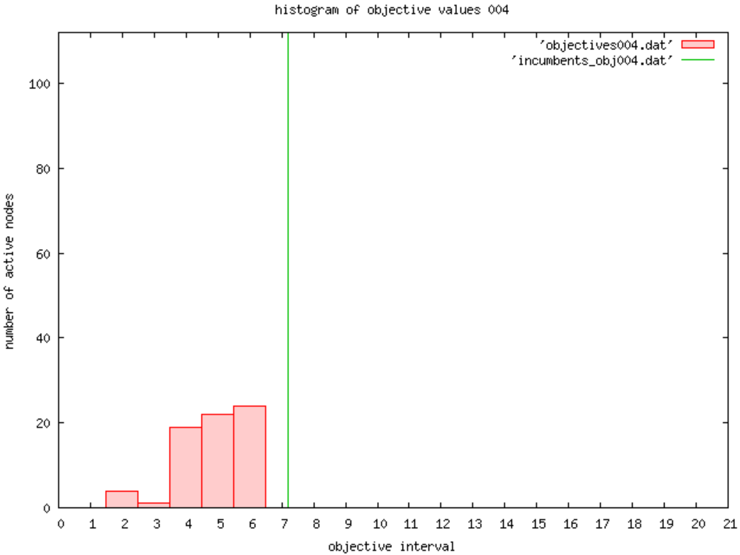
Measure of progress: Average gap doesn't work well



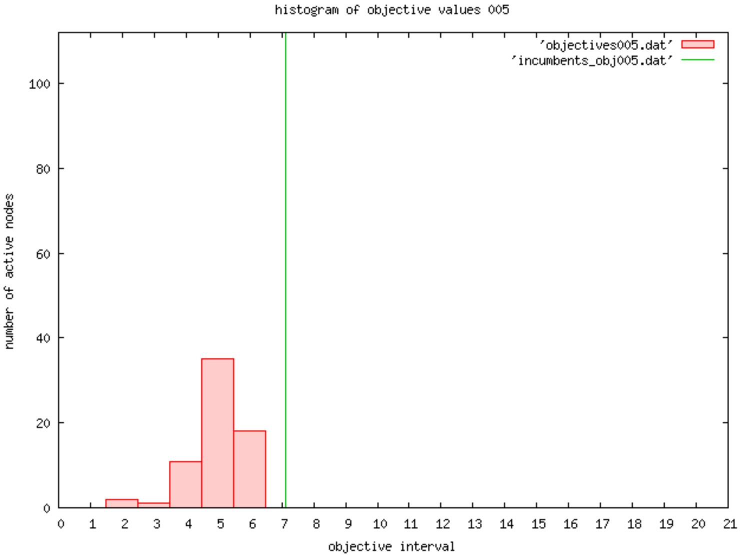
Measure of progress: Average gap doesn't work well



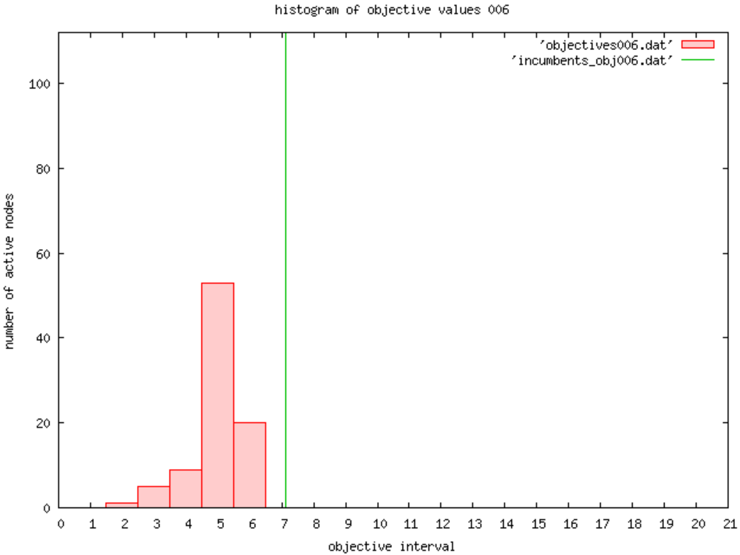
Measure of progress: Average gap doesn't work well



Measure of progress: Average gap doesn't work well



Measure of progress: Average gap doesn't work well



Measure of progress: Weighted sum of active node gaps

- Current measure: Weight each node based on depth. Let A be the set of active nodes, g_i be the gap for node i, and $depth_i$ be the depth of node i:

$$\sum_{i \in A} \frac{g_i}{2^{d_i}}$$

Measure of progress: Weighted sum of active node gaps

- Current measure: Weight each node based on depth. Let A be the set of active nodes, g_i be the gap for node i, and $depth_i$ be the depth of node i:

$$\sum_{i \in A} \frac{g_i}{2^{d_i}}$$

- Valuable properties:

Measure of progress: Weighted sum of active node gaps

- Current measure: Weight each node based on depth. Let A be the set of active nodes, g_i be the gap for node i, and $depth_i$ be the depth of node i:

$$\sum_{i \in A} \frac{g_i}{2^{d_i}}$$

- Valuable properties:
  - Sum of weights of children equals parent's weight

Measure of progress: Weighted sum of active node gaps

- Current measure: Weight each node based on depth. Let A be the set of active nodes, g_i be the gap for node i, and $depth_i$ be the depth of node i:

$$\sum_{i \in A} \frac{g_i}{2^{d_i}}$$

- Valuable properties:
  - Sum of weights of children equals parent's weight
  - Weights are constant

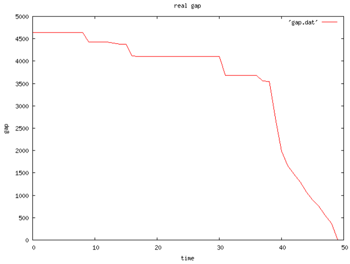
Measure of progress: Weighted sum of active node gaps

- Current measure: Weight each node based on depth. Let A be the set of active nodes, g_i be the gap for node i, and $depth_i$ be the depth of node i:

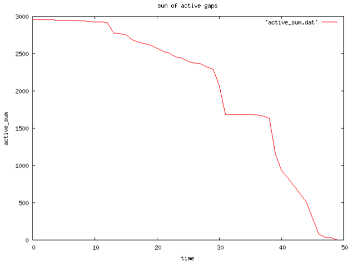
$$\sum_{i \in A} \frac{g_i}{2^{d_i}}$$

- Valuable properties:
  - Sum of weights of children equals parent's weight
  - Weights are constant
  - Therefore: Monotonic decreasing (as long as lp bounds of parent and child differ)

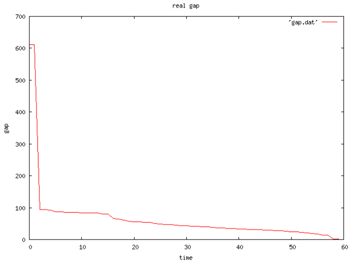
Example graphics



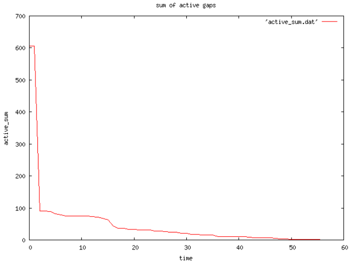
bell3a, CBC no cuts



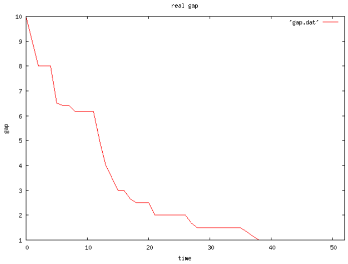
Example graphics



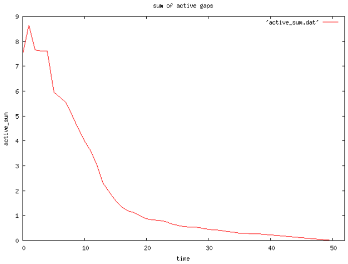
l152lav, CBC default



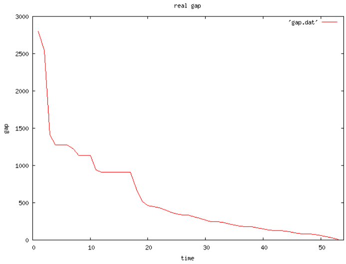
Example graphics



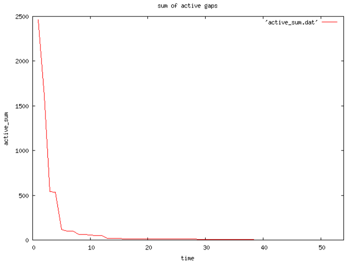
stein45, CBC default



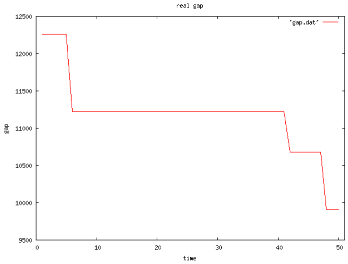
Example graphics



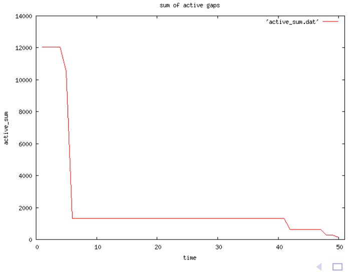
misc07, CBC default



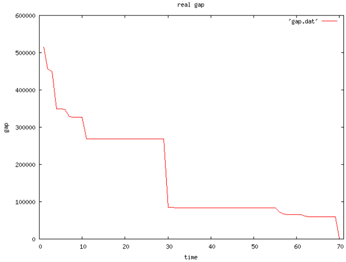
Example graphics



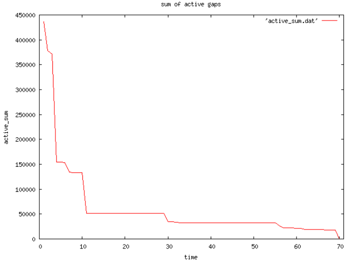
bell3a, GLPK intopt no cuts



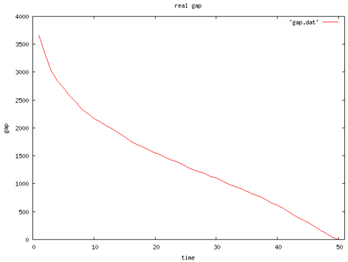
Example graphics



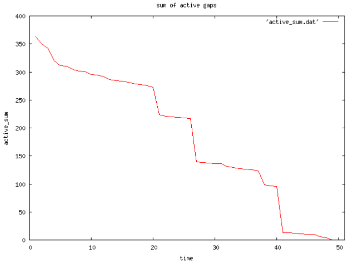
bell5, GLPK intopt no cuts



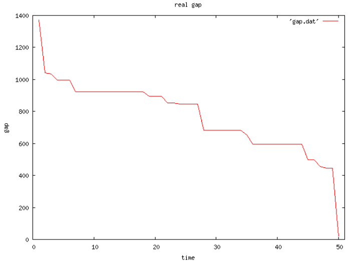
Example graphics



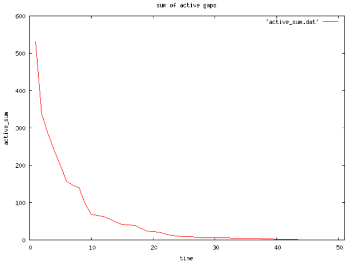
bell3a, GLPK standard, best bound



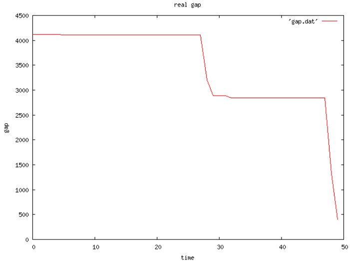
Example graphics



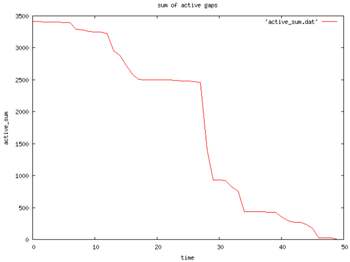
misc07, GLPK standard



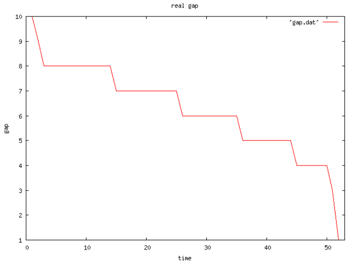
Example graphics



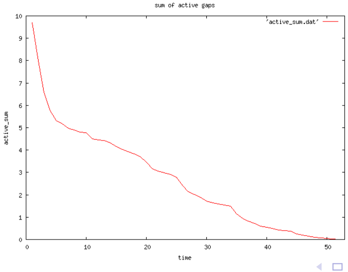
bell3a, CBC default



Example graphics



stein45, GLPK intopt with cuts



Strengths and weaknesses

- Strengths

Strengths and weaknesses

- Strengths
  - monotonic decreasing whenever child LP bound differs from parent

Strengths and weaknesses

- Strengths
  - monotonic decreasing whenever child LP bound differs from parent
  - generally smoother measure of progress

Strengths and weaknesses

- Strengths
  - monotonic decreasing whenever child LP bound differs from parent
  - generally smoother measure of progress
  - appears robust to different solvers and options

Strengths and weaknesses

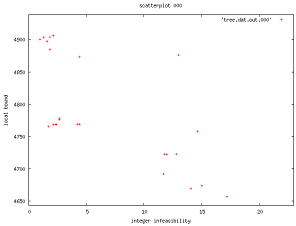
- Strengths
  - monotonic decreasing whenever child LP bound differs from parent
  - generally smoother measure of progress
  - appears robust to different solvers and options
- Weakness: still drops significantly when new incumbents found

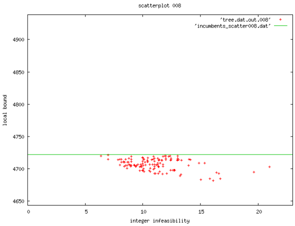
Moving forward: Predicting new incumbents

- To smooth the graph more, one approach is to anticipate new incumbents

Moving forward: Predicting new incumbents

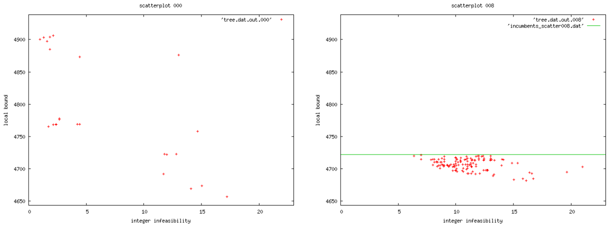
- To smooth the graph more, one approach is to anticipate new incumbents
- This is already done in the default node selection strategies (both based on best projection)





Moving forward: Predicting new incumbents

- To smooth the graph more, one approach is to anticipate new incumbents
- This is already done in the default node selection strategies (both based on best projection)



- However, the predictions do not appear to be consistently accurate, especially for big drops

Conclusions

- We have lots of information about b&b progress

Conclusions

- We have lots of information about b&b progress
- Valuable to represent data visually when considering summary measures

Conclusions

- We have lots of information about b&b progress
- Valuable to represent data visually when considering summary measures
- We should explore more data mining/machine learning applied to MIPs

Conclusions

- We have lots of information about b&b progress
- Valuable to represent data visually when considering summary measures
- We should explore more data mining/machine learning applied to MIPs
- Value of open-source codes: proven useful on real-world problems and allow full and easy access to information available during the algorithm

Current and Future work

- Finish examining measures of progress and publish these ideas, including code to generate graphics

Current and Future work

- Finish examining measures of progress and publish these ideas, including code to generate graphics
- Community may benefit from examining patterns

Current and Future work

- Finish examining measures of progress and publish these ideas, including code to generate graphics
- Community may benefit from examining patterns
- Estimate likelihood of better integer solutions

Current and Future work

- Finish examining measures of progress and publish these ideas, including code to generate graphics
- Community may benefit from examining patterns
- Estimate likelihood of better integer solutions
- Can other information be extracted: recommended node selection strategy or cuts?