

# Visualization and ILOG CPLEX

Georg Sander and Adrian Vasiliiu

ILOG SA, 9 rue de Verdun – BP 85, 94253 Gentilly Cedex, France  
{sander,vasiliu}@ilog.fr  
<http://www.ilog.com>

## 1 Introduction

Graph layout methodologies often solve difficult subproblems in order to satisfy the aesthetic constraints: finding the maximal planar subgraph, minimizing crossings, minimizing area, maximizing symmetries, etc. One standard approach is to translate the subproblem into a linear optimization problem (LOP) and to use a standard mathematical tool to find the solution. The standard solving tool for such LOPs is ILOG CPLEX.

ILOG CPLEX delivers high-performance, robust, flexible optimizers for solving linear, mixed-integer and quadratic programming problems (including mixed integer quadratic constrained problems). It is a component that includes C, C++ and Java API, and it is integrated via the Concert Technology into the ILOG Optimization Suite.

With ILOG JViews, we also offer a visualization suite that includes sophisticated graph layout algorithms. Our recent investigations focused on cross-product development and the question, how ILOG CPLEX can be used to help graph layout and how ILOG JViews can be used to help the LOP solving.

## 2 How ILOG CPLEX Helps Graph Layout

ILOG CPLEX can be used to solve various subproblems in graph layout. The following is a collection of algorithms found in the literature that make extensive use of ILOG CPLEX:

- Link Crossing – Jünger and Mutzel (GD'95, LNCS 1027, p. 337ff) compare a CPLEX implementation of the LOP for the 2-layer straightline crossing minimization problem with various other heuristics.
- Labeling Problem – Binucci e.a. (GD'02, LNCS 2528, p. 66ff) show an LOP algorithm to compute optimal label positions in an orthogonal drawing.
- Detecting Symmetries – Buchheim and Jünger (GD'01, LNCS 2265, p. 178ff) use ABACUS and CPLEX to detect automorphisms and symmetries in arbitrary graphs.
- Graph Layering Problem – Healy and Nikolov (GD'01, LNCS 2265, p. 16ff) partition a DAG into layers as needed for the hierarchical layout algorithm. Again, CPLEX was used as reference implementation.

- Orthogonal Layout Compaction – Klau and Mutzel (IPCO’99, LNCS 1610, p. 304ff) compress an orthogonal layout so that the edges length are minimized by applying a branch and cut approach that can be implemented with CPLEX.
- and many more.

### 3 How ILOG JViews Graph Layout Helps LOP Solving

ILOG CPLEX uses nonvisual algorithms on a mathematical model of the problem to be solved. Visualization is used to help CPLEX users to model their problems, and to detect or debug the internal behavior of the CPLEX routines.

- ILOG OPL Studio – This is the modeling tool for the ILOG Optimization Suite. It has a graphical GUI to visualize scheduling problems and their solutions.
- Branch and Bound Tree – Mixed integer problem are often solved by branch and bound. The branching structure is essentially a binary tree. Graph layout technology helps to visualize this tree. The RINS algorithm (Relaxation Induces Neighborhood Search) is a local search for feasible solutions of mixed integer problems and can be visualized this way.
- Precedence graph of linear equations – Many subroutines of CPLEX solve linear equation systems with sparse matrix. A precedence graph helps to analyze in which order the variables must be calculated. A reduction mechanism of the precedence graph led to a major performance boost of CPLEX. The precedence graph can be visualized.
- Parallel CPLEX – CPLEX can be executed on multiple processor machines. The debug trace of parallel CPLEX can be visualized, which is essentially a graph.

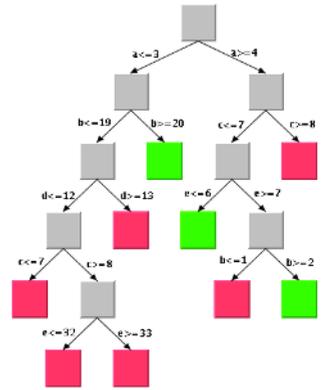
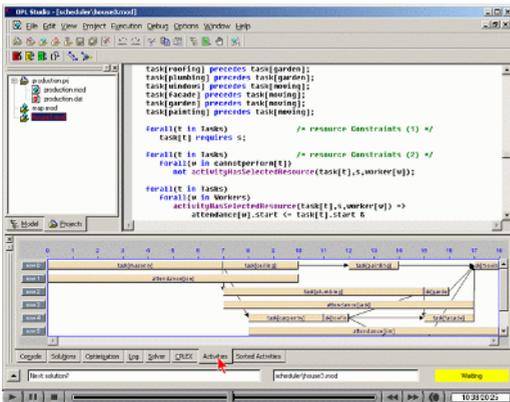


Fig. 1. Left: ILOG OPL Studio, Right: Branch and Bound Tree