भारतीय प्रबंध संस्थान बेंगलूर
INDIAN INSTITUTE OF MANAGEMENT
BANGALORE

The Decision Sciences Area at IIM Bangalore welcomes you to a webinar, titled:

'Optimizing Integer Programming Problems using a Branch-Line-and-Search (BLS) Algorithm'

by
Md Shahrukh Anjum

Date: 6th August, 2021
Time: 3:00 p.m. to 4:00 p.m.

Abstract:

It is well recognized that finding a feasible solution to a mixed-integer program (MIP) is a computationally difficult problem in practice. While there have been several attempts to improve the bound resulting from the relaxations at a given node within a branch-and-bound algorithm (via Lagrangian, Surrogate, Elimination, etc.), there has been relatively scant attention paid to generating good quality feasible solutions relatively early in the branch-and-bound process. In this work, we develop a novel heuristic to quickly generate incumbent solutions, which in turn helps fathom suboptimal branches early on in a branch-and-bound process (or any other equivalent search algorithm) and reduces the computational effort and memory required in finding the optimal solution. The proposed Branch-Line-and-Search (BLS) algorithm can be broadly categorized as an interior point line-search method built on the principle of tracing collinear integer points along the line segment (referred to as BLS line) formed by joining two integer points, called the base points in an integer lattice. These base points, obtained by a specific rounding scheme of the different nodes of the search tree, need not satisfy the MIP feasibility constraints. The BLS line can be constructed from the combination of two feasible and/or infeasible points. The nature of feasibility and location of each base point derives the relative position of the potential feasible collinear integer points in the lattice. The BLS line search is carried out at every iteration of the branching of the regular branch-and-bound algorithm and terminates as the certificate of optimality is obtained. We illustrate the efficacy of the proposed algorithm by solving a numerical example and testing on a set of generic MIP problems from MIPLIB.