

Column Generation Techniques for GAP

January 25, 2019

1 Literature Review

- Branch-and-Price : Column Generation for Solving Huge Integer Programs (Barnhart, et. al., 1998)
- A Branch-and-Price Algorithm for the Generalized Assignment Problem (Savelsbergh and Martin, 1997)
 - Dantzig-Wolfe decomposition for GAP (construct master and sub problems)
 - Column generation : additional columns of the restricted master problem are generated by solving the pricing problem.
 - Branching strategies : variable dichotomy(sing variable) and GUB dichotomy(set of variables)
- Chebyshev center based column generation (Lee and Park, 2011)
 - The column generation procedure based on the simplex algorithm often shows desperately slow convergence. (zig-zag movement)
 - Chebyshev center based column generation techniques
 - * Chebyshev center
 - * Proximity adjusted Chebyshev center
 - * Chebyshev center + Stabilization
 - * Proximity adjusted Chebyshev center + Stabilization
 - Computational experiments on the binpacking, VRP, GAP
 - The proposed algorithm could accelerate the column generation procedure.
- Comparison of bundle and classical column generation (O.Briant, et. al., 2006)
 - Bundle method : the dual solution is often constrained to a given interval, and any deviation from the interval is penalized by a penalty function.
 - The penalty function for stabilized column generation : a simple V-shaped function (stabilizing center, ϵ)
- Stabilized Column Generation (O. Du Merle, et. al., 1997)

2 Problems

2.1 A case study : Generalized Assignment Problem

Dantzig-Wolfe Decomposition ¹

$$\begin{aligned}
 \text{(P)} \quad & \min \sum_{i \in I} \sum_{k \in K} c_k^i x_k^i, \\
 \text{s.t.} \quad & \sum_{i \in I} \sum_{k \in K_i} \delta_k^j x_k^i \geq 1, \quad j \in J, \\
 & - \sum_{k \in K_i} x_k^i \geq -1, \quad \forall i \in I, \\
 & x_k^i \geq 0, \quad \forall k \in K_i, i \in I. \\
 \\
 \text{(D)} \quad & \max \sum_{j \in J} \pi_j - \sum_{i \in I} \phi_i, \\
 \text{s.t.} \quad & \sum_{j \in J} \delta_k^j \pi_j - \phi_i \leq c_k^i, \quad \forall k \in K_i, i \in I, \\
 & \pi_j \geq 0, \quad \forall j \in J, \\
 & \phi_i \geq 0, \quad \forall i \in I.
 \end{aligned}$$

The GAP oracle finds an assignment pattern while satisfying the knapsack constraints :

$$\max \sum_{j \in J} (\pi_j - c_{ij}) \delta_j, \quad \text{s.t.} \quad \sum_{j \in J} a_{ij} \delta_j \leq b_i, \delta_j \in \{0, 1\}, \quad \forall j \in J$$

Stabilization

$$\begin{aligned}
 (\tilde{P}) \quad & \min \sum_{i \in I} \sum_{k \in K} c_k^i x_k^i + \sum_{j \in J} \delta_j (\gamma_j^+ - \gamma_j^-) + \sum_{i \in I} \phi_i (y_i^+ - y_i^-), \\
 \text{s.t.} \quad & \sum_{i \in I} \sum_{k \in K_i} \delta_k^j x_k^i + \gamma_j^+ - \gamma_j^- \geq 1, \quad \forall j \in J, \\
 & - \sum_{k \in K_i} x_k^i + y_i^+ - y_i^- \geq -1, \quad \forall i \in I, \\
 & \gamma_j^+ \leq \epsilon, \quad \gamma_j^- \leq \epsilon, \quad \forall j \in J, \\
 & y_i^+ \leq \epsilon, \quad y_i^- \leq \epsilon, \quad \forall i \in I, \\
 & x_k^i \geq 0, \quad \forall k \in K_i, i \in I, \\
 & y_i^+ \geq 0, \quad \forall k \in K_i, i \in I.
 \end{aligned}$$

3 Preliminary Tests

Github page : <https://github.com/mody3062/CG>

¹The written mathematical formulation are from (Lee and Park, 2011)

Testing algorithms

- Classical column generation (Kelly's cutting plane)
- Stabilized column generation (O. Du Merle, et. al., 1997)

All codes for the both of algorithms are based on the pseudo code described in Figure 1 of (O. Du Merle, et. al., 1997).

Algorithmic parameters RMP was constructed with a single decision variable which is dummy. The coefficient of the dummy variable on the objective function was set to a sufficiently large value, which is the sum of listed values such that `np.sum(c,axis=1)`. For stabilized column generation algorithm, I fixed the parameter ϵ to 0.0001. (\because I don't understand the criteria for changing the parameter value(ϵ).)