INFORMS JOURNAL ON **COMPUTING**

A Restricted Dual Peaceman-Rachford Splitting Method for a Strengthened DNN Relaxation for QAP

Important: This code is being developed on an on-going basis at https://github.com/Xinxin-opt/2020.0336. Please use the link if you would like to get the most up-to-date version.

Cite

To cite this software, please cite the paper using its DOI and the software itself, using the following DOI.

DOI 10.5281/zenodo.3977566

Below is the BibTex for citing this version of the code.

```
@article{QAP2021PRSM,
  author =        {N. Graham, H. Hu, J. Im, X. Li and H. Wolkowicz},
  publisher =     {INFORMS Journal on Computing},
  title =         {rPRSM Version v1.0},
  year =          {2021},
  doi =           {10.5281/zenodo.3977566},
  url =           {https://github.com/INFORMSJoC/2020.0336},
}
```

Description

The goal of this software is to solve a doubly nonnegative **(DNN)** relaxation for the quadratic assignment problem **(QAP)** :

$$p^*_{\text{QAP}} := \min_{X\in\Pi} \operatorname{trace}(AXBX^T),$$

where A is the flow matrix, B is the distance matrix, and Π denotes the set of n × n permutation matrices, i.e.,

$$\Pi = \{X \in \mathbb{R}^{n\times n} : Xe = e, X^Te = e, X_{ij} \in \{0,1\}\}.$$

Users can provide problem instances in three ways to our software:

1. Users can provide their own instance;

2. Users can generate random instances;
3. Users can directly use the instance from QAPLIB.

Users can use the script file *main.m* to run tests using our software. To run tests, place the data file (if there are any) in the script folder. Before running the *main.m* file in Matlab, modify *main.m* properly to navigate your data and choose the right datetype. This file calls *qrun_tests.m* in src folder. Results can be found in results folder.

Important: A user provided data must meet the following requirements: the data matrices A,B are nonnegative, INTEGER valued and n-by-n symmetric matrices

Contents

script folder	
main.m	calls all relavant routines for testing

src folder	
qrun_tests.m	sets up options, calls the solver PRSM or ADMM, and outputs relevant information.
ADMM_QAP.m	solves the **DNN** relaxation using ADMM
PRSM_QAP.m	solves the **DNN** relaxation using PRSM
simplex_proj.m	projects a vector onto the simplex
sec2hms.m	converts seconds into hours-minutes-seconds
proj_dstochastic.m	projects a vector onto the set of doubly stochastic matrices

data folder	
large_instances	contains large size instances from QAPLIP
medium_instances	contains medium size instances from QAPLIP
small_instances	contains small size instances from QAPLIP
Optimal_values.m	contains optimal/best konwn bounds for each dataset

results folder	
results.mat	a struct Out. contains various information and a variable Y from the solver

Results

Outputs in .mat	
Y	optimal solution of **DNN** relaxation to **QAP**
Out.obj	history of trace(L*Y)
Out.iter	total number iterations
Out.feas	history of residual norm(Y-VRV, 'fro')/norm(Y, 'fro')
Out.pR	history of primal residual, norm(Y-VRV, 'fro')
Out.dR	history of dual residual, norm(Y-Y0, 'fro')
Out.Z	final dual variable Z
Out.R	final primal variable R
Out.Vhat	Vhat
Out.bestiter	last iteration that yields best bound
Out.ubest	best upper bound
Out.lbest	best lower bound
Out.L	modified objective function data L, trace(L*Y), after scaling and shifting
Out.Lorig	original objective function data L, trace(Lorig*Y)
Out.scale	scaling factor of the objective
Out.shift	shifting parameter of the objective
Out.ubdtime	time spent on computing upper bounds
Out.lbdstoptime	first time when the solver lower bound met the user provided lower bound (=opts.lbdstop). If it is never met, outputs -100.