STAT 540 Project:

Efficient Computational Methods for Discrete Optimization via Simulation

Xinru Li IME department

Overview of the project

- Discrete Optimization via Simulation (DOvS):

$$\min\ y(\mathbf{x}) \triangleq \mathrm{E}[Y(\mathbf{x})] \text{ subject to } \mathbf{x} \in \boldsymbol{\mathcal{X}}$$

where $\boldsymbol{\mathcal{X}}$ is a finite subset of *d*-dimensional integer lattice $\mathbb{Z}^d$.

Distribution of *Y*(x) unknown, and the expectation can be estimates using stochastic simulation.

"Sherman-Morrison-Woodbury"

- Main challenge:

In the algorithm, it needs diagonal and one column of the inverse of a matrix Q, whose size depends on $\boldsymbol{\mathcal{X}}$.

Example. If the feasible solution is n^4, then Q is $n^4 \times n^4$.

Q^{-1} needed every iteration, difference between two iterations is small. It wastes lots of time.

Gaussian Markov Random Field (GMRF)

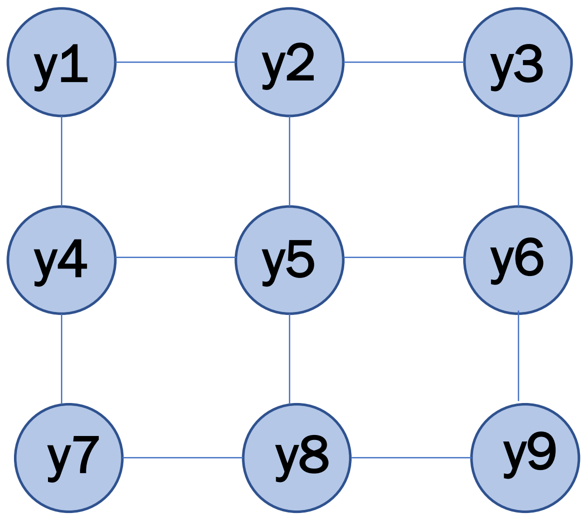
- A GMRF is a non-degenerate multivariate Gaussian random vector Y that is associated with an undirected and labeled graph.
- The pdf of the GMRF:

$$f(\mathbf{y} \mid \mu, \mathbf{Q}) = (2\pi)^{-n/2} |\mathbf{Q}|^{1/2} \exp\left(-\frac{1}{2}(\mathbf{y}-\mu)^\top \mathbf{Q}\,(\mathbf{y}-\mu)\right)$$

- "Markov" property:

$$\mathbb{Y}_i \perp \mathbb{Y}_{\mathcal{V}\setminus\{i,\mathcal{N}(i)\}} \mid \mathbb{Y}_{\mathcal{N}(i)} \quad \text{for every} \quad i \in \mathcal{V}.$$

- Salemi et al. (2017) firstly used GMRF solving DOvS problems.
- In that paper they proposed Gaussian Markov Improvement algorithm (GMIA).



Undirected and labeled graph
G = (V, E)

$$\mathcal{N}(\mathbf{x}) = \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x}-\mathbf{x}'\|_2 = 1\}$$

Precision matrix Q

- As the neighbor structure given last page,

$$\mathrm{Q}_{ij} \triangleq p(\mathbf{x}_i, \mathbf{x}_j\, ; \boldsymbol{\theta}) = \begin{cases} \boldsymbol{\theta}_0, & \text{if } \mathbf{x}_i = \mathbf{x}_j \\ -\boldsymbol{\theta}_0\boldsymbol{\theta}_k, & \text{if } |\mathbf{x}_i - \mathbf{x}_j| = \mathbf{e}_k \\ 0. & \text{otherwise} \end{cases}$$

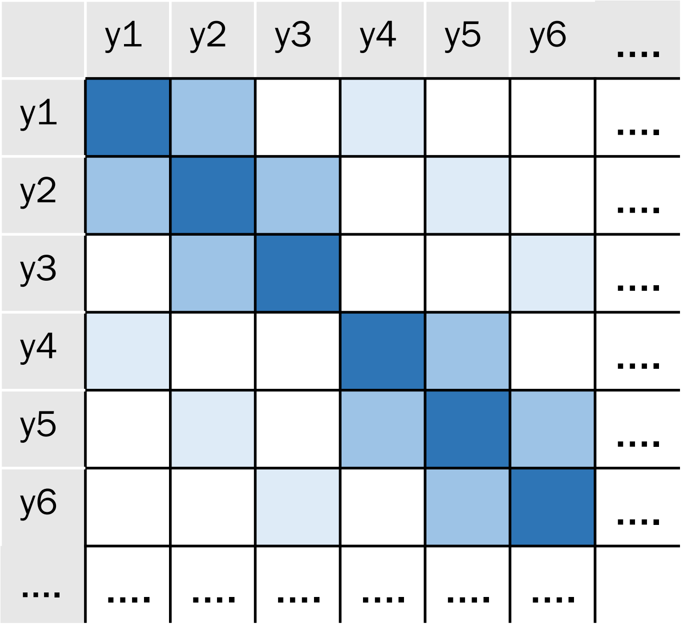
- Example:

$\boldsymbol{\theta}_0$ $\quad -\boldsymbol{\theta}_0\boldsymbol{\theta}_1$ $\quad -\boldsymbol{\theta}_0\boldsymbol{\theta}_2$ $\quad 0$

- Diagonals are precision, so during the method, the only changing part is the diagonals of Q.
- Important Properties of Q

➢ Q is sparse.

➢ Q is PD $\rightarrow \sum_{i=0}^{d} \boldsymbol{\theta}_i < 0.5$. (If d is large, then $\boldsymbol{\theta}_i$s are very small)



Structure of example precision matrix for the graph last page

Gaussian Markov Improvement Algorithm (GMIA)

[From Salemi et al. (2017)]

General idea:

1. Simulate a set of initial design points.
2. Get the GMRF parameters through MLE.
3. Find current optimal point $\tilde{\boldsymbol{x}}$ and max CEI point $\boldsymbol{x}_{\mathrm{CEI}}$, then simulate these two points.
4. Update the simulation output and check stopping criterion. If not satisfied, go to step 3.

Most expensive

- Complete expected improvement (CEI) [Jones et al. (1998)]

$$\begin{aligned}\mathrm{CEI}^t(\mathbf{x}) &= \mathrm{E}[\max(\mathbb{Y}(\tilde{\mathbf{x}}^t) - \mathbb{Y}(\mathbf{x}), 0)] \\ &= \left(M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})\right)\Phi\left(\frac{M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})}{\sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})}}\right) + \sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})}\ \phi\left(\frac{M^t(\tilde{\mathbf{x}}^t) - M^t(\mathbf{x})}{\sqrt{V^t(\tilde{\mathbf{x}}^t, \mathbf{x})}}\right)\end{aligned}$$

$$V^t(\tilde{\mathrm{x}}^t, \mathrm{x}) = V^t[\mathbb{Y}(\tilde{\mathrm{x}}^t) - \mathbb{Y}(\mathrm{x})] = V^t(\tilde{\mathrm{x}}^t) + V^t(\mathrm{x}) - 2C^t(\tilde{\mathrm{x}}^t, \mathrm{x})$$

Improved GMIA using SMW formula

- $e_{x^*} = (0,0,\ldots,0,1,0,\ldots,0)$ is a *d*-dimension vector, 1 corresponds to the column of x^*.
- The precision matrix Q is updated:

$$\mathbf{Q}^{t+1} = \mathbf{Q}^t + \Delta_1 \mathbf{e}'_{\mathbf{x}_{\mathrm{CEI}}} \mathbf{e}_{\mathbf{x}_{\mathrm{CEI}}} + \Delta_2 \mathbf{e}'_{\tilde{\mathbf{x}}} \mathbf{e}_{\tilde{\mathbf{x}}}$$

- Since there are always numerical errors, cumulating the changes for a period and compute the inverse from Q^t. (Still, I just need the diagonal and one column of the inverse)

- n = # all feasible solutions
- k = # cumulated changes (k << n)
- Then the size of Q is $n \times n$ and the size of U,V is $k \times n$.

SMW formula:

$(Q + UV')^{-1} = Q^{-1} + \cdots$

- From the experiment, improved GMIA really takes much less time compared with GMIA.

Some ideas from the failed simulation experiments

- When using SMW, need to check numerical errors. For my experiments, entries are small so the relative error will influence the final results a lot if using SMW each iteration.

  One way to solve:

  Set periods.

  At the beginning of a period, use Cholesky to compute an inverse Q^{-1}.

  During this period, cumulate the changes and compute the inverse using Q^{-1}.

- When computing the inverse of a matrix whose entries is pretty small,

  LU decomposition is more stable than Cholesky decomposition.

  But the cost of LU decomposition is $\frac{2}{3}n^3$, and the cost of Cholesky decomposition is $\frac{1}{3}n^3$.

Short comparison with continuous Gaussian

- Advantage for continuous Gaussian:

  Continuous methods just need the sampled points to construct the variance matrix (dense),

  so at the beginning of the algorithm, the matrix is small, computation is cheap.

- Disadvantage:

  The variance matrix is dense, so after long running, the matrix will become large,

  which makes computational cost expensive.

- Compared with continuous Gaussian, GMRF always use the same size of precision matrix, which depends on the size of the problem. In which, the advantage is the sparsity of the precision matrix. After implementing SMW, it will reduce the time dramatically.
- Results from continuous GRF coming soon…

Hope after my presentation, you can …

- have an general idea about GMRF, and the GMIA, which is specially for discrete optimization via simulation problems.
- Take some small ideas I had in my simulation experiment.
- Learn a new way to solve the numerical error.

In project …

- including more details about GMRF, GMIA and improved GMIA.
- Introducing continuous GRF-based algorithm to compare with GMIA and improved GMIA.
- Complete simulation experiments.
- Results of comparisons between continuous and GMIA.

If you are interested in my topic, welcome to read my project paper.

Thanks for listening! ☺