Personalization

Talk 5: GADMM

WEN Hao

2021-6-24

GADMM – Origin

Performance of distributed optimization algorithms is characterized by

- computation time (complexity)
- communication time (especially in cross-device scenarios)

GADMM – Origin

Performance of distributed optimization algorithms is characterized by

- computation time (complexity)
- communication time (especially in cross-device scenarios)

Communication time is determined by

- # communication rounds
- # channels (edges in the graph) per round
- bandwidth/power (data transmitted) per channel

GADMM – Origin

Performance of distributed optimization algorithms is characterized by

- computation time (complexity)
- communication time (especially in cross-device scenarios)

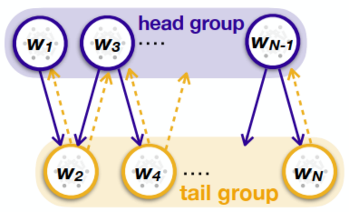
Communication time is determined by

- # communication rounds
- # channels (edges in the graph) per round
- bandwidth/power (data transmitted) per channel

And moreover, to fit the totally distributed (decentralized) clients network topology.

GADMM – Formulation

The Group ADMM (GADMM) algorithm [1] considers the following clients network topology



where all clients are divided into 2 groups, i.e. head group $\mathcal{N}_h$ and tail group $\mathcal{N}_t$. The problem hence is formulated as

$$\begin{aligned} &\text{minimize} \quad \frac{1}{N}\sum_{i=1}^{N} f_i(x_i) \\ &\text{subject to} \quad x_i = x_{i+1}, \quad i = 1, \cdots, N-1 \end{aligned}$$

GADMM – Algorithm

head group primal update (and transmit → tail group)

$$x_i^{k+1} = \arg\min_{x_i} \{ f_i(x_i) + \langle \lambda_{i-1}^k, x_{i-1}^k - x_i \rangle + \langle \lambda_i^k, x_i - x_{i+1}^k \rangle + \frac{\rho}{2} \| x_{i-1}^k - x_i \|^2 + \frac{\rho}{2} \| x_i - x_{i+1}^k \|^2 \}, \quad i \in \mathcal{N}_h$$

tail group primal update (and transmit → head group)

$$x_i^{k+1} = \arg\min_{x_i} \{ f_i(x_i) + \langle \lambda_{i-1}^k, x_{i-1}^{k+1} - x_i \rangle + \langle \lambda_i^k, x_i - x_{i+1}^{k+1} \rangle + \frac{\rho}{2} \| x_{i-1}^{k+1} - x_i \|^2 + \frac{\rho}{2} \| x_i - x_{i+1}^{k+1} \|^2 \}, \quad i \in \mathcal{N}_t$$

both groups dual update

$$\lambda_i^{k+1} = \lambda_i^k + \rho(x_i^{k+1} - x_{i+1}^{k+1})$$

GADMM – Remarks

- Consider the scenario with central (parameter) server, where the network is a star graph. It seems that the (vanilla) GADMM does NOT communicate (per round) less than ADMM for a star graph. However, in the cross-device scenarios, where connection to central server might be slow.
- the (chain) topology is too restrictive. Any graph with a vertex of degree > 2 is not able to be fitted in the GADMM settings.
- more?

Related Work – M-ADMM

In fact, $\exists$ previous work [2][1] which proposed Mixed Gauß-Seidel and Jacobian ADMM (M-ADMM).

M-ADMM partitions a multi-block problem (m blocks) into 2 groups $(1, \ldots, m')$ and $(m'+1, \ldots, m)$. The 2 groups update in serial, while each block within one group updates in parallel.

[1] C. Lu, J. Feng, S. Yan, and Z. Lin, A unified alternating direction method of multipliers by majorization minimization, IEEE transactions on pattern analysis and machine intelligence, 2017

Related Work – M-ADMM

Primal updates of M-ADMM are

$$x_i^{k+1} = \arg\min_{x_i} \{f_i(x_i) + \widetilde{\mathcal{L}}(x_1^k, \ldots, x_i, \ldots, x_m^k, \lambda)\}, \qquad 1 \leqslant i \leqslant m'$$

$$x_j^{k+1} = \arg\min_{x_i} \{f_i(x_i) + \widetilde{\mathcal{L}}(x_1^{k+1}, \ldots, x_{m'}^{k+1}, x_{m'+1}^k, \ldots, x_j, \ldots, x_m^k, \lambda)\}, \quad m' < j \leqslant m$$

where $\widetilde{\mathcal{L}}$ is some Lagrangian function, e.g. augmented Lagrangian, linearized Lagrangian, or majorant first-order surrogate of f_i.

CQ-GGADMM – Enhanced GADMM

To further address the problems that GADMM does not solve, CQ-GGADMM [3] is proposed

- limited graph topology
- unreduced bandwidth/power per channel

or does not really solve, e.g. # communication per round

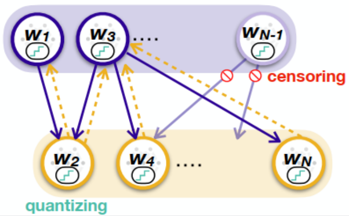
CQ-GGADMM – Enhanced GADMM

To further address the problems that GADMM does not solve, CQ-GGADMM [3] is proposed

- limited graph topology
- unreduced bandwidth/power per channel

or does not really solve, e.g. # communication per round

The connection graph topology (corr. to the "G")



which is a bipartite and connected graph.

CQ-GGADMM – Enhanced GADMM

Corr. optimization problem hence is formulated as

$$\begin{aligned} &\text{minimize} \quad \frac{1}{N}\sum_{i=1}^{N} f_i(x_i) \\ &\text{subject to} \quad x_i = x_j, \quad (i,j) \in \mathscr{E} \end{aligned}$$

CQ-GGADMM – Enhanced GADMM

Corr. optimization problem hence is formulated as

$$\begin{aligned} \text{minimize} \quad & \frac{1}{N}\sum_{i=1}^{N} f_i(x_i) \\ \text{subject to} \quad & x_i = x_j, \quad (i,j) \in \mathscr{E} \end{aligned}$$

Other techniques include

- quantization (corr. to the "Q"), reduces bandwidth/power per channel
- censoring (corr. to the "C"), reduces communication per round

Related Work – LAG & LASG

Censoring is a technique such that local parameters are updated and transmitted, only when parameters change large enough:

$$\widehat{x}_i^{k+1} = \begin{cases} Q(x_i^{k+1}) & \text{if } \|\widehat{x}_i^k - Q(x_i^{k+1})\| \geqslant \tau_0 \xi^{k+1} \\ \widehat{x}_i^k & \text{otherwise} \end{cases}$$

where $\widehat{x}$ denotes the quantized parameters and $Q(\cdot)$ refer to the quantization process[2]

[2] should be studied in details later?

Related Work – LAG & LASG

Censoring is a technique such that local parameters are updated and transmitted, only when parameters change large enough:

$$\widehat{x}_i^{k+1} = \begin{cases} Q(x_i^{k+1}) & \text{if } \|\widehat{x}_i^k - Q(x_i^{k+1})\| \geqslant \tau_0 \xi^{k+1} \\ \widehat{x}_i^k & \text{otherwise} \end{cases}$$

where $\widehat{x}$ denotes the quantized parameters and $Q(\cdot)$ refer to the quantization process[2]

Censoring, I think, originates from previous work, e.g. LAG [4] (and later LASG [5]) where "censoring" is referred to as "Lazy Aggregation"

[2] should be studied in details later?

References I

[1] A. Elgabli, J. Park, A. S. Bedi, M. Bennis, and V. Aggarwal, "GADMM: Fast and Communication Efficient Framework for Distributed Machine Learning," *Journal of Machine Learning Research*, vol. 21, no. 76, pp. 1–39, 2020.

[2] C. Lu, J. Feng, S. Yan, and Z. Lin, "A Unified Alternating Direction Method of Multipliers by Majorization Minimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 527–541, 2017.

[3] C. B. Issaid, A. Elgabli, J. Park, and M. Bennis, "Communication Efficient Distributed Learning with Censored, Quantized, and Generalized Group ADMM," *arXiv preprint arXiv:2009.06459*, 2020.

References II

[4] T. Chen, G. Giannakis, T. Sun, and W. Yin, "LAG: Lazily Aggregated Gradient for Communication-Efficient Distributed Learning," in *Advances in Neural Information Processing Systems* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, Curran Associates, Inc., 2018.

[5] T. Chen, Y. Sun, and W. Yin, "LASG: Lazily Aggregated Stochastic Gradients for Communication-Efficient Distributed Learning," 2020.