7 可扩展机器学习系统中的优化算法

为了应对大数据机器学习问题,出现了一批采用分布式架构的可扩展的机器学习系统.分布式架构大致可以分为两类:第 1 类是不含参数服务器架构的系统,其大多为整体同步并行计算模型;第 2 类是采用参数服务器架构的系统,也是目前机器学习系统的主流发展方向,主要为延迟同步并行计算模型.在原始的主从架构平台中,从节点在完成计算后需要和主节点进行交互.然而在大数据环境下,单一节点难以存储大规模参数集,同时对节点通信、参数计算等造成单点瓶颈,因此,为应对分布式环境下的大规模数据集,参数服务器架构应运而生.参数服务器架构是指模型的参数采用一个统一的分布式服务器组作为主节点来进行存储管理,参数服务器组中的每个服务器分别对应不同的从节点组,从而不存在单点故障.由于算法的实现离不开平台,因此,本节对分布式平台进行简介,包括其实现的适用于该系统的分布式优化算法,并总结分析不同平台中算法的应用场景.

第 1 类中,基于整体同步并行计算模型的机器学习系统主要包括 Hadoop Mahout[24]、Spark MLlib[25]和 GraphLab[71].其中,Mahout 和 Spark 主要采用 MapReduce 编程模型.通过 Mahout 系统最新版本介绍,其实现了一种采用 SGD 算法的逻辑回归模型,并没有单独的 SGD 算法实现,也没有其他模型来使用该算法.Spark MLlib 实现了 Mini-Batch SGD 算法、L-BFGS 算法和 PG 算法.其中,PG 算法用来求解 L1 正则化问题,Mini-Batch SGD 算法和 L-BFGS 算法都适用于 MLlib 算法库中模型的求解.但是,由于 L-BFGS 算法的运行速率快于 Mini-Batch SGD 算法,因此最新版本中的 MLlib 库通常使用 L-BFGS 算法作为求解算法.然而,Spark 等开源系统不支持参数服务器架构,在模型规模上都无法支持互联网级别机器学习模型训练的需求.同时,大量优化算法在大量数据集上的收敛速度均比 Spark 上实现的原始 SGD 算法快几十倍,因此,Spark 上的优化算法有很大的改进空间.CMU 开发的 GraphLab[71]采用 gather-apply-scatter 的图计算模型,从节点从相邻节点收集信息,并将该信息发送到主节点进行汇总,主节点将更新后的值传给从节点;最后,从节点更新相邻边信息进行新一轮计算.作为分布式图模型,实现了 SGD 算法,以及由于 GraphLab 平台的协同过滤工具箱的需要实现的 Bias-SGD 算法.虽然 GraphLab 用图来作抽象可以解决大部分机器学习问题,但仍然有很多问题无法高效求解,比如深度学习中的多层结构.

第 2 类机器学习系统都采用参数服务器架构加 SSP 更新策略,主要包括 CMU 的 Parameter Server[24]和 Petuum[23],它们均实现了一种新的分布式 SGD 算法.腾讯的 Angel 机器学习平台也是基于参数服务器架构,以实现多种业界最新技术和腾讯自主研发技术,如异步分布式 SGD、多线程参数共享模式 Hogwild!等方法.作为机器学习的延伸,采用参数服务器架构的深度学习系统逐渐涌起.Google Distbelief[26]是 Google 第 1 代分布式深度学习平台,主要实现了改进的分布式随机梯度下降法(downpour SGD)以及对深度学习有很好效果的 L-BFGS 算法.随后,Google 公司第 2 代分布式深度学习平台 Tensorflow[27]实现了若干添加学习率优化器的梯度下降法,包括 GradientDescentOptimizer、AdadeltaOptimizer、AdagradOptimizer、MomentumOptimizer、AdamOptimizer、FtrlOptimizer、RMSPropOptimizer 等.同时,Li 等人的 MxNet[25]平台也是类似的实现方式,主要实现了 GD 算法和 PG 算法.

综合来看,大多数分布式机器学习平台都实现了 SGD 算法,但是仅限于基本的 Mini-Batch SGD,用于求解机器学习问题.其中,新一代的机器学习平台都采用了分布式参数服务器架构,实现了数据并行和模型并行,这些架构本身就是一种新的算法实现,不同于本文前面介绍的算法理论.从深度学习角度来看,由于非凸函数的求解理论研究不足,目前,实验结果表明,添加自适应学习率的 SGD 算法可以满足求解要求,是主流的方法.Google Distbelief 和 Spark MLlib 都还实现了 L-BFGS 算法,同时,实验结果还表明:在一些情况下,其运行效率要优于 SGD.但是,为了避免系统过于复杂,Tensorflow 并未实现 L-BFGS 算法.虽然各个平台都可以解决机器学习问题,但是具体问题的求解需要考虑具体的应用场景,例如数据特性、数据规模、运行结果需求等.同时,可以结合具体系统的特性与其已有的优化算法来进行平台与算法的选择.例如,想要保证结果的准确性,对运行时间没有太高要求,则可优先选择 BSP 模型的平台.对于决定了运行平台,选择哪种优化算法来求解具体模型,可根据求解模型和数据的特性来选择优化算法.例如,如果模型是 SVM,则可选择实现坐标下降法;如果带有约束条件,则可选择 ADMM 算法.

表 11 总结了现有机器学习平台其优化算法实现的情况.

Table 11 Implementation of optimization algorithms on existing machine learning platforms

表 11 现有机器学习平台优化算法实现情况

通信模式		系统	内置算法
无参数服务器架构	主从架构机器学习系统	Mahout Spark	SGD Mini-BatchSGD, PG, L-BFGS
	图架构机器学习系统	GraphLab	Mini-BatchSGD, Bias-SGD
参数服务器架构	通用机器学习系统	Parameter server Petuum Angel	Mini-Batch SGD Mini-BatchSGD SGD, Hogwild!
	深度学习系统	DistBelief	Downpour SGD, L-BFGS
		Tensorflow	SGD, Adadelta_SGD, Adam_SGD, Adagrad_SGD, Momentum_SGD, Ftrl_SGD, RMSProp_SGD
		MxNet	Mini-Batch SGD, PG

8 分析与讨论

飞速增长的数据量使得模型处理需要更快的速度,机器学习算法的效率是最关键的问题之一.而机器学习算法的效率更多地依赖于优化方法的改进,因此,优化算法作为机器学习的重要组成部分,近年来在并行与分布式机器学习领域获得了广泛关注,取得了诸多研究成果,得到了快速发展.本文从模型训练优化角度出发,对梯度下降算法、二阶优化算法、邻近梯度算法、坐标下降算法和交替方向乘子算法这 5 种不同类型的优化算法的并行与分布式优化策略进行综述.通过层次化分类,将各种算法按照目标函数类型总结为如图 5 所示.

图中箭头表示论文中该算法可用于求解此类目标函数,不代表其不能求解其他类型的目标函数,只是每一种算法针对目标函数从理论上有一定的优势,实际应用中的性能还需基于数据集进行评测分析.从图中可知:大部分优化算法求解的问题域与其原始理论解释保持一致,部分算法突破了原有设定,进行了相关的改进.通过查看图 5,从本质上加深了对算法优化技巧的理解,帮助开发者求解模型时对优化算法的选择,为相关优化算法的并行与分布式实现提供参考,并且可以交叉探索将优化算法应用到新的目标函数类型上.对算法综述的同时,调研分析了现有分布式学习平台优化算法的实现程度,全面总结了优化算法从理论到实际的发展情况.通过总结发现,5 类分布式优化算法都可以适用于分布式环境下大规模问题的求解.其中,梯度下降法、二阶优化算法、交替方向乘子法可以解决凸函数与非凸函数的问题:梯度下降法的使用范围更广,在现有分布式学习平台上均有部署,求解简单且性能较好,基本上可以满足机器学习与深度学习的需求;二阶优化算法求解效率更高,虽然计算稍复杂些,但对于传统机器学习,该方法很受欢迎,部署范围较广;交替方向乘子法独特的算法架构很适合于分布式环境,但在求解无约束问题时,需要构造约束条件,将无约束问题转换为有约束问题来求解.目前,在实际应用中还处于研究阶段;邻近梯度法主要解决 L1 正则化带来的不可求导问题,只有少数平台实现了该算法;坐标下降法适合于分布式环境下求解多维数据或者函数不可求导问题,SVM 模型通常利用该算法求解,性能较好,但是由于梯度法或邻近梯度法可以对其进行替换,在实际平台中没有应用.

尽管机器学习优化算法在并行与分布式方面取得了众多的进展,但仍然存在如下几点不足.

(1) 在算法使用多样性方面.

在现有大规模数据环境下,应用最广泛、关注最多的优化算法是梯度下降法,因为其实现简单,使用场景广泛.但是由于应用的精细化,不同的应用场景需要不同的机器学习模型,对于不同的目标函数、数据规模可能需要不同的优化策略,不能一概而论.然而,其他优化算法在实际应用中并不常见,如何结合各种优化算法的特性、提高具体应用的运行效率,还有待进一步加以研究.

(2) 在算法扩展性方面.

对于分布式算法,扩展性作为重要指标在现有分布式算法中没有得到重视.随着数据规模的扩大,在分布式环境下可以配置更多的机器,如何有效地利用这些计算资源,分布式机器学习优化算法的可扩展性至关重要.

(3) 在与系统结合方面.

随着集群机器的普通化,异构环境下不同机器的速度不一,同步将会造成大量的同步等待,从而越来越多的

研究转变到异步更新策略上.对于更新策略为异步的情况,高维稀疏数据往往能取得更好的加速效果.因此,如何更好地利用现有系统、如何将现有算法移植到异步环境下、如何应用到稠密数据上还有待研究.

(4) 在非凸函数优化方面.

现有优化算法的目标函数主要针对凸函数,然而关注度逐渐升高的深度学习的目标函数主要为非凸函数.目前,非凸函数的主要求解策略是添加自适应学习率的随机梯度下降法.虽然梯度下降法能够较好地解决问题,但是对于求解大规模深度学习问题仍存在较大难度,存在收敛速度较慢等不足.利用其他优化算法求解神经网络相关的非凸函数问题仍有待进一步加以研究.

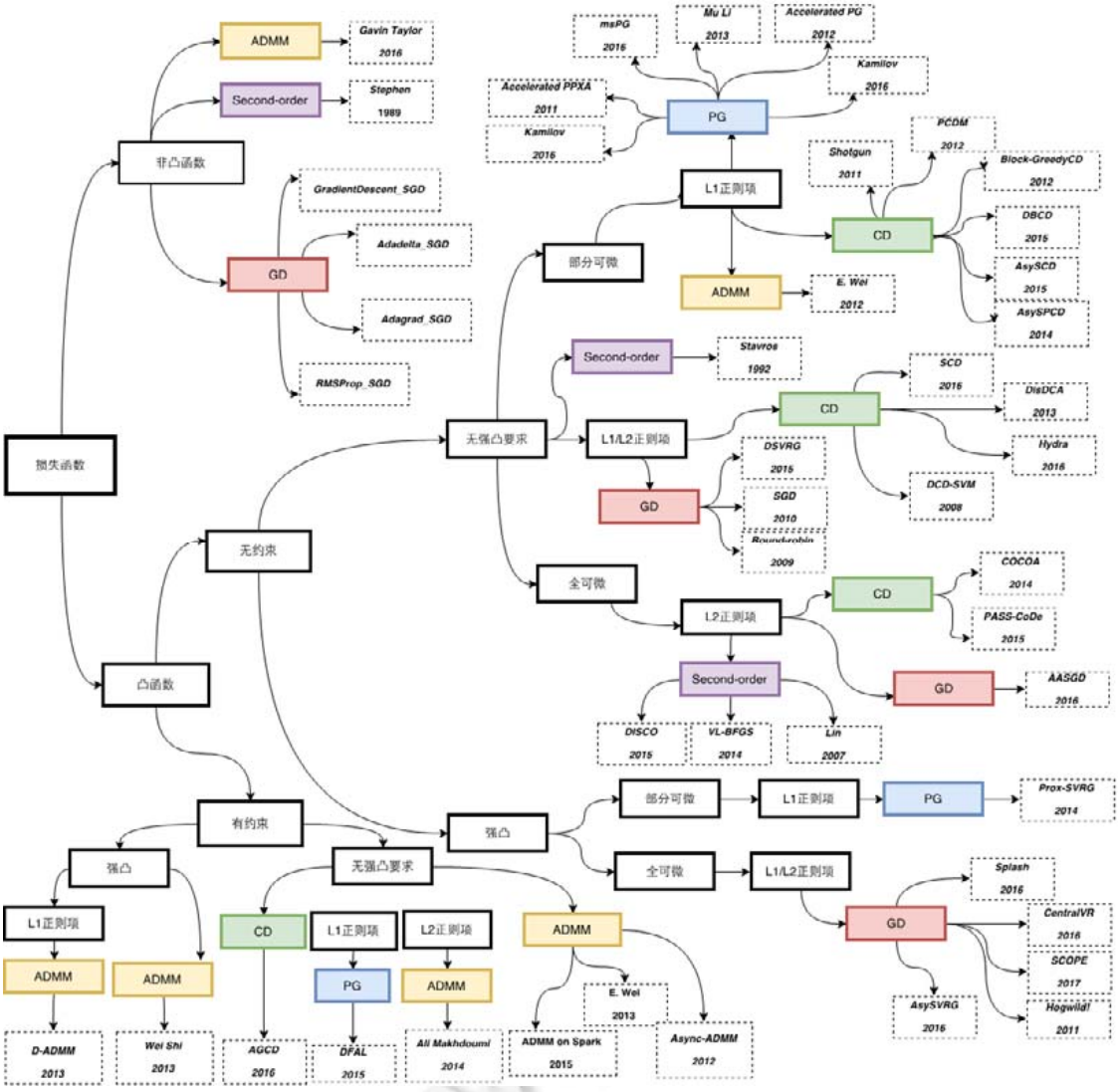

Fig.5 Hierarchical classification of optimization algorithm by objective functions

图 5 基于目标函数类型的优化算法层次化分类

由于机器学习优化方法的研究方兴未艾,对其所进行的研究工作尚不完善,在与平台相结合的并行与分布式方面仍需要做大量工作.未来的研究方向主要可以从如下几个方面加以展开.

(1) 与系统相结合的分布式优化算法.

随着数据体量的增大,与系统相结合的分布式优化算法进行模型求解将需要更大的内存以及更快的计算效率.然而现有分布式算法大多只是理论研究,基于真实应用平台的分布式算法研究仍处于探索阶段.未来可以基于真实的分布式机器学习平台运行多种优化算法,找出影响运行速度的优化算法或平台因素,从而进行相应的改进.

(2) 与深度学习相关的优化算法.

现有分布式优化算法对非凸函数的研究较少,主要是利用添加自适应学习率的随机梯度下降法,但是梯度下降法并不能解决全部的深度学习问题.因此,针对非凸优化机器学习问题,还需要大量的理论研究.同时,还需要基于分布式深度学习平台进行实例测试,从理论和实践上改进非凸函数的优化求解问题.

9 总 结

综上所述,现有并行与分布式优化方法已经取得了较好的科研和实际应用成果,不同的优化算法在并行与分布式环境下针对不同的目标函数有不同的改进,对于不同的应用也可以得到较高的处理效率,在实际应用平台上部署效果较好.但仍有创新的空间,如果在现有优化算法优化策略的基础上,将优化策略与现有并行与分布式机器学习平台相结合,推出更加适用于不同应用场景的优化算法,将会形成更多创新的算法,有利于大规模机器学习问题的优化求解,满足实际应用需求,使其应用前景更加广阔.

References:

[1] Léon B, Frank EC, Jorge N. Optimization methods for large-scale machine learning. arXiv: 1606.04838v1, 2016.

[2] Neal P, Stephen B. Proximal algorithms. Foundations and Trends in Optimization, 2014,1(3):127−239. [doi: 10.1561/2400000003]

[3] Stephen JW. Coordinate descent algorithms. arXiv: 1502.04759v1, 2015.

[4] Stephen B, Neal P, Eric C, Borja P, Jonathan E. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning, 2011,3(1):1−122.

[5] Eric X, Qirong H, Xie PT, Wei D. Strategies and principles of distributed machine learning on big data. Engineering, 2016,2(2): 179−195. [doi: 10.1016/J.ENG.2016.02.008]

[6] Frédéric L, Frédéric G, David B. Bulk synchronous parallel ML: Modular implementation and performance prediction. In: Proc. of the Int'l Conf. on Computational Science. 2005. 1046−1054. [doi: 10.1007/11428848_132]

[7] Nesterov Y. Introductory Lectures on Convex Optimization: A Basic Course. Springer-Verlag, 2004. xviii−236.

[8] Meng XR, Joseph B, Burak Y, Evan S, Shivaram V, Davies L, Jeremy F, DB T, Manish M, Sean O, Doris X, Reynold X, Michael JF, Reza Z, Matei Z, Ameet T. MLlib: Machine learning in apache spark. The Journal of Machine Learning Research, 2016,17(1): 1235−1241.

[9] Martin AZ, Markus W, Alexander S, Li LH. Parallelized stochastic gradient descent. In: Advances in Neural Information Processing Systems. 2010. 2595−2603. http://papers.nips.cc/paper/4006-parallelized-stochastic-gradient-descent.pdf

[10] Leen TK, Orr GB. Optimal stochastic search and adaptive momentum. In: Advances in Neural Information Processing Systems. 1994. 477−484. http://papers.nips.cc/paper/772-optimal-stochastic-search-and-adaptive-momentum.pdf

[11] Rie J, Zhang T. Accelerating stochastic gradient descent using predictive variance reduction. In: Advances in Neural Information Processing Systems. 2013. 315−323. http://papers.nips.cc/paper/4937-accelerating-stochastic-gradient-descent-using-predictive-variance-reduction.pdf

[12] Aaron D, Francis B, Simon LJ. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In: Advances in Neural Information Processing Systems. 2014. 1646−1654. http://papers.nips.cc/paper/5258-saga-a-fast-incremental-gradient-method-with-support-for-non-strongly-convex-composite-objectives.pdf

[13] Nicol S, Yu J, Simon G. A stochastic quasi-newton method for online convex optimization. The Journal of Machine Learning Research, 2007,2:436−443.

[14] Goldfarb D. A family of variable metric updates derived by variational means. Mathematics of Computation, 1970,24(109):23−26. [doi: 10.1090/S0025-5718-1970-0258249-6]

[15] Liu DC, Nocedal J. On the limited memory BFGS for large scale optimization. Mathematical Programming, 1989,45(1):503−528. [doi: 10.1007/BF01589116]

[16] Combettes PL, Wajs VR. Signal recovery by proximal forward-backward splitting. Multiscale Modeling & Simulation, 2005,4(4): 1168−1200. [doi: 10.1137/050626090]

[17] Ram SS, Nedich A, Veeravalli VV. Incremental stochastic subgradient algorithms for convex optimization. SIAM Journal on Optimization, 2009,20(2):691−717. [doi: 10.1137/080726380]

[18] Luo ZQ, Tseng P. On the convergence of the coordinate descent method for convex differentiable minimization. Journal of Optimization Theory and Applications, 1992,72(1):7−35. [doi: 10.1007/BF00939948]

[19] Nesterov Y. Efficiency of coordinate descent methods on huge-scale optimization problems. SIAM Journal on Optimization, 2012, 22(2):341−362. [doi: 10.1137/100802001]

[20] Dhillon IS, Ravikumar P, Tewari A. Nearest neighbor based greedy coordinate descent. In: Advances in Neural Information Processing Systems. 2011. 2160−2168. http://papers.nips.cc/paper/4425-nearest-neighbor-based-greedy-coordinate-descent.pdf

[21] Canutescu A, Dunbrack R. Cyclic coordinate descent: A robotics algorithm for protein loop closure. Protein Science, 2003,12(5): 963−972. [doi: 10.1110/ps.0242703]

[22] Zhao SY, Xiang R, Shi YH, Gao P, Li WJ. SCOPE: Scalable composite optimization for learning on spark. In: Proc. of the Association for the Advancement of Artificial Intelligence. 2017. 2928−2934.

[23] Qirong H, James C, Jin K, Seunghak L, Phillip G, Garth G, Gregory G, Eric X. More effective distributed ML via a stale synchronous parallel parameter server. In: Advances in Neural Information Processing Systems. 2013. 1223−1231. http://papers.nips.cc/paper/4894-more-effective-distributed-ml-via-a-stale-synchronous-parallel-parameter-server.pdf

[24] Mu L, David A, Jun P, Alexander S, Amr A, Vanja J, James L, Eugene S, Bor-Yiing S. Scaling distributed machine learning with the parameter server. OSDI, 2014,1(10.4):3. [doi: 10.1145/2640087.2644155]

[25] Chen TQ, Li M, Li YT, Lin M, Wang NY, Wang MJ, Xiao TJ, Xu B, Zhang CY, Zhang Z. MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems. arXiv: 1512.01274, 2015.

[26] Jeffrey D, Greg C, Rajat M, Chen K, Matthieu D, Quoc VL, Mark M, Marc R, Andrew S, Paul T, Yang K, Andrew NG. Large scale distributed deep networks. In: Proc. of the Advances in Neural Information Processing Systems. 2012. 1223−1231. http://papers.nips.cc/paper/4687-large-scale-distributed-deep-networks.pdf

[27] Martin A, Ashish A, Paul B, Eugene B, Chen ZF, Craig C, Greg C, Andy D, Jeffrey D, Matthieu D, Sanjay G, Ian G, Andrew H, Geoffrey I, Michael I, Jia YQ, Rafal J, Lukasz K, Manjunath K, Josh L, Dan M, Rajat M, Sherry M, Derek M, Chris O, Mike S, Jonathon S, Benoit S, Ilya S, Kunal T, Paul T, Vincent V, Vijay V, Fernanda V, Oriol V, Pete W, Martin W, Martin W, Yu Y, Zheng XQ. TensorFlow: A system for large-scale machine learning. In: Proc. of the USENIX Symp. on Operating Systems Design and Implementation, Vol.16. 2016. 265−283. https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf

[28] John L, Alexander S, Martin Z. Slow learners are aast. In: Advances in Neural Information Processing Systems, Vol.22. 2009. 2331−2339. https://papers.nips.cc/paper/3888-slow-learners-are-fast.pdf

[29] Feng N, Recht B, Re C, Wright SJ. Hogwild!: A lock-free approach to parallelizing stochastic gradient descent. In: Advances in Neural Information Processing Systems, Vol.24. 2011. 693−701. http://papers.nips.cc/paper/4390-hogwild-a-lock-free-approach-to-parallelizing-stochastic-gradient-descent.pdf

[30] Zhao SY. Fast asynchronous parallel stochastic gradient descent: A lock-free approach with convergence guarantee. In: Proc. of the Association for the Advancement of Artificial Intelligence. 2016. 2379−2385. http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/download/12442/11887

[31] Meng Q, Chen W, Yu JC, Wang TF, Ma ZM, Liu TY. Asychronous accelerated stochastic gradient descent. In: Proc. of the Int'l Joint Conf. on Artificial Intelligence. 2016. 1853−1859. https://www.ijcai.org/Proceedings/16/Papers/265.pdf

[32] Jason L, Lin QH, Ma TY, Yang TB. Distributed stochastic variance reduced gradient. arXiv: 1507.07595v2, 2015.

[33] Soham D, Tom G. Efficient distributed SGD with variance reduction. In: Proc. of the 16th IEEE Int'l Conf. on Data Mining (ICDM). IEEE, 2016. 111−120. [doi: 10.1109/ICDM.2016.0022]

[34] Zhang YC, Michael IJ. Splash: User-Friendly programming interface for parallelizing stochastic algorithms. arXiv: 1506.07552, 2015.

[35] Steffen R, Dennis F, Eugene JS, Su BY. Robust large-scale machine learning in the cloud. In: Proc. of the 22nd ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2016. 1125−1134. [doi: 10.1145/2939672.2939790]

[36] Stavros Z, Mustafa CP. Parallel block-partitioning of truncated newton for nonlinear network. SIAM Journal on Scientific and Statistical Computing, 1992,13(5):1173−1193. [doi: 10.1137/0913068]

[37] Lin CJ, Ruby CW, Keerthi SS. Trust region newton method for large-scale logistic regression. In: Proc. of the 24th Int'l Conf. on Machine Learning. ACM Press, 2007. 561−568. [doi: 10.1145/1390681.1390703]

[38] Chen WZ, Wang ZH, Zhou JR. Large-Scale L-BFGS using MapReduce. In: Advances in Neural Information Processing Systems, Vol.27. 2014. 1332−1340. http://papers.nips.cc/paper/5333-large-scale-l-bfgs-using-mapreduce.pdf

[39] Zhang YC, Lin X. DiSCO: Distributed optimization for self-concordant empirical loss. The Journal of Machine Learning Research, 2015,37:362−370.

[40] Nelly P, Caroline C, Jean-Christophe P. Parallel proximal algorithm for image restoration using hybrid regularization. IEEE Trans. on Image Processing, 2011,20(9):2450−2462. [doi: 10.1109/TIP.2011.2128335]

[41] Kamilov US. Parallel proximal methods for total variation minimization. In: Proc. of the 2016 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016. 4697−4701. [doi: 10.1109/ICASSP.2016.7472568]

[42] Kamilov US. A parallel proximal algorithm for anisotropic total variation minimization. IEEE Trans. on Image Processing, 2017, 26(2):539−548. [doi: 10.1109/TIP.2016.2629449]

[43] Chen AI, Asuman O. A fast distributed proximal-gradient method. In: Proc. of the 50th Annual Allerton Conf. on Communication, Control, and Computing (Allerton). IEEE, 2012. 601−608. [doi: 10.1109/Allerton.2012.6483273]

[44] Lin X, Zhang T. A proximal stochastic gradient method with progressive variance reduction. SIAM Journal on Optimization, 2014, 24(4):2057−2075. [doi: 10.1137/140961791]

[45] Mu L, David A, Alexander S. Distributed delayed proximal gradient methods. In: Proc. of the NIPS Workshop on Optimization for Machine Learning. 2013. 3. http://www.cs.cmu.edu/afs/cs/user/muli/www/file/ddp.pdf

[46] Zhou Y, Yu YL, Dai W, Liang YB, Eric X. On convergence of model parallel proximal gradient algorithm for stale synchronous parallel system. The Journal of Machine Learning Research, 2016,51:713−722.

[47] Aybat NS, Wang Z, Iyengar G. An asynchronous distributed proximal gradient method for composite convex optimization. arXiv:1409.8547v2, 2015.

[48] Mota JF, Xavier JM, Aguiar PM, Püschel M. D-ADMM: A communication-efficient distributed algorithm for separable optimization. IEEE Trans. on Signal Processing, 2013,61(10):2718−2723. [doi: 10.1109/TSP.2013.2254478]

[49] Peter R, Martin T. Parallel coordinate descent methods for big data optimization. Mathematical Programming, 2016,156(1-2): 433−484. [doi: 10.1007/s10107-015-0901-6]

[50] Liu J, Stephen JW, Christopher R, Victor B, Srikrishna S. An asynchronous parallel stochastic coordinate descent algorithm. The Journal of Machine Learning Research, 2015,16(1):285−322.

[51] Scherrer C, Tewari A, Halappanavar M, Haglin D. Feature clustering for accelerating parallel coordinate descent. In: Advances in Neural Information Processing Systems. 2012. 28−36. http://papers.nips.cc/paper/4674-feature-clustering-for-accelerating-parallel-coordinate-descent.pdf

[52] Scherrer C, Tewari A, Halappanavar M, Haglin D. Scaling up coordinate descent algorithms for large L1 regularization problems. In: Proc. of the 29th Int'l Conf. on Machine Learning. Omnipress, 2012. 355−362.

[53] Yang Y, Lian XR, Liu J, Yu HF, Dhillon IS, Demmel J, Hsieh CJ. Asynchronous parallel greedy coordinate descent. In: Advances in Neural Information Processing Systems. 2016. 4682−4690. http://papers.nips.cc/paper/6070-asynchronous-parallel-greedy-coordinate-descent.pdf

[54] Liu J, Stephen JW. Asynchronous stochastic coordinate descent: Parallelism and convergence properties. SIAM Journal on Optimization, 2015,25(1):351−376. [doi: 10.1137/140961134]

[55] Hsieh CJ, Yu HF, Dhillon IS. PASSCoDe: Parallel asynchronous stochastic dual co-ordinate descent. The Journal of Machine Learning Research, 2015,37:2370−2379.

[56] Hsieh CJ, Chang KW, Lin CJ, Keerthi SS, Sundarajan S. A dual coordinate descent method for large-scale linear SVM. In: Proc. of the 25th Int'l Conf. on Machine Learning. ACM Press, 2008. 408−415. [doi: 10.1145/1390156.1390208]

[57] Yang TB. Trading computation for communication: Distributed stochastic dual coordinate ascent. In: Advances in Neural Information Processing Systems. 2013. 629−637. http://papers.nips.cc/paper/5114-trading-computation-for-communication-distributed-stochastic-dual-coordinate-ascent.pdf

[58] Yang TB, Zhu SH, Jin R, Lin YQ. Analysis of distributed stochastic dual coordinate ascent. arXiv: 1312.1031, 2013.

[59] Martin J, Virginia S, Martin T, Joathan T, Sanjay K, Thomas H, Michael IJ. Communication-Efficient distributed dual coordinate ascent. In: Advances in Neural Information Processing Systems. 2014. 3068−3076. http://papers.nips.cc/paper/5599-communication-efficient-distributed-dual-coordinate-ascent.pdf

[60] Mahajan D, Keerthi SS, Sundararajan S. A distributed block coordinate descent method for training L1 regularized linear classifiers. arXiv: 1405.4544, 2014.

[61] Dhar S, Yi C, Ramakrishnan N, Shah M. ADMM based scalable machine learning on spark. In: Proc. of the 2015 IEEE Int'l Conf. on Big Data. IEEE. 2015. 1174−1182. [doi: 10.1109/BigData.2015.7363871]

[62] Stephen GN, Ariela S. Block truncated-newton methods for parallel optimization. Mathematical Programming, 1989,45(1): 529−546. [doi: 10.1007/BF01589117]

[63] Gavin T, Ryan B, Xu Z, Bharat S, Ankit P, Tom G. Training neural networks without gradients: A scalable ADMM approach. The Journal of Machine Learning Research, 2016,48:2722−2731.

[64] Zhang R, Kwok JT. Asynchronous distributed ADMM for consensus optimization. In: Proc. of the 31st Int'l Conf. on Machine Learning. 2014. 1701−1709. http://proceedings.mlr.press/v32/zhange14.pdf

[65] Shi W, Ling Q, Yuan K, Wu G, Yin W. On the linear convergence of the ADMM in decentralized consensus optimization. arXiv: 1307.5561v4, 2014. [doi: 10.1109/TSP.2014.2304432]

[66] Bradley JK, Kyrola A, Bickson D, Guestrin C. Parallel coordinate descent for L1-regularized loss minimization. arXiv: 1105.5379, 2011.

[67] Wei E, Asuman O. On the $O(1/k)$ convergence of asynchronous distributed alternating direction method of multipliers. In: Proc. of the 2013 IEEE Global Conf. on Signal and Information Processing. IEEE, 2013. 551−554. [doi: 10.1109/GlobalSIP.2013.6736937]

[68] Ali M, Asuman O. Broadcast-Based distributed alternating direction method of multipliers. In: Proc. of the 52nd Annual Allerton Conf. on Communication, Control, and Computing (Allerton). IEEE, 2014. 270−277. [doi: 10.1109/ALLERTON.2014.7028466]

[69] Wei E, Asuman O. Distributed alternating direction method of multipliers. In: Proc. of the 51st IEEE Annual Conf. on Decision and Control (CDC). IEEE, 2012. 5445−5450. [doi: 10.1109/CDC.2012.6425904]

[70] Peter R, Martin T. Distributed coordinate descent method for learning with big data. The Journal of Machine Learning Research, 2016,17(1):2657−2681.

[71] Low Y, Gonzalez JE, Kyrola A, Bickson D, Guestrin CE, Hellerstein J. Graphlab: A new parallel framework formachine learning. Computer Science, 2014. https://arxiv.org/ftp/arxiv/papers/1408/1408.2041.pdf

亢良伊(1993－),女,山西临汾人,本科生,主要研究领域为分布式机器学习,深度学习及其优化.

刘杰(1982－),男,博士,副研究员,CCF 专业会员,主要研究领域为机器学习,分布式系统,软件工程.

王建飞(1992－),男,本科生,主要研究领域为机器学习,分布式系统.

叶丹(1971－),女,博士,高级工程师,博士生导师,CCF 高级会员,主要研究领域为网络分布式系统,软件工程.