Adaptive Inexact Sequential Quadratic Programming via Iterative Randomized Sketching

Ilgee Hong ILGEE@UCHICAGO.EDU
Department of Statistics, The University of Chicago.

Sen Na SENNA@BERKELEY.EDU
ICSI and Department of Statistics, University of California, Berkeley.

Mladen Kolar MLADEN.KOLAR@CHICAGOBOOTH.EDU
Booth School of Business, The University of Chicago.

Abstract

We consider solving nonlinear optimization problems with equality constraints. We propose a randomized algorithm based on sequential quadratic programming (SQP) with a differentiable exact augmented Lagrangian as the merit function. In each SQP iteration, we solve the Newton system inexactly via iterative randomized sketching. The accuracy of the inexact solution and the penalty parameter of the augmented Lagrangian are adaptively controlled in the algorithm to ensure that the inexact random search direction is a descent direction of the augmented Lagrangian. This allows us to establish global convergence almost surely. Moreover, we show that a unit stepsize is admissible for the inexact search direction provided the iterate lies in a neighborhood of the solution. Based on this result, we show that the proposed algorithm exploits local linear convergence. We apply the algorithm on benchmark nonlinear problems in CUTEst test set and on constrained logistic regression with datasets from LIBSVM to demonstrate its superior performance.

1. Introduction

We consider the nonlinear equality-constrained optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) \qquad \text{s.t. } c(\boldsymbol{x}) = \mathbf{0}, \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the objective function and $c : \mathbb{R}^n \to \mathbb{R}^m$ are equality constraints. There exist numerous methods for solving Problem (1.1), including projected first- and second-order methods, penalty methods, augmented Lagrangian methods, and sequential quadratic programming (SQP). In this paper, we focus on solving (1.1) via SQP, which is one of the leading second-order methods for constrained optimization problems [9, 10, 14]. The algorithms in this class typically enjoy global convergence guarantees, and require a few iterations to find a local solution. However, the computational cost of SQP algorithms is dominated by solving one (or more) Newton system in each iteration, which can be prohibitive for large-scale problems.

To reduce the per-iteration computational cost, [4] proposed an inexact SQP algorithm where, in each iteration, the Newton system is approximately solved using a deterministic iterative solver and the stepsizes are chosen based on a penalized merit function. With suitable conditions on the quality of the inexact solution, the authors showed that the inexact search direction is still a descent direction of the merit function and the algorithm enjoys global convergence. Despite the

© I. Hong, S. Na & M. Kolar.

solid theoretical underpinnings, the algorithm of [4] suffers from few drawbacks. First, for each SQP iteration, the algorithm relies on a few fixed tuning parameters (κ_1, κ_2, ϵ, β) for bounding the residuals of the iterative solver. These parameters may substantially affect the performance of the algorithm and have to be chosen carefully. In particular, a tighter residual bound will lead to more inner loop iterations to compute a more precise step. However, the cost of more inner loop iterations must be balanced against a possible decrease in the outer loop iterations for finding the local solution. Second, the algorithm uses a nonsmooth merit function $\phi_\pi(\boldsymbol{x}) = f(\boldsymbol{x}) + \pi\|c(\boldsymbol{x})\|$ when performing the line search, which is known to cause the Maratos effect—a unit stepsize may not be accepted near the solution. Such an effect leads to a slow local convergence [5]. Third, the local behavior of that algorithm has not been rigorously analyzed.

In this paper, we propose a randomized SQP algorithm to solve Problem (1.1) in which the Newton system in the inner loop is solved using the iterative randomized sketching (IRS) [8]. Thus, the proposed method could be seen as a randomized extension of [4]. Furthermore, instead of using fixed bound to control the accuracy of the inexact search direction throughout all SQP iterations, the proposed method adaptively controls the accuracy of the inexact solution to balance between the number of inner and outer loop iterations whilst the method achieves fast local convergence. We use a differentiable *exact* augmented Lagrangian as the merit function of the form

$$\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}, \boldsymbol{\lambda}) = \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) + \frac{\eta_1}{2}\|c(\boldsymbol{x})\|^2 + \frac{\eta_2}{2}\|\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda})\|^2, \tag{1.2}$$

where $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T c(\boldsymbol{x})$ is the Lagrangian function of Problem (1.1) with $\boldsymbol{\lambda} \in \mathbb{R}^m$ being the Lagrangian multipliers, and $\boldsymbol{\eta} = (\eta_1, \eta_2)$ is the penalty parameter. The benefit of using an *exact* penalty function is that a stationary point $(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star)$ of (1.2) is also a stationary point of Problem (1.1) and vice versa, provided that η_1 is sufficiently large and η_2 is sufficiently small [2, Proposition 4.15]. Further, the smoothness of the merit function in (1.2) effectively overcomes the Maratos effect [3]. We emphasize three novelties of the proposed algorithm. First, we use the iterative randomized sketching [8] to compute an inexact solution of the Newton system. Projecting a large Newton system into a smaller one and obtaining an approximate solution leads to large computational savings [8, 16, 17]. Second, the algorithm adaptively selects a parameter that controls a bound on the residuals when accepting the search direction. As a result, the inexact solution of the Newton system is a descent direction of the merit function, and is accurate enough to guarantee the global and local linear convergence of the algorithm. Empirically, our adaptive algorithm results in smaller KKT residuals (the sum of the feasibility error and the optimality error) and fewer gradient evaluations. Third, despite the randomness in the inexact search direction brought by the randomized solver, we establish the almost sure global convergence. Furthermore, we show that the algorithm locally selects a unit stepsize even with the adaptive step acceptance condition, which leads to a local linear convergence rate. Such a local result complements the existing literature on inexact SQP algorithms.

2. Method

We propose an adaptive inexact SQP algorithm that uses iterative randomized sketching to solve the Newton system in the inner loop. We use $\|\cdot\|$ to denote the ℓ_2 norm for vectors and the operator norm for matrices. At the k-th outer iteration, we let $f_k = f(\boldsymbol{x}_k)$, etc., to simplify the notation.

When a constraint qualification holds, the first-order necessary conditions for $\boldsymbol{x}^\star$ to be a solution to Problem (1.1) are that there exist multipliers $\boldsymbol{\lambda}^\star$ such that

$$\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star) \end{pmatrix} = \begin{pmatrix} \nabla f(\boldsymbol{x}^\star) + G^T(\boldsymbol{x}^\star)\boldsymbol{\lambda}^\star \\ c(\boldsymbol{x}^\star) \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tag{2.1}$$

where $G(\boldsymbol{x}) = \nabla^T c(\boldsymbol{x}) = (\nabla c_1(\boldsymbol{x}), \ldots, \nabla c_m(\boldsymbol{x}))^T \in \mathbb{R}^{m \times n}$ is the constraint Jacobian. In each outer iteration k, the SQP algorithm finds the search direction $(\Delta \boldsymbol{x}_k, \Delta \boldsymbol{\lambda}_k)$ by solving the following Newton system

$$\begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0} \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} = - \begin{pmatrix} \nabla f_k + G_k^T \boldsymbol{\lambda}_k \\ c_k \end{pmatrix}, \tag{2.2}$$

where $B(\boldsymbol{x}, \boldsymbol{\lambda})$ is the Hessian of the Lagrangian $\nabla^2_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = H(\boldsymbol{x}, \boldsymbol{\lambda})$ or its symmetric perturbation. Let $\Gamma_k = \begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}$ and we rewrite the Newton system (2.2) by

$$\Gamma_k \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} = - \begin{pmatrix} \nabla f_k + G_k^T \boldsymbol{\lambda}_k \\ c_k \end{pmatrix}. \tag{2.3}$$

Instead of finding the exact Newton direction $(\Delta \boldsymbol{x}_k, \Delta \boldsymbol{\lambda}_k)$, we apply the iterative randomized sketching to obtain an inexact solution $(\tilde{\Delta} \boldsymbol{x}_k, \tilde{\Delta} \boldsymbol{\lambda}_k)$ to (2.3). In particular, we let $S \in \mathbb{R}^{(n+m)\times d}$ be a random sketch matrix which has some probability distribution $\mathcal{P}$ and for each outer iteration k and inner iteration j, we specify each random matrix by $S_{k,j} \sim S$. For j-th inexact solution $(\tilde{\Delta} \boldsymbol{x}_{k,j}, \tilde{\Delta} \boldsymbol{\lambda}_{k,j})$, we define the residual vectors of the Newton system by

$$\boldsymbol{r}_{k,j} = \begin{pmatrix} \boldsymbol{r}^p_{k,j} \\ \boldsymbol{r}^d_{k,j} \end{pmatrix} = \Gamma_k \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_{k,j} \\ \tilde{\Delta} \boldsymbol{\lambda}_{k,j} \end{pmatrix} + \begin{pmatrix} \nabla f_k + G_k^T \boldsymbol{\lambda}_k \\ c_k \end{pmatrix}. \tag{2.4}$$

Then the inner loop iteration updates the solution as

$$\begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_{k,j+1} \\ \tilde{\Delta} \boldsymbol{\lambda}_{k,j+1} \end{pmatrix} = \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_{k,j} \\ \tilde{\Delta} \boldsymbol{\lambda}_{k,j} \end{pmatrix} - W_{k,j} \begin{pmatrix} \boldsymbol{r}^p_{k,j} \\ \boldsymbol{r}^d_{k,j} \end{pmatrix}, \tag{2.5}$$

where $W_{k,j} = \Gamma_k^T S_{k,j} \left(S_{k,j}^T \Gamma_k \Gamma_k^T S_{k,j} \right)^{-1} S_{k,j}^T \in \mathbb{R}^{(n+m)\times(n+m)}$. Now we define

$$\delta_k^{\text{trial}} = \left(\frac{1}{2} - \beta \right) \frac{\eta_{2,k}}{2\Psi_k^2 (3\Upsilon_k + 4\eta_{2,k}\Upsilon_k^2 + \eta_{1,k}\Upsilon_k^2)}, \tag{2.6}$$

where $\Upsilon_k = \|B_k\| \vee \|G_k\| \vee \|H_k\|$ and Ψ_k is defined in Lemma 6. At each outer iteration k, we force the adaptive parameter δ_k, which controls the accuracy of the inexact solution of (2.3), to be smaller than δ_k^{trial}. This procedure ensures the algorithm selects a unit stepsize locally, so that it enjoys the local linear convergence near a stationary point of Problem (1.1). For the simplicity of notation, we drop the inner iteration j from $(\tilde{\Delta} \boldsymbol{x}_{k,j}, \tilde{\Delta} \boldsymbol{\lambda}_{k,j})$ and $\boldsymbol{r}_{k,j}$ when we generally refer to the inexact search direction and residual vector. The following condition describes when a search direction will be accepted.

Step Acceptance Condition. Given $\eta_{1,k}, \eta_{2,k} > 0$ and $0 < \delta_k \leq \delta_k^{\text{trial}}$, a step $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ that is computed via (2.5) is acceptable if

$$\|\boldsymbol{r}_k\| \leq \delta_k \frac{\|\nabla \mathcal{L}_k\|}{\|\Gamma_k\| \, \Psi_k} \tag{2.7}$$

and

$$\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \leq -\frac{\eta_{2,k}}{2} \|\nabla \mathcal{L}_k\|^2 . \tag{2.8}$$

In each outer iteration k, we first update the inexact search direction by (2.5) until the residual at the inner iteration j satisfies (2.7) with a given δ_k. Then, we check if the inexact search direction is a descent direction of (1.2); that is, whether (2.8) is satisfied for given $(\eta_{1,k}, \eta_{2,k})$. If it does not satisfy (2.8), we increase $\eta_{1,k}$, and decrease $\eta_{2,k}$ and δ_k as

$$\eta_{1,k} \leftarrow \eta_{1,k}\nu^2, \quad \eta_{2,k} \leftarrow \eta_{2,k}/\nu, \quad \delta_k \leftarrow (\delta_k/\nu^4 \wedge \delta_k^{\text{trial}}) \tag{2.9}$$

where $\nu > 1$ is a given constant. We repeat the above two steps until we find an inexact search direction which satisfies (2.7) and (2.8) with appropriate $(\eta_{1,k}, \eta_{2,k}, \delta_k)$. We design this scheme using double while loops in Algorithm 1. The stepsize α_k is selected to satisfy the Armijo condition

$$\mathcal{L}_{\boldsymbol{\eta}}^{k+1} \leq \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \beta \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}, \tag{2.10}$$

and the iterate is updated as

$$\begin{pmatrix} \boldsymbol{x}_{k+1} \\ \boldsymbol{\lambda}_{k+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_k \\ \boldsymbol{\lambda}_k \end{pmatrix} + \alpha_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}. \tag{2.11}$$

Our Algorithm 1 is presented in Appendix C.

3. Convergence Analysis

We now study well-posedness of Algorithm 1, and establish global and local linear convergence guarantees. We emphasize that the randomness plays a key role in the analysis since the inexact search direction is calculated by the iterative randomized solver. Compared with an algorithm that uses a deterministic iterative solver, our inexact search direction is stochastic. As a result, all the components of the algorithm that are affected by the search direction are also random; for example, (2.7), (2.8), and (2.10). Our analysis relies on the following assumption.

Assumption 1 *All the iterates $\{\boldsymbol{x}_k\}_{k\geq 0}$ belong to an open convex set $\mathcal{X}$. The objective function f is twice continuously differentiable and bounded over $\mathcal{X}$. Its gradient ∇f and Hessian $\nabla^2 f$ are Lipschitz continuous and bounded over $\mathcal{X}$. The constraint function c is twice continuously differentiable, Lipschitz continuous, and bounded over $\mathcal{X}$. Its Jacobian G and Hessian of each coordinate are Lipschitz continuous and bounded below over $\mathcal{X}$.*

Assumption 2 *The Jacobian matrices $\{G_k\}_{k\geq 0}$ have full row rank. There exist constants $\xi_B, \Upsilon_B > 0$, such that, for any outer iteration $k \geq 0$, $\boldsymbol{z}^T B_k \boldsymbol{z} \geq \xi_B \|\boldsymbol{z}\|^2$ for any $\boldsymbol{z} \in \{\boldsymbol{z} : G_k \boldsymbol{z} = 0\}$ and $\|B_k\| \leq \Upsilon_B$.*

Assumption 3 *The random sketch matrix S satisfies $\mathcal{P}(S^T \boldsymbol{z} \neq \mathbf{0}) > 0$ for any $\boldsymbol{z} \in \mathbb{R}^{n+m} \backslash \{\mathbf{0}\}$. For any outer and inner iteration $k, j \geq 0$, $S_{k,j} \overset{i.i.d.}{\sim} S$.*

Assumption 1 does not make any assumptions about the set Λ that contains the dual iterates $\{\boldsymbol{\lambda}_k\}_{k\geq 0}$. The boundedness of Λ can be proven based on the algorithm itself; see Lemma 13 in Appendix A. Assumption 2 implies that Γ_k in (2.3) is invertible. Therefore, for any outer iteration k, the Newton system (2.3) has a unique solution. This is a standard assumption in the SQP literature [3]. Assumption 3 is used specifically to establish the well-posedness of Algorithm 1. In Lemma 7, we first define the subsequence of the inner iteration $\{j_l\}_{l\geq 0}$ where the reduction of the error step occurs, and show that the event $\mathcal{A}_k = \cap_{l=1}^{L}\{j_l < \infty\}$ happens with probability 1. Thus, conditioned on the event $\mathcal{A}_k$, the error linearly decays in those iterations (see Lemma 8). Lemma 9 shows that for each outer iteration k, conditioned on the event $\mathcal{A}_k$, almost surely, there exists finite inner iteration such that the first component in *Step Acceptance Condition* (2.7) is satisfied. We denote this event as $\mathcal{B}_k$. In Lemma 10, we show that conditioned on the event $\mathcal{A}_k \cap \mathcal{B}_k$, the second component in *Step Acceptance Condition* (2.8) is satisfied. Thus, Lemma 9 and 10 imply that the double while loop in Algorithm 1 terminates in finite time. Furthermore, Lemma 11 shows that all adaptive parameters (η_1, η_2, δ) will be fixed at some values after a number of outer iterations. The formal statements of Lemma 7–11 are presented in Appendix A. Finally, we establish the global convergence of Algorithm 1 in Theorem 1.

Theorem 1 (Global convergence) *Suppose Assumption 1, 2, 3 hold for the iterates $\{(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)\}_{k\geq 0}$ generated by Algorithm 1. Then $\|\nabla \mathcal{L}_k\| \to 0$ as $k \to \infty$ almost surely.*

Next, we establish local linear convergence guarantees of Algorithm 1. We first present two additional assumptions that are necessary to the local behaviour analysis.

Assumption 4 *The third derivative of the objective function $\nabla^3 f$ exists and continuous over $\mathcal{X}$. The third derivative of the constraints $\nabla^3 c_i$ exists and continuous over $\mathcal{X}$ for all $i \in \{1, \ldots, m\}$.*

Assumption 5 *For any outer iteration $k \geq 0$, $\|H_k - B_k\| = o(1)$.*

Assumption 4 strengthens the condition of the objective function f and constraints c in Assumption 1 to thrice continuously differentiability. For Assumption 4, when using the augmented Lagrangian as the merit function (see (1.2)), it is common to assume the existence of third derivatives of f and c_i, since the Hessian of the augmented Lagrangian $\nabla^2_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}$ requires $\nabla^3 f$ and $\nabla^3 c_i$ to exist. The existence of third derivatives is only necessary for analysis, and they are never computed in practice. Assumption 5 is standard in the SQP literature and is needed to show local superlinear or quadratic convergence [3]. Now we establish local linear convergence of Algorithm 1 in Theorem 2.

Theorem 2 (Local linear convergence) *Let $(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star)$ be a stationary point of (1.1). Suppose Assumption 1, 2, 3, 4, 5 hold for the iterates $\{(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)\}_{k\geq 0}$ generated by Algorithm 1, and $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) \to (\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star)$. Then for all sufficiently large outer iteration k, almost surely, $\alpha_k = 1$, and*

$$\left\| \begin{pmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \leq \delta^\star \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|,$$

where $\delta^\star$ be the stabilized value of $\delta \in (0, 1)$.

Proof of Theorem 1 and 2 are given in Appendix B.

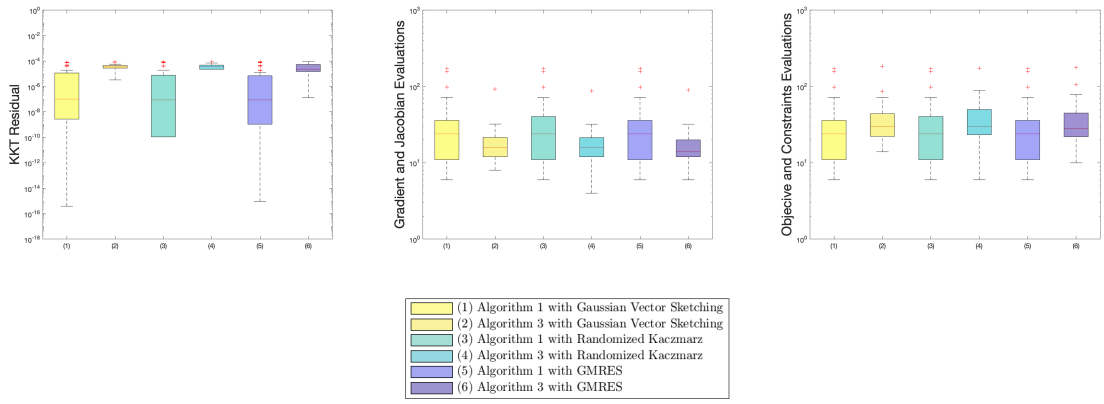

Figure 1: KKT residual, number of gradient and Jacobian evaluations, and number of objective and constraints evaluations boxplots for Algorithm 1 and Algorithm 3 on CUTEst problems.

4. Experiments

We implement three inexact SQP algorithms to solve benchmark nonlinear problems in CUTEst test set [7] and solve constrained logistic regression with datasets from LIBSVM [6]. The considered three algorithms are Algorithm 1 (the proposed algorithm), Algorithm 2: [4] with the ℓ_1 penalized merit function, and Algorithm 3: adaptive version of Algorithm 2. We use two randomized iterative solvers and one deterministic iterative solver for (2.3): Gaussian vector sketch [8, Section 3.2], Randomized Kaczmarz [8, Section 3.3], and GMRES [15]. We evaluate each algorithm with the following three criteria: (1) the KKT residual ($\|\nabla \mathcal{L}_k\|$), (2) the number of gradient and Jacobian evaluations, and (3) the number of objective and constraints evaluations. We first present the comparison between Algorithms 1 and 3 on CUTEst set in Figure 1.

From Figure 1, we observe that Algorithm 1 outperforms Algorithm 3 in terms of the KKT residual and number of objective and constraints evaluations. This result is expected as Algorithm 1 uses tighter bounds on the residuals of the iterative solver to guarantee fast local convergence. This results in steeper decrease in the merit function at each iteration and fewer number of outer iterations. However, as we mentioned earlier, smaller number of outer iterations yields possible increase in the number of inner iterations required to satisfy (2.7). We can reduce this cost by applying IRS, which substantially saves the computational complexities by projecting (2.3) into a smaller space for each inner iteration.

On the one hand, we see Algorithm 3 shows slightly lower number of gradient and Jacobian evaluations than Algorithm 1. This is because for each iteration, Algorithm 1 finds a stepsize α_k to satisfy the Armijo condition (2.10), and $\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k + \alpha_k \tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + \alpha_k \tilde{\Delta}\boldsymbol{\lambda}_k)$ in (2.10) requires the gradient and Jacobian to be evaluated at each new trial point. For Algorithm 3, however, the gradient and Jacobian are not involved in the evaluations of the ℓ_1 penalized merit function at new trial points.

Further comparisons between Algorithms 2 and 3, and experiments on LIBSVM datasets are in Appendix D due to the limitation of space. The code is available at: `https://github.com/IlgeeHong/Randomized-SQP`.

References

[1] Albert S Berahas, Frank E Curtis, Michael J O'Neill, and Daniel P Robinson. A stochastic sequential quadratic optimization algorithm for nonlinear equality constrained optimization with rank-deficient jacobians. *arXiv preprint arXiv:2106.13015*, 2021. URL https://arxiv.org/abs/2106.13015.

[2] Dimitri P Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014. URL https://www.mit.edu/~dimitrib/lagr_mult.html.

[3] Paul T. Boggs and Jon W. Tolle. Sequential quadratic programming. *Acta Numerica*, 4:1–51, jan 1995. doi: 10.1017/s0962492900002518. URL https://doi.org/10.1017/S0962492900002518.

[4] Richard H. Byrd, Frank E. Curtis, and Jorge Nocedal. An inexact SQP method for equality constrained optimization. *SIAM Journal on Optimization*, 19(1):351–369, jan 2008. doi: 10.1137/060674004. URL https://doi.org/10.1137/060674004.

[5] R. M. Chamberlain, M. J. D. Powell, C. Lemarechal, and H. C. Pedersen. The watchdog technique for forcing convergence in algorithms for constrained optimization. In *Mathematical Programming Studies*, pages 1–17. Springer Berlin Heidelberg, 1982. doi: 10.1007/bfb0120945. URL https://doi.org/10.1007/BFb0120945.

[6] Chih-Chung Chang and Chih-Jen Lin. LIBSVM. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, apr 2011. doi: 10.1145/1961189.1961199. URL https://doi.org/10.1145/1961189.1961199.

[7] Nicholas I. M. Gould, Dominique Orban, and Philippe L. Toint. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications*, 60(3):545–557, aug 2014. doi: 10.1007/s10589-014-9687-3. URL https://doi.org/10.1007/s10589-014-9687-3.

[8] Robert M. Gower and Peter Richtárik. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690, jan 2015. doi: 10.1137/15m1025487. URL https://doi.org/10.1137/15M1025487.

[9] S. P. Han. A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3):297–309, jul 1977. doi: 10.1007/bf00932858. URL https://doi.org/10.1007/BF00932858.

[10] S. P. Han and O. L. Mangasarian. Exact penalty functions in nonlinear programming. *Mathematical Programming*, 17(1):251–269, dec 1979. doi: 10.1007/bf01588250. URL https://doi.org/10.1007/BF01588250.

[11] Sen Na, Mihai Anitescu, and Mladen Kolar. A fast temporal decomposition procedure for long-horizon nonlinear dynamic programming. *arXiv preprint arXiv:2107.11560*, 2021. URL https://arxiv.org/abs/2107.11560.

[12] Sen Na, Mihai Anitescu, and Mladen Kolar. An adaptive stochastic sequential quadratic programming with differentiable exact augmented lagrangians. *Mathematical Programming*, pages 1–71, jun 2022. doi: 10.1007/s10107-022-01846-z. URL https://doi.org/10.1007/s10107-022-01846-z.

[13] Vivak Patel, Mohammad Jahangoshahi, and Daniel A. Maldonado. An implicit representation and iterative solution of randomly sketched linear systems. *SIAM Journal on Matrix Analysis and Applications*, 42(2):800–831, jan 2021. doi: 10.1137/19m1259481. URL https://doi.org/10.1137/19M1259481.

[14] M. J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In *Lecture Notes in Mathematics*, pages 144–157. Springer Berlin Heidelberg, 1978. doi: 10.1007/bfb0067703. URL https://doi.org/10.1007/BFb0067703.

[15] Youcef Saad and Martin H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, jul 1986. doi: 10.1137/0907058. URL https://doi.org/10.1137/0907058.

[16] Thomas Strohmer and Roman Vershynin. A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, apr 2008. doi: 10.1007/s00041-008-9030-4. URL https://doi.org/10.1007/s00041-008-9030-4.

[17] David P. Woodruff. Computational advertising: Techniques for targeting relevant ads. *Foundations and Trends® in Theoretical Computer Science*, 10(1-2):1–157, 2014. doi: 10.1561/0400000060. URL http://dx.doi.org/10.1561/0400000060.

Appendix A.

Lemma 3 (Upper bound on Hessian of Lagrangian) *Under Assumption 1, 2, 3, for any outer iteration k, there exists a uniform constant $\Upsilon_H > 0$, independent of k, such that for any outer iteration k, $\|H_k\| \leq \Upsilon_H$.*

Lemma 4 (Upper bound on Newton matrix) *Under Assumption 1, 2, for any outer iteration k, there exists a uniform constant $\Upsilon_N > 0$, independent of k, such that for any outer iteration k, $\|\Gamma_k\| \leq \Upsilon_N$.*

Lemma 5 (Boundedness on Jacobian of constraints) *Under Assumption 2, there exist constants $\kappa_G, \xi_G > 0$ such that for any outer iteration k, $\xi_G I \preceq G_k G_k^T \preceq \kappa_G I$.*

Lemma 6 (Upper bound on Newton matrix inverse) *Under Assumption 2, for any outer iteration k, we let $\Psi_k = \dfrac{7(\|B_k\|^2 \vee 1)}{\xi_B(\sigma_{1,k} \wedge 1)}$ where $\sigma_{1,k}$ is the smallest eigenvalue of $G_k G_k^T$. Then for any outer iteration k, $\left\|\Gamma_k^{-1}\right\| \leq \Psi_k$.*

For later usage, we further define $\Psi = \sup_{k\geq 0}\{\Psi_k\}$ and $\Upsilon = \sup_{k\geq 0}\{\Upsilon_k\}$.

Lemma 7 *Let $Q_{k,j}$ be a random matrix with orthonormal columns that form a basis of $row(S_{k,j}^T \Gamma_k)$. Let $\{j_l\}_{l\geq 0}$ be a subsequence of the inner iteration where $j_0 = 0$ and j_l be the l-th iteration such that*

$$col\left(Q_{k,j_{l-1}+1}\right) + \cdots + col\left(Q_{k,j_l}\right) = \mathbb{R}^{n+m}.$$

Let L be any given positive integer. Under Assumption 2, 3, for any outer iteration k, conditioned on the event that the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$, the event

$$\mathcal{A}_k = \cap_{l=1}^{L}\{j_l < \infty\} \tag{A.1}$$

happens with probability 1.

Lemma 8 (Subsequence of error linearly decays) *Under Assumption 2, 3, for any outer iteration k and for any positive integer L, conditioned on the event that the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and $\mathcal{A}_k$ in (A.1), there exists a sequence of random variables $\{\gamma_{k,l}\}_{l=1}^{L}$ where $\gamma_{k,l} \overset{i.i.d.}{\sim} \gamma_k \in [0,1)$ such that, for any $l \leq L$,*

$$\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j_l} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_l} - \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\| \leq \gamma_{k,l}\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j_{l-1}} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_{l-1}} - \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|.$$

Lemma 9 (Error of inexact solution) *For any $\delta_k \in (0,1)$, let J_k be the inner iteration such that*

$$\|\boldsymbol{r}_{k,J_k}\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\|\,\Psi_k}.$$

Under Assumption 1, 2, 3, for any outer iteration k, conditioned on the event that the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and $\mathcal{A}_k$ in (A.1), the event

$$\mathcal{B}_k = \{J_k < \infty\} \tag{A.2}$$

happens with probability 1. Moreover, conditioned on the event $\mathcal{A}_k \cap \mathcal{B}_k$, if we let $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k) = (\tilde{\Delta}\boldsymbol{x}_{k,J_k}, \tilde{\Delta}\boldsymbol{\lambda}_{k,J_k})$, then

$$\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \left\| \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\|. \tag{A.3}$$

Lemma 10 (Descent direction of inexact step) *Let $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ be the inexact solution that satisfies (2.7). Under Assumption 1, 2, 3, for any outer iteration k, conditioned on the event that the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and $\mathcal{A}_k \cap \mathcal{B}_k$ in (A.1) and (A.2), if*

$$\eta_{1,k} \geq \frac{17\kappa_G}{\eta_{2,k}\xi_G^2}, \quad \eta_{2,k} \leq \frac{\xi_B}{12\Upsilon^2}, \quad \delta_k \leq \frac{\eta_{2,k}\xi_G}{16\eta_{1,k}\Upsilon^2},$$

then we have

$$\begin{pmatrix} \nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \leq -\frac{\eta_{2,k}}{2} \left\| \nabla\mathcal{L}_k \right\|^2.$$

Lemma 11 (Stability of adaptive parameters) *Under Assumption 1, 2, 3, after sufficiently large outer iteration k, all adaptive parameters (η_1, η_2, δ) are stabilized almost surely.*

Lemma 12 (Armijo condition) *Under Assumption 1, 2, 3, for any outer iteration k, conditioned on the event that the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and $\mathcal{A}_k \cap \mathcal{B}_k$ in (A.1) and (A.2), the Armijo condition (2.10) is satisfied. Moreover, there exists a uniform constant $\alpha_{\min} > 0$, independent of k, such that for any k, $0 < \alpha_{\min} \leq \alpha_k$.*

Lemma 13 (Boundedness of dual variable) *Under Assumption 1, 2, 3, almost surely, $\{\boldsymbol{\lambda}_k\}_{k\geq 0}$ produced by Algorithm 1 is bounded.*

Appendix B. Proof of Lemma and Theorem

B.1. Proof of Lemma 3

Proof Under Assumption 1, 2, 3, Lemma 13 shows $\{\boldsymbol{\lambda}_k\}_{k\geq 0}$ is bounded. Using Assumption 1, we have for any outer iteration k, $\nabla^2 f_k$, $\nabla^2 c_{i,k}$ are all bounded. Then we get

$$\|H_k\| = \left\|\nabla^2_{\boldsymbol{xx}}\mathcal{L}_k\right\| = \left\|\nabla^2 f_k + \sum_{i=1}^{m}\lambda_{i,k}\nabla^2 c_{i,k}\right\| \leq \left\|\nabla^2 f_k\right\| + \max_i\left\{\,|\lambda_{i,k}|\,\right\}\sum_{i=1}^{m}\left\|\nabla^2 c_{i,k}\right\| \leq \Upsilon_H.$$

This ends proof of Lemma 3. ■

B.2. Proof of Lemma 4

Proof Assumption 1 implies that for any outer iteration k, there exists a uniform constant $\Upsilon_G > 0$, independent of k, such that for any outer iteration k, $\|G_k\| \leq \Upsilon_G$. Using this fact together with Assumption 2, we get

$$\|\Gamma_k\| \leq \|B_k\| + 2\,\|G_k\| \leq \Upsilon_B + 2\Upsilon_G \leq \Upsilon_N.$$

This ends proof of Lemma 4. ■

B.3. Proof of Lemma 5

Proof Assumption 2 implies that for any outer iteration k, $G_kG_k^T$ is positive definite. For any outer iteration k, let $\sigma_{m,k} \geq \cdots \geq \sigma_{1,k} > 0$ be the eigenvalues of $G_kG_k^T$. Then we can show $\sigma_{1,k}I \preceq G_kG_k^T \preceq \sigma_{m,k}I$. If we let $\xi_G = \inf_{k\geq 0}\{\sigma_{1,k}\}$, and $\kappa_G = \sup_{k\geq 0}\{\sigma_{m,k}\}$ then for any outer iteration k, $\xi_G I \preceq G_kG_k^T \preceq \kappa_G I$. This ends proof of Lemma 5. ■

B.4. Proof of Lemma 6

Proof For any outer iteration k, let Z_k be a matrix which has orthonormal columns spanning the null space of G_k. Using Assumption 2, we have $Z_k^TB_kZ_k \succeq \xi_B I$ and $G_k^T(G_kG_k^T)^{-1}G_k + Z_kZ_k^T = I$. Appendix C.1. in [11] and [12] implies that

$$\Gamma_k^{-1} = \begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0}\end{pmatrix}^{-1} = \begin{pmatrix} \mathcal{K}_1 & \mathcal{K}_2^T \\ \mathcal{K}_2 & \mathcal{K}_3\end{pmatrix}$$

where

$$\mathcal{K}_1 = Z_k(Z_k^TB_kZ_k)^{-1}Z_k^T, \quad \mathcal{K}_2 = (G_kG_k^T)^{-1}G_k(I - B_kZ_k(Z_k^TB_kZ_k)^{-1}Z_k^T)$$
$$\mathcal{K}_3 = (G_kG_k^T)^{-1}G_k(B_kZ_k(Z_k^TB_kZ_k)^{-1}Z_k^TB_k - B_k)G_k^T(G_kG_k^T)^{-1}.$$

Taking ℓ_2 norm on both sides yields,

$$\|\mathcal{K}_1\| \leq \frac{1}{\xi_B}, \quad \|\mathcal{K}_2\| \leq \|(G_kG_k^T)^{-1}G_k\|\left(1 + \frac{\|B_k\|}{\xi_B}\right) \leq \frac{1}{\sqrt{\sigma_{1,k}}}\left(1 + \frac{\|B_k\|}{\xi_B}\right),$$
$$\|\mathcal{K}_3\| \leq \|(G_kG_k^T)^{-1}G_k\|^2\left(\|B_k\| + \frac{\|B_k\|^2}{\xi_B}\right) \leq \frac{1}{\sigma_{1,k}}\left(\|B_k\| + \frac{\|B_k\|^2}{\xi_B}\right).$$

Let $\xi_B \leq 1$ and assume $\sigma_{1,k} \leq 1 \leq \|B_k\|$. Then we get

$$\begin{aligned}\left\|\Gamma_k^{-1}\right\| &\leq \|\mathcal{K}_1\| + 2\|\mathcal{K}_2\| + \|\mathcal{K}_3\| \\ &\leq \frac{1}{\xi_B} + \frac{2}{\sqrt{\sigma_{1,k}}}\left(1 + \frac{\|B_k\|}{\xi_B}\right) + \frac{1}{\sigma_{1,k}}\left(\|B_k\| + \frac{\|B_k\|^2}{\xi_B}\right) \\ &\leq \frac{5\,\|B_k\|}{\sqrt{\sigma_{1,k}}\xi_B} + \frac{2\,\|B_k\|^2}{\sigma_{1,k}\xi_B} \leq \frac{7\,\|B_k\|^2}{\sigma_{1,k}\xi_B}.\end{aligned}$$

Since we assume $\sigma_{1,k} \leq 1 \leq \|B_k\|$, it follows that

$$\left\|\Gamma_k^{-1}\right\| \leq \frac{7(\|B_k\|^2 \vee 1)}{\xi_B(\sigma_{1,k} \wedge 1)} = \Psi_k.$$

This ends proof of Lemma 6. ■

B.5. Proof of Lemma 7

Proof Let $k \geq 0$ and we suppose the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$. Using (2.3) and (2.4), we rewrite the updating rule of IRS (2.5) as

$$\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j+1} \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j+1}\end{pmatrix} = \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j} \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j}\end{pmatrix} - W_{k,j}\left(\Gamma_k\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j} \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j}\end{pmatrix} + \nabla\mathcal{L}_k\right) = \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j} \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j}\end{pmatrix} - W_{k,j}\Gamma_k\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j} - \Delta\boldsymbol{\lambda}_k\end{pmatrix}.$$

If we let $\boldsymbol{e}_{k,j} = \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_{k,j} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j} - \Delta\boldsymbol{\lambda}_k\end{pmatrix}$, then the above display can be rewritten as

$$\boldsymbol{e}_{k,j+1} = \boldsymbol{e}_{k,j} - W_{k,j}\Gamma_k\boldsymbol{e}_{k,j}. \tag{B.1}$$

Using the fact that $W_{k,j}\Gamma_k = \Gamma_k^T S_{k,j+1}(S_{k,j+1}^T\Gamma_k\Gamma_k^T S_{k,j+1})^{-1}S_{k,j+1}^T\Gamma_k$ forms an orthogonal projection onto $row(S_{k,j+1}^T\Gamma_k)$, (B.1) can be simplified as

$$\boldsymbol{e}_{k,j+1} = \boldsymbol{e}_{k,j} - Q_{k,j+1}Q_{k,j+1}^T\boldsymbol{e}_{k,j}. \tag{B.2}$$

Let $j_0 = 0$ and j_l be the l-th iteration such that

$$col\left(Q_{k,j_{l-1}+1}\right) + \cdots + col\left(Q_{k,j_l}\right) = row(\Gamma_k) = \mathbb{R}^{n+m},$$

otherwise let j_l be infinite. Since Γ_k is invertible, Assumption 3 implies $\mathcal{P}(S^T\Gamma_k\boldsymbol{z} \neq 0) > 0$ for any $\boldsymbol{z} \in \mathbb{R}^{n+m}\backslash\{0\}$. Given the relationship between $row(S^T\Gamma_k)$ and Q_k, we further get $\mathcal{P}(Q_k^T\boldsymbol{z} \neq 0) > 0$ for any $\boldsymbol{z} \in \mathbb{R}^{n+m}\backslash\{0\}$. We denote the lower bound of this probability as $\pi_k \in (0, 1]$. Since $Q_{k,j} \overset{i.i.d.}{\sim} Q_k$, conditioned on the event $\{j_{l-1} < \infty\}$, the probability that $\sum_{i=0}^{t+1} col\left(Q_{k,j_{l-1}+i}\right)$ grows in dimension relative to $\sum_{i=0}^{t} col\left(Q_{k,j_{l-1}+i}\right)$, when $dim\left(\sum_{i=0}^{t} col\left(Q_{k,j_{l-1}+i}\right)\right) < n+m$ is at least π_k. As a result, conditioned on the event $\{j_{l-1} < \infty\}$, the probability that the event $\left\{dim\left(\sum_{i=0}^{t+1} col\left(Q_{k,j_{l-1}+i}\right)\right) > dim\left(\sum_{i=0}^{t} col\left(Q_{k,j_{l-1}+i}\right)\right)\right\}$ happens $n+m$ times in N iterations with $N \geq n+m$ is dominated by a negative binomial distribution. Thus,

$$\text{for } N \geq n+m, \mathcal{P}(j_l = N | j_{l-1} < \infty) \leq \binom{N-1}{n+m-1}(1-\pi)^{N-n-m}\pi^{n+m}.$$

Taking $N \to \infty$, we get for any $l \in \mathbb{N}$,

$$\mathcal{P}(j_l = \infty | j_{l-1} < \infty) = 0.$$

Therefore, for any $l \in \mathbb{N}$, $\mathcal{P}(j_l < \infty | j_{l-1} < \infty) = 1$. Let L be given positive integer. Then for any $l \leq L$,

$$\begin{aligned}\mathcal{P}\left(\cap_{l=1}^{L}\{j_l < \infty\}\right) &= \mathcal{P}(j_1 < \infty) \times \mathcal{P}(j_2 < \infty | j_1 < \infty) \times \cdots \times \mathcal{P}(j_L < \infty | j_{L-1} < \infty, \ldots, j_1 < \infty) \\ &= \mathcal{P}(j_1 < \infty) \times \mathcal{P}(j_2 < \infty | j_1 < \infty) \times \cdots \times \mathcal{P}(j_L < \infty | j_{L-1} < \infty) \\ &= 1.\end{aligned}$$

This ends proof of Lemma 7. ■

B.6. Proof of Lemma 8

Proof Let $k \geq 0$ and we suppose the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and the event $\mathcal{A}_k$ happens. Let L be given positive integer. We denote $\boldsymbol{q}_{k,j,h}$ be the h-th column of $Q_{k,j}$. Using (B.2), we have for any $l \leq L$,

$$\boldsymbol{e}_{k,j_l} = \left(\Pi_{j=j_{l-1}+1}^{j_l}\left(\Pi_{h=1}^{p}\left(I - \boldsymbol{q}_{k,j,h}\boldsymbol{q}_{k,j,h}^T\right)\right)\right)\boldsymbol{e}_{k,j_{l-1}}.$$

Taking ℓ_2 norm on both sides yields

$$\|\boldsymbol{e}_{k,j_l}\| \leq \left\|\Pi_{j=j_{l-1}+1}^{j_l}\left(\Pi_{h=1}^{p}\left(I - \boldsymbol{q}_{k,j,h}\boldsymbol{q}_{k,j,h}^T\right)\right)\right\| \left\|\boldsymbol{e}_{k,j_{l-1}}\right\|.$$

Let $\mathcal{F}_{k,l}$ denote all matrices $F_{k,l}$ where the columns of $F_{k,l}$ are the vectors $\{f_{k,l,1}, \ldots, f_{k,l,n+m}\} \subset \{\boldsymbol{q}_{k,j_{l-1}+1,1}, \ldots, \boldsymbol{q}_{k,j_l,d}\}$ that are a maximal linearly independent subset. Theorem 4 in [13] implies that

$$\left\|\Pi_{j=j_{l-1}+1}^{j_l}\left(\Pi_{h=1}^{p}\left(I - \boldsymbol{q}_{k,j,h}\boldsymbol{q}_{k,j,h}^T\right)\right)\right\| \leq \sqrt{1 - min_{F_{k,l} \in \mathcal{F}_{k,l}} det(F_{k,l}^T F_{k,l})}.$$

For any $l \leq L$, define

$$\gamma_{k,l} = \sqrt{1 - min_{F_{k,l} \in \mathcal{F}_{k,l}} det(F_{k,l}^T F_{k,l})}.$$

Then we have for any $l \leq L$,

$$\|\boldsymbol{e}_{k,j_l}\| \leq \gamma_{k,l} \left\|\boldsymbol{e}_{k,j_{l-1}}\right\|.$$

Using the fact that $F_{k,l}^T F_{k,l}$ is positive definite and Hadamard's inequality, we have $\{\gamma_{k,l}\}_{l \leq L} \subset [0, 1)$. Let $\mathcal{Q}_{k,l} = \{Q_{k,j_{l-1}+1}, \ldots, Q_{k,j_l}\}$. Using Assumption 3, we get $\mathcal{Q}_{k,1}, \ldots, \mathcal{Q}_{k,L} \overset{i.i.d}{\sim} \mathcal{Q}_k$, hence, $\gamma_{k,1}, \ldots, \gamma_{k,L} \overset{i.i.d}{\sim} \gamma_k$. This ends proof of Lemma 8. ■

B.7. Proof of Lemma 9

Proof Let $k \geq 0$ and we suppose the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and the event $\mathcal{A}_k$ happens. Using Lemma 8, we have $\mathcal{P}(\gamma_k = 1) = 0$, hence, there exists $\tau_k \in (0, 1)$ such that $\mathcal{P}(\gamma_k \leq \tau_k) > 0$. We denote the lower bound of $\mathcal{P}(\gamma_k \leq \tau_k)$ by $\pi_k \in (0, 1]$. Let $\bar{N}$ be the smallest positive integer such that $\bar{N} \geq log(\delta_k / \|\Gamma_k\|^2 \Psi_k^2)/log(\tau_k) + 1$. Then we have $\tau_k^{\bar{N}} \leq \dfrac{\delta_k}{\|\Gamma_k\|^2 \Psi_k^2}$. Now we consider the procedure where for each iteration l, we generate $\gamma_{k,l}$ from a distribution of γ_k independently. Let L_k be the iteration such that

$$I\{\gamma_{k,1} \leq \tau_k\} + \cdots + I\{\gamma_{k,L_k} \leq \tau_k\} = \bar{N},$$

otherwise let L_k be infinite. Since for any $l \in \mathbb{N}$, $\mathcal{P}(\gamma_{k,l} \leq \tau_k) \geq \pi_k$ and $\gamma_{k,l} \overset{i.i.d.}{\sim} \gamma_k$, the probability that the event $\{\gamma_{k,l} \leq \tau_k\}$ happens $\bar{N}$ times in N iterations with $N \geq \bar{N}$ is dominated by a negative binomial distribuiton. Thus,

$$\text{for } N \geq \bar{N}, \mathcal{P}(L_k = N) \leq \binom{N-1}{\bar{N}-1} (1-\pi_k)^{N-\bar{N}} \pi_k^{\bar{N}}.$$

Taking $N \to \infty$, we get

$$\mathcal{P}(L_k = \infty) = 0.$$

Therefore, L_k is finite with probability 1. Now letting $L = L_k$ and applying Lemma 8, we have

$$\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,j_{L_k}} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_{L_k}} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \left(\Pi_{l=1}^{L_k} \gamma_{k,l}\right) \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,0} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,0} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| = \left(\Pi_{l=1}^{L_k} \gamma_{k,l}\right) \left\| \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\|.$$

Using this expression together with (2.3), (2.4), and Lemma 6 which says $\left\|\Gamma_k^{-1}\right\| \leq \Psi_k$, we get

$$\begin{aligned} \left\|\boldsymbol{r}_{k,j_{L_k}}\right\| &= \left\| \Gamma_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,j_{L_k}} \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_{L_k}} \end{pmatrix} + \nabla\mathcal{L}_k \right\| = \left\| \Gamma_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,j_{L_k}} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_{L_k}} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \|\Gamma_k\| \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,j_{L_k}} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,j_{L_k}} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \\ &\leq \Pi_{l=1}^{L_k}(\gamma_{k,l}) \|\Gamma_k\| \left\| \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \Pi_{l=1}^{L_k}(\gamma_{k,l}) \|\Gamma_k\| \left\|\Gamma_k^{-1}\right\| \|\nabla\mathcal{L}_k\| \\ &\leq \Pi_{l=1}^{L_k}(\gamma_{k,l}) \|\Gamma_k\| \Psi_k \|\nabla\mathcal{L}_k\|. \end{aligned} \tag{B.3}$$

Using (B.3) we have

$$\begin{aligned} \left\{ I\{\gamma_{k,1} \leq \tau_k\} + \cdots + I\{\gamma_{k,L_k} \leq \tau_k\} = \bar{N} \right\} &\Rightarrow \left\{ \Pi_{l=1}^{L_k}(\gamma_{k,l}) \leq \tau_k^{\bar{N}} \right\} \\ &\Rightarrow \left\{ \Pi_{l=1}^{L_k}(\gamma_{k,l}) \leq \frac{\delta_k}{\|\Gamma_k\|^2 \Psi_k^2} \right\} \\ &\Rightarrow \left\{ \left\|\boldsymbol{r}_{k,j_{L_k}}\right\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\| \Psi_k} \right\} \end{aligned}$$

Finally, if we let $J_k = j_{L_k}$, then we obtain

$$\mathcal{P}\left(\text{there exists finite } J_k \text{ such that } \|\boldsymbol{r}_{k,J_k}\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\| \Psi_k} \,\middle|\, (\boldsymbol{x}_k, \boldsymbol{\lambda}_k), \mathcal{A}_k \right) = 1.$$

Now conditioned on the event $\mathcal{A}_k \cap \mathcal{B}_k$, we get

$$\begin{aligned} \|\boldsymbol{r}_{k,J_k}\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\| \Psi_k} &\Rightarrow \Psi_k \left\| \Gamma_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,J_k} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,J_k} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\|} \\ &\Rightarrow \left\|\Gamma_k^{-1}\right\| \left\| \Gamma_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,J_k} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,J_k} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\|} \\ &\Rightarrow \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,J_k} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,J_k} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \frac{\|\nabla\mathcal{L}_k\|}{\|\Gamma_k\|} \\ &\Rightarrow \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_{k,J_k} - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_{k,J_k} - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \frac{\|\Gamma_k\| \left\| \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\|}{\|\Gamma_k\|} \\ &\Rightarrow \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \delta_k \left\| \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} \right\|. \end{aligned}$$

This ends proof of Lemma 9. ■

B.8. Proof of Lemma 10

Proof Let $k \geq 0$ and we suppose the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and the event $\mathcal{A}_k \cap \mathcal{B}_k$ happens. We start from dividing $\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k \end{pmatrix}$ into two terms as follows:

$$\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k \end{pmatrix} = \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} + \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k - \Delta \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k - \Delta \boldsymbol{\lambda}_k \end{pmatrix}. \tag{B.4}$$

First, we develop the first term and obtain

$$\begin{aligned}
\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} &= \begin{pmatrix} (I + \eta_{2,k} H_k) \nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{1,k} G_k^T c_k \\ c_k + \eta_{2,k} G_k \nabla_{\boldsymbol{x}} \mathcal{L}_k \end{pmatrix}^T \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \\
&= \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix}^T \begin{pmatrix} I + \eta_{2,k} H_k & \eta_{1,k} G_k^T \\ \eta_{2,k} G_k & I \end{pmatrix} \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_k \\ c_k \end{pmatrix} \\
&= -\begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix}^T \begin{pmatrix} I + \eta_{2,k} H_k & \eta_{1,k} G_k^T \\ \eta_{2,k} G_k & I \end{pmatrix} \begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0} \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \\
&= -\begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix}^T \begin{pmatrix} (I + \eta_{2,k} H_k) B_k + \eta_{1,k} G_k^T G_k & (I + \eta_{2,k} H_k) G_k^T \\ G_k (I + \eta_{2,k} B_k) & \eta_{2,k} G_k G_k^T \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \\
&= -\Delta \boldsymbol{x}_k{}^T \left((I + \eta_{2,k} H_k) B_k + \frac{\eta_{1,k}}{2} G_k^T G_k \right) \Delta \boldsymbol{x}_k - \frac{\eta_{1,k}}{2} \Delta \boldsymbol{x}_k{}^T G_k^T G_k \Delta \boldsymbol{x}_k \\
&\quad - \eta_{2,k} \Delta \boldsymbol{\lambda}_k{}^T G_k G_k^T \Delta \boldsymbol{\lambda}_k - \Delta \boldsymbol{\lambda}_k{}^T G_k \left(2I + \eta_{2,k} (B_k + H_k) \right) \Delta \boldsymbol{x}_k \\
&= -\Delta \boldsymbol{x}_k{}^T \left((I + \eta_{2,k} H_k) B_k + \frac{\eta_{1,k}}{2} G_k^T G_k \right) \Delta \boldsymbol{x}_k - \frac{\eta_{1,k}}{2} \left\| G_k \Delta \boldsymbol{x}_k \right\|^2 - \eta_{2,k} \left\| G_k^T \Delta \boldsymbol{\lambda}_k \right\|^2 \\
&\quad - \Delta \boldsymbol{\lambda}_k{}^T G_k \left(2I + \eta_{2,k} (B_k + H_k) \right) \Delta \boldsymbol{x}_k.
\end{aligned}$$

Using (2.2) we have $G_k \Delta \boldsymbol{x} = -c_k$ and $G_k^T \Delta \boldsymbol{\lambda}_k = -\left(B_k \Delta \boldsymbol{x}_k + \nabla_{\boldsymbol{x}} \mathcal{L}_k \right)$. Using this expression together with the above display, we obtain

$$\begin{aligned}
&\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \\
&= -\Delta \boldsymbol{x}_k{}^T \left((I + \eta_{2,k} H_k) B_k + \frac{\eta_{1,k}}{2} G_k^T G_k \right) \Delta \boldsymbol{x}_k - \frac{\eta_{1,k}}{2} \left\| c_k \right\|^2 - \eta_{2,k} \left\| B_k \Delta \boldsymbol{x}_k + \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 \\
&\quad - \Delta \boldsymbol{\lambda}_k{}^T G_k \left(2I + \eta_{2,k} (B_k + H_k) \right) \Delta \boldsymbol{x}_k \\
&= -\Delta \boldsymbol{x}_k{}^T \left((I + \eta_{2,k} H_k) B_k + \frac{\eta_{1,k}}{2} G_k^T G_k \right) \Delta \boldsymbol{x}_k - \frac{\eta_{1,k}}{2} \left\| c_k \right\|^2 - \frac{\eta_{2,k}}{2} \left\| \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 + \frac{\eta_{2,k}}{2} \left\| \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 \\
&\quad - \eta_{2,k} \left\| B_k \Delta \boldsymbol{x}_k + \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 - \Delta \boldsymbol{\lambda}_k{}^T G_k \left(2I + \eta_{2,k} (B_k + H_k) \right) \Delta \boldsymbol{x}_k.
\end{aligned}$$

Taking the forth and fifth terms from the above display and using the expression $\nabla_{\boldsymbol{x}} \mathcal{L}_k = -\left(B_k \Delta \boldsymbol{x}_k + G_k^T \Delta \boldsymbol{\lambda}_k \right)$, we obtain

$$\begin{aligned}
&\frac{\eta_{2,k}}{2} \left\| \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 - \eta_{2,k} \left\| B_k \Delta \boldsymbol{x}_k + \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 \\
&= -\eta_{2,k} \left\| B_k \Delta \boldsymbol{x}_k \right\|^2 - 2\eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k \nabla_{\boldsymbol{x}} \mathcal{L}_k - \frac{\eta_{2,k}}{2} \left\| \nabla_{\boldsymbol{x}} \mathcal{L}_k \right\|^2 \\
&= -\eta_{2,k} \left\| B_k \Delta \boldsymbol{x}_k \right\|^2 + 2\eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k \left(B_k \Delta \boldsymbol{x}_k + G_k^T \Delta \boldsymbol{\lambda}_k \right) - \frac{\eta_{2,k}}{2} \left\| B_k \Delta \boldsymbol{x}_k + G_k^T \Delta \boldsymbol{\lambda}_k \right\|^2 \\
&= \eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k B_k \Delta \boldsymbol{x}_k + \eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k G_k^T \Delta \boldsymbol{\lambda}_k - \frac{\eta_{2,k}}{2} \left\| B_k \Delta \boldsymbol{x}_k \right\|^2 - \frac{\eta_{2,k}}{2} \left\| G_k^T \Delta \boldsymbol{\lambda}_k \right\|^2 \\
&\leq \eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k B_k \Delta \boldsymbol{x}_k + \eta_{2,k} \Delta \boldsymbol{x}_k{}^T B_k G_k^T \Delta \boldsymbol{\lambda}_k - \frac{\eta_{2,k}}{2} \left\| G_k^T \Delta \boldsymbol{\lambda}_k \right\|^2.
\end{aligned}$$

Combining the above two displays we get

$$\begin{aligned}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix} \leq & -\Delta\boldsymbol{x}_k{}^T\left((I+\eta_{2,k}H_k)B_k+\frac{\eta_{1,k}}{2}G_k^TG_k\right)\Delta\boldsymbol{x}_k-\frac{\eta_{1,k}}{2}\|c_k\|^2-\frac{\eta_{2,k}}{2}\|\nabla_{\boldsymbol{x}}\mathcal{L}_k\|^2\\ & +\eta_{2,k}\Delta\boldsymbol{x}_k{}^TB_kB_k\Delta\boldsymbol{x}_k+\eta_{2,k}\Delta\boldsymbol{\lambda}_k{}^TG_kB_k\Delta\boldsymbol{x}_k-\frac{\eta_{2,k}}{2}\|G_k^T\Delta\boldsymbol{\lambda}_k\|_2^2\\ & -\Delta\boldsymbol{\lambda}_k{}^TG_k\left(2I+\eta_{2,k}(B_k+H_k)\right)\Delta\boldsymbol{x}_k.\end{aligned}$$

Assuming $\eta_{1,k}\geq\eta_{2,k}$ at the moment and using Cauchy-Schwarz inequality, we get

$$\begin{aligned}&\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2-\Delta\boldsymbol{x}_k{}^T\left((I+\eta_{2,k}(H_k-B_k))B_k+\frac{\eta_{1,k}}{2}G_k^TG_k\right)\Delta\boldsymbol{x}_k-\frac{\eta_{2,k}}{2}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2\\ &\quad-\Delta\boldsymbol{\lambda}_k{}^TG_k\left(2I+\eta_{2,k}H_k\right)\Delta\boldsymbol{x}_k\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2-\eta_{2,k}\Delta\boldsymbol{x}_k{}^T(H_k-B_k)B_k\Delta\boldsymbol{x}_k-\Delta\boldsymbol{x}_k{}^TB_k\Delta\boldsymbol{x}_k-\frac{\eta_{1,k}}{2}\Delta\boldsymbol{x}_k{}^TG_k^TG_k\Delta\boldsymbol{x}_k\\ &\quad-\frac{\eta_{2,k}}{2}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2-2\Delta\boldsymbol{\lambda}_k{}^TG_k\Delta\boldsymbol{x}_k-\eta_{2,k}\Delta\boldsymbol{\lambda}_k{}^TG_kH_k\Delta\boldsymbol{x}_k\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2+\eta_{2,k}\|(H_k-B_k)\Delta\boldsymbol{x}_k\|\|B_k\Delta\boldsymbol{x}_k\|-\Delta\boldsymbol{x}_k{}^TB_k\Delta\boldsymbol{x}_k-\frac{\eta_{1,k}}{2}\Delta\boldsymbol{x}_k{}^TG_k^TG_k\Delta\boldsymbol{x}_k\\ &\quad-\frac{\eta_{2,k}}{2}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2+2\|\Delta\boldsymbol{\lambda}_k\|\|G_k\Delta\boldsymbol{x}_k\|+\eta_{2,k}\Upsilon\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|\|\Delta\boldsymbol{x}_k\|\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2+2\eta_{2,k}\Upsilon^2\|\Delta\boldsymbol{x}_k\|^2-\Delta\boldsymbol{x}_k{}^T\left(B_k+\frac{\eta_{1,k}}{2}G_k^TG_k\right)\Delta\boldsymbol{x}_k-\frac{\eta_{2,k}}{2}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2\\ &\quad+2\|\Delta\boldsymbol{\lambda}_k\|\|G_k\Delta\boldsymbol{x}_k\|+\eta_{2,k}\Upsilon\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|\|\Delta\boldsymbol{x}_k\|.\end{aligned}$$

Now we apply Young's inequality for the last two terms. Note that

$$\begin{aligned}\eta_{2,k}\Upsilon\|G_k^T\Delta\boldsymbol{\lambda}_k\|\|\Delta\boldsymbol{x}_k\| &= \left(\sqrt{\frac{\eta_{2,k}}{2}}\|G_k^T\Delta\boldsymbol{\lambda}_k\|\right)\left(\sqrt{2\eta_{2,k}}\Upsilon\|\Delta\boldsymbol{x}_k\|\right)\\ 2\|\Delta\boldsymbol{\lambda}_k\|\|G_k\Delta\boldsymbol{x}_k\| &= \left(\frac{\sqrt{\eta_{2,k}\xi_G}}{2}\|\Delta\boldsymbol{\lambda}_k\|\right)\left(\frac{4}{\sqrt{\eta_{2,k}\xi_G}}\|G_k\Delta\boldsymbol{x}_k\|\right).\end{aligned}$$

Using the above expression, we get

$$\eta_{2,k}\Upsilon\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|\|\Delta\boldsymbol{x}_k\|\leq\frac{\eta_{2,k}}{4}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2+\eta_{2,k}\Upsilon^2\|\Delta\boldsymbol{x}_k\|^2$$

and

$$2\|\Delta\boldsymbol{\lambda}_k\|\|G_k\Delta\boldsymbol{x}_k\|\leq\frac{\eta_{2,k}\xi_G}{8}\|\Delta\boldsymbol{\lambda}_k\|^2+\frac{8}{\eta_{2,k}\xi_G}\|G_k\Delta\boldsymbol{x}_k\|^2.$$

Using Lemma 5, we obtain

$$\begin{aligned}&\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2+3\eta_{2,k}\Upsilon^2\|\Delta\boldsymbol{x}_k\|^2-\frac{\eta_{2,k}}{4}\left\|G_k^T\Delta\boldsymbol{\lambda}_k\right\|^2+\frac{\eta,k2\xi_G}{8}\|\Delta\boldsymbol{\lambda}_k\|^2+\frac{8}{\eta_{2,k}\xi_G}\|G_k\Delta\boldsymbol{x}_k\|^2\\ &\quad-\Delta\boldsymbol{x}_k{}^T\left(B_k+\frac{\eta_{1,k}}{2}G_k^TG_k\right)\Delta\boldsymbol{x}_k\\ &\leq-\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2+3\eta_{2,k}\Upsilon^2\|\Delta\boldsymbol{x}_k\|^2-\frac{\eta_{2,k}\xi_G}{8}\|\Delta\boldsymbol{\lambda}_k\|^2-\Delta\boldsymbol{x}_k{}^T\left(B_k+\left(\frac{\eta_{1,k}}{2}-\frac{8}{\eta_{2,k}\xi_G}\right)G_k^TG_k\right)\Delta\boldsymbol{x}_k.\end{aligned}\tag{B.5}$$

In order to bound the second and fourth terms from the above display, we decompose $\Delta \boldsymbol{x}_k$ as $\Delta \boldsymbol{x}_k = \Delta \boldsymbol{u}_k + \Delta \boldsymbol{v}_k$ where $\Delta \boldsymbol{u}_k \in Null(G_k)$ and $\Delta \boldsymbol{v}_k \in Image(G_k^T)$. Then we have $\|\Delta \boldsymbol{x}_k\|^2 = \|\Delta \boldsymbol{u}_k\|^2 + \|\Delta \boldsymbol{v}_k\|^2$ and $\Delta \boldsymbol{v}_k = G_k^T \Delta \bar{\boldsymbol{v}}_k$ for some $\Delta \bar{\boldsymbol{v}}_k$. Using Lemma 5, we get $\|\Delta \boldsymbol{v}_k\|^2 = \|G_k^T \Delta \bar{\boldsymbol{v}}_k\|^2 \leq \kappa_G \|\Delta \bar{\boldsymbol{v}}_k\|^2$ and further obtain

$$\|G_k \Delta \boldsymbol{x}_k\|^2 = \|G_k \Delta \boldsymbol{v}_k\|^2 = \left\|G_k G_k^T \Delta \bar{\boldsymbol{v}}_k\right\|^2 \geq \xi_G^2 \|\Delta \bar{\boldsymbol{v}}_k\|^2 \geq \frac{\xi_G^2}{\kappa_G} \|\Delta \boldsymbol{v}_k\|^2 .$$

Using the above expressions and Cauchy-Schwarz inequality, and assuming $\eta_{1,k} \geq 16/(\eta_{2,k}\xi_G)$ at the momonet, we get

$$\begin{aligned}
&3\eta_{2,k}\Upsilon^2 \|\Delta \boldsymbol{x}_k\|^2 - \Delta \boldsymbol{x}_k{}^T \left(B_k + \left(\frac{\eta_{1,k}}{2} - \frac{8}{\eta_{2,k}\xi_G}\right) G_k^T G_k\right) \Delta \boldsymbol{x}_k \\
&= 3\eta_{2,k}\Upsilon^2 \|\Delta \boldsymbol{x}_k\|^2 - \Delta \boldsymbol{u}_k^T B_k \Delta \boldsymbol{u}_k - 2\Delta \boldsymbol{u}_k^T B_k \Delta \boldsymbol{v}_k - \Delta \boldsymbol{v}_k^T B_k \Delta \boldsymbol{v}_k - \left(\frac{\eta_{1,k}}{2} - \frac{8}{\eta_{2,k}\xi_G}\right) \|G_k \Delta \boldsymbol{x}_k\|^2 \\
&\leq 3\eta_{2,k}\Upsilon^2 \|\Delta \boldsymbol{x}_k\|^2 - \xi_B \|\Delta \boldsymbol{u}_k\|^2 + 2\Upsilon \|\Delta \boldsymbol{u}_k\| \|\Delta \boldsymbol{v}_k\| + \Upsilon \|\Delta \boldsymbol{v}_k\|^2 - \left(\frac{\eta_{1,k}}{2} - \frac{8}{\eta_{2,k}\xi_G}\right) \frac{\xi_G^2}{\kappa_G} \|\Delta \boldsymbol{v}_k\|^2 \\
&\leq \left(3\eta_{2,k}\Upsilon^2 - \xi_B\right) \|\Delta \boldsymbol{x}_k\|^2 + 2\Upsilon \|\Delta \boldsymbol{u}_k\| \|\Delta \boldsymbol{v}_k\| + (\xi_B + \Upsilon) \|\Delta \boldsymbol{v}_k\|^2 - \left(\frac{\eta_{1,k}\xi_G^2}{2\kappa_G} - \frac{8\xi_G}{\eta_{2,k}\kappa_G}\right) \|\Delta \boldsymbol{v}_k\|^2 .
\end{aligned}$$

Now we apply Young's inequality for the second term. Note that

$$2\Upsilon \|\Delta \boldsymbol{u}_k\| \|\Delta \boldsymbol{v}_k\| = (\sqrt{\xi_B} \|\Delta \boldsymbol{u}_k\|) \left(\frac{2\Upsilon}{\sqrt{\xi_B}} \|\Delta \boldsymbol{v}_k\|\right).$$

Using the above expression, we get

$$2\Upsilon \|\Delta \boldsymbol{u}_k\| \|\Delta \boldsymbol{v}_k\| \leq \frac{\xi_B}{2} \|\Delta \boldsymbol{u}_k\|^2 + \frac{2\Upsilon^2}{\xi_B} \|\Delta \boldsymbol{v}_k\|^2 \leq \frac{\xi_B}{2} \|\Delta \boldsymbol{x}_k\|^2 + \frac{2\Upsilon^2}{\xi_B} \|\Delta \boldsymbol{v}_k\|^2 .$$

This leads to

$$\begin{aligned}
&3\eta_{2,k}\Upsilon^2 \|\Delta \boldsymbol{x}_k\|^2 - \Delta \boldsymbol{x}_k{}^T \left(B_k + \left(\frac{\eta_{1,k}}{2} - \frac{8}{\eta_{2,k}\xi_G}\right) G_k^T G_k\right) \Delta \boldsymbol{x}_k \\
&\leq \left(3\eta_{2,k}\Upsilon^2 - \frac{\xi_B}{2}\right) \|\Delta \boldsymbol{x}_k\|^2 + \frac{2\Upsilon^2}{\xi_B} \|\Delta \boldsymbol{v}_k\|^2 + (\xi_B + \Upsilon) \|\Delta \boldsymbol{v}_k\|^2 - \left(\frac{\eta_{1,k}\xi_G^2}{2\kappa_G} - \frac{8\xi_G}{\eta_{2,k}\kappa_G}\right) \|\Delta \boldsymbol{v}_k\|^2 \\
&\leq \left(3\eta_{2,k}\Upsilon^2 - \frac{\xi_B}{2}\right) \|\Delta \boldsymbol{x}_k\|^2 + \left(\frac{2\Upsilon^2}{\xi_B} + \xi_B + \Upsilon + \frac{8\xi_G}{\eta_{2,k}\kappa_G} - \frac{\eta_{1,k}\xi_G^2}{2\kappa_G}\right) \|\Delta \boldsymbol{v}_k\|^2 .
\end{aligned}$$

Plugging the above inequality back into (B.5), we get

$$\begin{aligned}
&\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \\
&\leq -\frac{\eta_{2,k}}{2} \|\nabla \mathcal{L}_k\|^2 + \left(3\eta_{2,k}\Upsilon^2 - \frac{\xi_B}{2}\right) \|\Delta \boldsymbol{x}_k\|^2 + \left(\frac{2\Upsilon^2}{\xi_B} + \xi_B + \Upsilon + \frac{8\xi_G}{\eta_{2,k}\kappa_G} - \frac{\eta_{1,k}\xi_G^2}{2\kappa_G}\right) \|\Delta \boldsymbol{v}_k\|^2 - \frac{\eta_{2,k}\xi_G}{8} \|\Delta \boldsymbol{\lambda}_k\|^2 .
\end{aligned}$$

In order to make the upper bound negative, we let

$$\eta_{2,k} \leq \frac{\xi_B}{12\Upsilon^2}. \tag{B.6}$$

Furthermore, without loss of generality, we assume $\kappa_G \wedge \Upsilon/2 \geq 1 \geq \xi_B \vee \xi_G$. Using this assumption together with (B.6), we obtain

$$\frac{2\Upsilon^2}{\xi_B} + \xi_B + \Upsilon + \frac{8\xi_G}{\eta_{2,k}\kappa_G} \leq \frac{2\Upsilon^2}{\xi_B} + \frac{3\Upsilon}{2} + \frac{8\xi_G}{\eta_{2,k}\kappa_G} \leq \frac{3\Upsilon^2}{\xi_B} + \frac{8\xi_G}{\eta_{2,k}\kappa_G} \leq \frac{1}{4\eta_{2,k}} + \frac{8\xi_G}{\eta_{2,k}\kappa_G} \leq \frac{1}{4\eta_{2,k}} + \frac{8}{\eta_{2,k}} \leq \frac{8.5}{\eta_{2,k}}.$$

Using (B.6) togther with the above inequality, we get

$$\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\leq-\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2-\frac{\xi_B}{4}\left\|\Delta\boldsymbol{x}_k\right\|^2+\left(\frac{8.5}{\eta_{2,k}}-\frac{\eta_{1,k}\xi_G^2}{2\kappa_G}\right)\left\|\Delta\boldsymbol{v}_k\right\|^2-\frac{\eta_{2,k}\xi_G}{8}\left\|\Delta\boldsymbol{\lambda}_k\right\|^2.$$

In order to make the upper bound negative, we let

$$\eta_{1,k}\geq\frac{17\kappa_G}{\eta_{2,k}\xi_G^2}. \tag{B.7}$$

Note that (B.6) and (B.7) imply $\eta_{1,k}\geq\eta_{2,k}$ and $\eta_{1,k}\geq 16/(\eta_{2,k}\xi_G)$, hence, justify our previous assumption. Using (B.6) and (B.7), we finally have

$$\begin{aligned}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}&\leq-\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2-\frac{\eta_{2,k}\xi_G}{8}\left\|\Delta\boldsymbol{\lambda}_k\right\|^2-\frac{\xi_B}{4}\left\|\Delta\boldsymbol{x}_k\right\|^2\\ &\leq-\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2-\frac{\eta_{2,k}\xi_G}{8}\left\|\Delta\boldsymbol{\lambda}_k\right\|^2-\frac{\eta_{2,k}\xi_G}{8}\left\|\Delta\boldsymbol{x}_k\right\|^2\\ &\leq-\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2-\frac{\eta_{2,k}\xi_G}{8}\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2.\end{aligned} \tag{B.8}$$

Now we develop the second term of (B.4). Using Cauchy-Schwarz inequality and (B.6), we get

$$\begin{aligned}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}&=\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}^T\begin{pmatrix}I+\eta_{2,k}H_k & \eta_{1,k}G_k^T\\ \eta_{2,k}G_k & I\end{pmatrix}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_k\\ c_k\end{pmatrix}\\ &=-\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}^T\begin{pmatrix}(I+\eta_{2,k}H_k)B_k+\eta_{1,k}G_k^TG_k & (I+\eta_{2,k}H_k)G_k^T\\ G_k(I+\eta_{2,k}B_k) & \eta_{2,k}G_kG_k^T\end{pmatrix}\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\\ &\leq\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left((1+\eta_{2,k}\Upsilon)\Upsilon+(\eta_{1,k}+\eta_{2,k})\Upsilon^2+2(1+\eta_{2,k}\Upsilon)\Upsilon\right)\\ &\leq\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left(3\Upsilon+4\eta_{2,k}\Upsilon^2+\eta_{1,k}\Upsilon^2\right)\\ &\leq\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left(3\Upsilon+\frac{\xi_B}{3}+\eta_{1,k}\Upsilon^2\right).\end{aligned}$$

Furthermore, without loss of generality, we assume $\kappa_G\wedge\Upsilon/2\geq 1\geq\xi_B\vee\xi_G$. Using (B.6) and (B.7), we get $\eta_{1,k}\geq(17\kappa_G)/(\eta_{2,k}\xi_G^2)\geq 17/(\eta_{2,k}\xi_G)\geq(17\times 12\Upsilon^2)/(\xi_B\xi_G)$. Then we have $19/6\leq\eta_{1,k}\Upsilon$ and further obtain

$$3\Upsilon+\frac{\xi_B}{3}+\eta_{1,k}\Upsilon^2\leq\frac{19\Upsilon}{6}+\eta_{1,k}\Upsilon^2\leq 2\eta_{1,k}\Upsilon^2.$$

Using this inequality, we finally have

$$\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\leq 2\eta_{1,k}\Upsilon^2\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|. \tag{B.9}$$

Now plugging (B.8) and (B.9) back into (B.4) and using (A.3), we obtain

$$\begin{aligned}
\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} &= \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix} + \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\\
&\le -\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2 - \frac{\eta_{2,k}\xi_G}{8}\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 + 2\eta_{1,k}\Upsilon^2\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\\
&\le -\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2 - \frac{\eta_{2,k}\xi_G}{8}\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 + 2\delta_k\eta_{1,k}\Upsilon^2\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\\
&\le -\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2 - \left(\frac{\eta_{2,k}\xi_G}{8} - 2\delta_k\eta_{1,k}\Upsilon^2\right)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2.
\end{aligned}$$

In order to make the upper bound negative, we let

$$\delta_k \le \frac{\eta_{2,k}\xi_G}{16\eta_{1,k}\Upsilon^2}, \tag{B.10}$$

and obtain

$$\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} \le -\frac{\eta_{2,k}}{2}\left\|\nabla\mathcal{L}_k\right\|^2.$$

This ends proof of Lemma 10. ■

B.9. Proof of Lemma 11

Proof We suppose the event $\cap_{k=0}^{\infty}(\mathcal{A}_k\cap\mathcal{B}_k)$ happens. We start from finding the lower bound of $\frac{\delta_k^{\text{trial}}\eta_{1,k}}{\eta_{2,k}}$. Since the updating rule of the adaptive parameters (2.9) increases $\eta_{1,k}$ by a factor of ν^2 and decreases $\eta_{2,k}$ by a factor of $1/\nu$, we have that $\eta_{1,0}\le\eta_{1,k}$ and $\eta_{2,0}\ge\eta_{2,k}$ for all $k\ge 0$. Using this fact, we have that for any $k\ge 0$,

$$\begin{aligned}
2\Psi_k^2(3\Upsilon_k + 4\eta_{2,k}\Upsilon_k^2 + \eta_{1,k}\Upsilon_k^2) &\le 2\Psi^2(3\Upsilon + 4\eta_{2,0}\Upsilon^2 + \eta_{1,k}\Upsilon^2)\\
&\le 6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2 + 2\eta_{1,k}\Psi^2\Upsilon^2\\
&\le 6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2 + 2\eta_{1,k}\Psi^2\Upsilon^2\\
&\le \frac{\eta_{1,k}}{\eta_{1,0}}(6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2) + 2\eta_{1,k}\Psi^2\Upsilon^2\\
&\le \eta_{1,k}\left(\frac{6\Psi^2\Upsilon}{\eta_{1,0}} + \frac{8\eta_{2,0}\Psi^2\Upsilon^2}{\eta_{1,0}} + 2\Psi^2\Upsilon^2\right).
\end{aligned}$$

Using the above display, we get

$$\delta_k^{\text{trial}} = \left(\frac{1}{2}-\beta\right)\frac{\eta_{2,k}}{2\Psi_k^2(3\Upsilon_k + 4\eta_{2,k}\Upsilon_k^2 + \eta_{1,k}\Upsilon_k^2)} \ge \frac{\eta_{2,k}}{\eta_{1,k}}\left(\frac{1}{2}-\beta\right)\frac{\eta_{1,0}}{6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2 + 2\eta_{1,0}\Psi^2\Upsilon^2},$$

and obtain

$$\frac{\delta_k^{\text{trial}}\eta_{1,k}}{\eta_{2,k}} \ge \left(\frac{1}{2}-\beta\right)\frac{\eta_{1,0}}{6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2 + 2\eta_{1,0}\Psi^2\Upsilon^2}. \tag{B.11}$$

Using Lemma 10 and (B.11), we obtain the conditions for all adaptive parameters to be stabilized as

$$\eta_{1,k}\eta_{2,k} \ge \frac{17\kappa_G}{\xi_G^2},\quad \eta_{2,k}\le\frac{\xi_B}{12\Upsilon^2},\quad \frac{\delta_k\eta_{1,k}}{\eta_{2,k}} \le \frac{\xi_G}{16\Upsilon^2}\wedge\left(\frac{1}{2}-\beta\right)\frac{\eta_{1,0}}{6\Psi^2\Upsilon + 8\eta_{2,0}\Psi^2\Upsilon^2 + 2\eta_{1,0}\Psi^2\Upsilon^2}. \tag{B.12}$$

Note that the lower bound of $\eta_{1,k}\eta_{2,k}$ and the upper bound of $\eta_{2,k}$, $\delta_k\eta_{1,k}/\eta_{2,k}$ do not depend on k. The updating rule of the adaptive parameters (2.9) implies that $\eta_{1,k}\eta_{2,k}$ increases by a factor of ν, $\eta_{2,k}$ decreases by a factor of $1/\nu$, and $\delta_k\eta_{1,k}/\eta_{2,k}$ decreases at least by a factor of $1/\nu$. Thus, conditioned on the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$, all parameters are stabilized after sufficiently large outer iterations k. Now, using the fact that $\mathcal{P}(\mathcal{A}_k|\boldsymbol{x}_k, \boldsymbol{\lambda}_k) = 1$ and $\mathcal{P}(\mathcal{B}_k|\mathcal{A}_k, \boldsymbol{x}_k, \boldsymbol{\lambda}_k) = 1$, we have $\mathcal{P}(\mathcal{A}_k \cap \mathcal{B}_k|\boldsymbol{x}_k, \boldsymbol{\lambda}_k) = 1$. Using Boole's inequality,

$$\begin{aligned}
\mathcal{P}\left(\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)\right) &= 1 - \mathcal{P}(\cup_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)^c) \\
&\geq 1 - \sum_{k=0}^{\infty} \mathcal{P}((\mathcal{A}_k \cap \mathcal{B}_k)^c) \\
&= 1 - \sum_{k=0}^{\infty} \iint_{\mathcal{X}\times\Lambda} \mathcal{P}\left((\mathcal{A}_k \cap \mathcal{B}_k)^c|\boldsymbol{x}_k, \boldsymbol{\lambda}_k\right) \mathcal{P}\left((X_k, \Lambda_k) = (\boldsymbol{x}_k, \boldsymbol{\lambda}_k)\right) d(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) \\
&= 1 - \sum_{k=0}^{\infty} \iint_{\mathcal{X}\times\Lambda} 0 \cdot \mathcal{P}\left((X_k, \Lambda_k) = (\boldsymbol{x}_k, \boldsymbol{\lambda}_k)\right) d(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) \\
&= 1.
\end{aligned} \tag{B.13}$$

Therefore, the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens with probability 1, hence, after sufficiently large outer iterations k, all parameters are stabilized almost surely. This ends proof of Lemma 11. ■

B.10. Proof of Lemma 12

Proof Let $k \geq 0$ and we suppose the algorithm reaches $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ and the event $\mathcal{A}_k \cap \mathcal{B}_k$ happens. We start from establishing Lipschitz continuity of $\nabla\mathcal{L}_{\boldsymbol{\eta}}$. Note that

$$\nabla\mathcal{L}_{\boldsymbol{\eta}} = \begin{pmatrix} (I + \eta_2 H)\nabla_{\boldsymbol{x}}\mathcal{L} + \eta_1 G^T c \\ c + \eta_2 G \nabla_x \mathcal{L} \end{pmatrix}.$$

Using Assumption 1, we have $H, G, \nabla_{\boldsymbol{x}}\mathcal{L}$, and c are all Lipschitz continuous and bounded over $\mathcal{X}$. Using this fact we have $\nabla\mathcal{L}_{\boldsymbol{\eta}}$ is also Lipschitz continuous over $\mathcal{X}$. We denote Γ be the Lipschitz constant for $\nabla\mathcal{L}_{\boldsymbol{\eta}}$. Now we let C be a line segment given by the vector function $\boldsymbol{s}(t) = (\boldsymbol{x}_k + t\alpha_k\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + t\alpha_k\tilde{\Delta}\boldsymbol{\lambda}_k)$ where $0 \leq t \leq 1$. Using this expression together with the fundamental theorem for line integrals, we get

$$\begin{aligned}
&\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k + \alpha_k\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + \alpha_k\tilde{\Delta}\boldsymbol{\lambda}_k) \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \int_C \nabla\mathcal{L}_{\boldsymbol{\eta}} \cdot d\boldsymbol{s} \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \left(\nabla\mathcal{L}_{\boldsymbol{\eta}}^k\right)^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \int_C \nabla\mathcal{L}_{\boldsymbol{\eta}} \cdot d\boldsymbol{s} - \alpha_k \left(\nabla\mathcal{L}_{\boldsymbol{\eta}}^k\right)^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \left(\nabla\mathcal{L}_{\boldsymbol{\eta}}^k\right)^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \alpha_k \int_0^1 \nabla\mathcal{L}_{\boldsymbol{\eta}}^T(\boldsymbol{x}_k + t\alpha_k\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + t\alpha_k\tilde{\Delta}\boldsymbol{\lambda}_k)\begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} dt - \alpha_k \left(\nabla\mathcal{L}_{\boldsymbol{\eta}}^k\right)^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \begin{pmatrix} \nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \alpha_k \int_0^1 \left[\nabla\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k + t\alpha_k\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + t\alpha_k\tilde{\Delta}\boldsymbol{\lambda}_k) - \nabla\mathcal{L}_{\boldsymbol{\eta}}^k\right]^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} dt \\
&\leq \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \begin{pmatrix} \nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \alpha_k \int_0^1 \Gamma t\alpha_k \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 dt \\
&\leq \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \begin{pmatrix} \nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \alpha_k^2 \frac{\Gamma}{2} \left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2.
\end{aligned}$$

Using (A.3), we have

$$\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\| \leq \delta_k \left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\| + \left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\| \leq (\delta_k+1)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\| \leq 2\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|. \tag{B.14}$$

Using this expression together with Lemma 10, we get

$$\begin{aligned}
\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k+\alpha_k\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k+\alpha_k\tilde{\Delta}\boldsymbol{\lambda}_k) &\leq \mathcal{L}^k_{\boldsymbol{\eta}} + \alpha_k \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \alpha_k^2\frac{\Gamma}{2}\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\\
&\leq \mathcal{L}^k_{\boldsymbol{\eta}} + \alpha_k \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + 2\alpha_k^2\Gamma\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\\
&\leq \mathcal{L}^k_{\boldsymbol{\eta}} + \alpha_k \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + 2\alpha_k^2\Gamma\Psi^2\left\|\nabla\mathcal{L}_k\right\|^2\\
&\leq \mathcal{L}^k_{\boldsymbol{\eta}} + \alpha_k \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} - \frac{4\alpha_k^2\Gamma\Psi^2}{\eta_{2,k}}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\\
&\leq \mathcal{L}^k_{\boldsymbol{\eta}} + \left(1-\frac{4\Gamma\Psi^2\alpha_k}{\eta_2^\star}\right)\alpha_k \begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}^k_{\boldsymbol{\eta}}\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}^k_{\boldsymbol{\eta}}\end{pmatrix}^T \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix},
\end{aligned}$$

where $\eta_2^\star$ is the stabilized value of η_2. Using the above display, if

$$\left(1-\frac{4\Gamma\Psi^2\alpha_k}{\eta_2^\star}\right) \geq \beta \leftrightarrow \alpha_k \leq \frac{(1-\beta)\eta_2^\star}{4\Gamma\Psi^2},$$

then the Armijo condition is satisfied. Moreover, since the upper bound of α_k does not depend on k, we can find $l \geq 0$, independent of k, such that for any k, $0 < (\rho)^l \leq \frac{(1-\beta)\eta_2}{4\Gamma\Psi^2}$. Finally, if we let $\alpha_{\min} = (\rho)^l$, then for any outer iteration k, we have $0 < \alpha_{\min} \leq \alpha_k$. This ends proof of Lemma 12. ■

B.11. Proof of Lemma 13

Proof We suppose the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens. Using Assumption 1, we let $k_f, k_c, k_g > 0$ be constants such that $|f_k| \leq k_f$, $\|c_k\| \leq k_c$, and $\|\nabla f_k\| \leq k_g$. Using this fact together with Lemma 5 and Cauchy-Schwarz inequality, we get for any $k \geq 0$,

$$\begin{aligned}
\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) &= \mathcal{L}_k + \frac{\eta_{1,k}}{2}\left\|c_k\right\|^2 + \frac{\eta_{2,k}}{2}\left\|\nabla f_k + G_k^T\boldsymbol{\lambda}_k\right\|^2\\
&= f_k + \boldsymbol{\lambda}_k^T c_k + \frac{\eta_{1,k}}{2}\left\|c_k\right\|^2 + \frac{\eta_{2,k}}{2}\left\|\nabla f_k + G_k^T\boldsymbol{\lambda}_k\right\|^2\\
&= f_k + \boldsymbol{\lambda}_k^T c_k + \frac{\eta_{1,k}}{2}\left\|c_k\right\|^2 + \frac{\eta_{2,k}}{2}\left\|\nabla f_k\right\|^2 + \eta_{2,k}\boldsymbol{\lambda}_k^T G_k\nabla f_k + \frac{\eta_{2,k}}{2}\boldsymbol{\lambda}_k^T G_k G_k^T\boldsymbol{\lambda}_k\\
&\geq f_k + \boldsymbol{\lambda}_k^T(c_k + \eta_{2,k}G_k\nabla f_k) + \frac{\eta_{2,k}\xi_G}{2}\left\|\boldsymbol{\lambda}_k\right\|^2\\
&\geq f_k - \left\|\boldsymbol{\lambda}_k\right\|\left\|c_k + \eta_{2,k}G_k\nabla f_k\right\| + \frac{\eta_{2,k}\xi_G}{2}\left\|\boldsymbol{\lambda}_k\right\|^2\\
&\geq -\left|f_k\right| - \left\|\boldsymbol{\lambda}_k\right\|\left(\left\|c_k\right\| + \eta_{2,k}\left\|G_k\right\|\left\|\nabla f_k\right\|\right) + \frac{\eta_{2,k}\xi_G}{2}\left\|\boldsymbol{\lambda}_k\right\|^2\\
&\geq -k_f - \left\|\boldsymbol{\lambda}_k\right\|\left(k_c + \eta_{2,0}\Upsilon_G k_g\right) + \frac{\eta_2^\star\xi_G}{2}\left\|\boldsymbol{\lambda}_k\right\|^2,
\end{aligned}$$

where $\eta_2^\star$ is the stabilized value of η_2. Using Lemma 10 and 12, we have that $\mathcal{L}_\eta(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) \leq \mathcal{L}_\eta(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)$ for all $k \geq 0$. This leads to

$$\frac{\eta_2^\star \xi_G}{2} \|\boldsymbol{\lambda}_k\|^2 - \|\boldsymbol{\lambda}_k\| (k_c + \eta_{2,0} \Upsilon_G k_g) \leq \mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k, \boldsymbol{\lambda}_k) + k_f \leq |\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)| + k_f. \tag{B.15}$$

If we let $K_1 = \dfrac{\eta_2^\star \xi_G}{2} > 0$, $K_2 = k_c + \eta_{2,0} \Upsilon_G k_g > 0$, and $K_3 = |\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)| + k_f > 0$, then we get for any $k \geq 0$,

$$K_1 \|\boldsymbol{\lambda}_k\|^2 - K_2 \|\boldsymbol{\lambda}_k\| \leq K_3.$$

This implies that $\{\boldsymbol{\lambda}_k\}_{k \geq 0}$ is bounded. Using (B.13), the event $\cap_{k=0}^{\infty} (\mathcal{A}_k \cap \mathcal{B}_k)$ happens with probability 1, hence, $\{\boldsymbol{\lambda}_k\}_{k \geq 0}$ is bounded almost surely. This ends proof of Lemma 13. ■

B.12. Proof of Theorem 1

Proof We suppose the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens. Using Lemma 10 and 12, we have that for any $k \geq 0$,

$$\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k \end{pmatrix} \leq -\frac{\eta_{2,k}}{2} \|\nabla \mathcal{L}_k\|^2$$

and

$$\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k + \alpha_k \tilde{\Delta} \boldsymbol{x}_k, \boldsymbol{\lambda}_k + \alpha_k \tilde{\Delta} \boldsymbol{\lambda}_k) \leq \mathcal{L}_{\boldsymbol{\eta}}^k + \alpha_k \beta \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k \end{pmatrix}$$

for some $\alpha_k \in (0, 1]$. Combining the above displays, we have that for any $k \geq 0$,

$$\mathcal{L}_{\boldsymbol{\eta}}^{k+1} - \mathcal{L}_{\boldsymbol{\eta}}^k \leq -\frac{\eta_{2,k} \alpha_k \beta}{2} \|\nabla \mathcal{L}_k\|^2 \leq -\frac{\eta_2^\star \alpha_{\min} \beta}{2} \|\nabla \mathcal{L}_k\|^2 ,$$

where $\eta_2^\star$ is the stabilized value of η_2. Summing over k, we have

$$\sum_{k=0}^{\infty} \|\nabla \mathcal{L}_k\|^2 \leq \frac{2}{\eta_2^\star \alpha_k \beta} \left(\mathcal{L}_{\boldsymbol{\eta}}^0 - \min_{\mathcal{X} \times \Lambda} \{\mathcal{L}_\eta(\boldsymbol{x}, \boldsymbol{\lambda})\} \right) < \infty.$$

Therefore, $\|\nabla \mathcal{L}_k\| \to 0$ as $k \to \infty$. Using (B.13), the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens with probability 1, hence, $\mathcal{P}\left(\|\nabla \mathcal{L}_k\| \to 0 \text{ as } k \to \infty\right) = 1$. This ends proof of Theorem 1. ■

B.13. Proof of Theorem 2

Proof We suppose the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens. We first show for all sufficiently large k, almost surely, unit stepsize is admissible. It suffices to show that for all sufficiently large k,

$$\mathcal{L}_{\boldsymbol{\eta}}(\boldsymbol{x}_k + \tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + \tilde{\Delta}\boldsymbol{\lambda}_k) \leq \mathcal{L}_{\boldsymbol{\eta}}^k + \beta \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \quad \textit{almost surely}. \tag{B.16}$$

Using the fact that for any $k \geq 0$,

$$\nabla \mathcal{L}_{\boldsymbol{\eta}}^k = \begin{pmatrix} (I + \eta_{2,k} H_k)\nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{1,k} G_k^T c_k \\ c_k + \eta_{2,k} G_k \nabla_{\boldsymbol{x}} \mathcal{L}_k \end{pmatrix}$$

and

$$\nabla(M \cdot \boldsymbol{c}) = \boldsymbol{c} \cdot \nabla M^T + M \cdot \nabla \boldsymbol{c}$$

where $M \in \mathbb{R}^{m \times n}$ and $\boldsymbol{c} \in \mathbb{R}^n$, we get

$$\begin{aligned}
\nabla_{\boldsymbol{xx}}^2 \mathcal{L}_{\boldsymbol{\eta}}^k &= \nabla_{\boldsymbol{x}} \left(\nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{2,k} H_k \nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{1,k} G_k^T c_k \right) \\
&= H_k + \eta_{2,k} \left(\nabla_{\boldsymbol{x}} \mathcal{L}_k \cdot \nabla_{\boldsymbol{x}} H_k + H_k^2 \right) + \eta_{1,k} \left(c_k \cdot \nabla G_k + G_k^T G_k \right), \\
\nabla_{\boldsymbol{\lambda\lambda}}^2 \mathcal{L}_{\boldsymbol{\eta}}^k &= \nabla_{\boldsymbol{\lambda}} \left(c_k + \eta_{2,k} G_k \nabla_{\boldsymbol{x}} \mathcal{L}_k \right) = \eta_{2,k} G_k \nabla_{\boldsymbol{x\lambda}}^2 \mathcal{L}_k = \eta_{2,k} G_k G_k^T, \\
\nabla_{\boldsymbol{x\lambda}}^2 \mathcal{L}_{\boldsymbol{\eta}}^k &= \nabla_{\boldsymbol{\lambda}} \left(\nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{2,k} H_k \nabla_{\boldsymbol{x}} \mathcal{L}_k + \eta_{1,k} G_k^T c^k \right) = G_k^T + \eta_{2,k} \left(\nabla_{\boldsymbol{x}} \mathcal{L}_k \cdot \nabla_{\boldsymbol{\lambda}} H_k + H_k G_k^T \right),
\end{aligned}$$

where $\nabla_{\boldsymbol{x}} H_k = \nabla_{\boldsymbol{xxx}}^3 f_k + \sum_{i=1}^m \lambda_{i,k} \nabla_{\boldsymbol{xxx}}^3 c_{i,k}$ and $\nabla_{\boldsymbol{\lambda}} H_k = \nabla_{\boldsymbol{xx}}^2 c_k$. Using Assumption 4, we have that the third derivatives of f and c are continuous, hence, $\nabla^2 \mathcal{L}_{\boldsymbol{\eta}}^k$ is continuous over $\mathcal{X}$. Now we let

$$\mathcal{H}_k = \begin{pmatrix} H_k + \eta_{2,k} H_k^2 + \eta_{1,k} G_k^T G_k & G_k^T + \eta_{2,k} H_k G_k^T \\ G_k + \eta_{2,k} G_k H_k & \eta_{2,k} G_k G_k^T \end{pmatrix}.$$

Using $\|\nabla \mathcal{L}_k\| = \left\| \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_k \\ c_k \end{pmatrix} \right\| = o(1)$, we have $\nabla^2 \mathcal{L}_{\boldsymbol{\eta}}^k = \mathcal{H}_k + o(1)$. Applying Taylor's theorem to the augmented Lagrangian merit function about $(\boldsymbol{x}_k, \boldsymbol{\lambda}_k)$ yields

$$\begin{aligned}
&\mathcal{L}_{\eta}(\boldsymbol{x}_k + \tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}_k + \tilde{\Delta}\boldsymbol{\lambda}_k) \\
&\leq \mathcal{L}_{\boldsymbol{\eta}}^k + \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \nabla^2 \mathcal{L}_{\boldsymbol{\eta}}^k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + o\left(\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 \right) \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \mathcal{H}_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + o\left(\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 \right) \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2} \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \mathcal{H}_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + o\left(\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 \right) \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2} \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \mathcal{H}_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \\
&\quad - \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \begin{pmatrix} (I + \eta_{2,k} H_k) B_k + \eta_{1,k} G_k^T G_k & (I + \eta_{2,k} H_k) G_k^T \\ G_k (I + \eta_{2,k} B_k) & \eta_{2,k} G_k G_k^T \end{pmatrix} \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} + o\left(\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 \right) \\
&= \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2} \begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \mathcal{H}_k \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k - \Delta\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k - \Delta\boldsymbol{\lambda}_k \end{pmatrix} \\
&\quad + \frac{1}{2} \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix}^T \left(\mathcal{H}_k - \begin{pmatrix} (I + \eta_{2,k} H_k) B_k + \eta_{1,k} G_k^T G_k & (I + \eta_{2,k} H_k) G_k^T \\ G_k (I + \eta_{2,k} B_k) & \eta_{2,k} G^k G_k^T \end{pmatrix} \right) \begin{pmatrix} \Delta\boldsymbol{x}_k \\ \Delta\boldsymbol{\lambda}_k \end{pmatrix} + o\left(\left\| \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} \right\|^2 \right).
\end{aligned}$$

This leads to

$$\mathcal{L}_\eta(\boldsymbol{x}_k+\tilde{\Delta}\boldsymbol{x}_k,\boldsymbol{\lambda}_k+\tilde{\Delta}\boldsymbol{\lambda}_k) \le \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \frac{1}{2}\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}^T\mathcal{H}_k\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}$$
$$+\frac{1}{2}\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}^T\begin{pmatrix}(I+\eta_{2,k}H_k)(H_k-B_k) & \mathbf{0}\\ \eta_{2,k}G_k(H_k-B_k) & \mathbf{0}\end{pmatrix}\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix} + o\left(\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\right).$$

Now we let $\Upsilon_k = \|B_k\|\vee\|H_k\|\vee\|G_k\|$. Using Assumption 5, we get $\|(H_k-B_k)\Delta\boldsymbol{x}_k\| \le \|(H_k-B_k)\|\,\|\Delta\boldsymbol{x}_k\| = o(\|\Delta\boldsymbol{x}_k\|)$. Using this expression together with (A.3), (B.14), and $o(\|(\tilde{\Delta}\boldsymbol{x}_k,\tilde{\Delta}\boldsymbol{\lambda}_k)\|) = o(\|(\Delta\boldsymbol{x}_k,\Delta\boldsymbol{\lambda}_k)\|)$, we have that for any $k\ge 0$,

$$\begin{aligned}
&\mathcal{L}_\eta(\boldsymbol{x}_k+\tilde{\Delta}\boldsymbol{x}_k,\boldsymbol{\lambda}_k+\tilde{\Delta}\boldsymbol{\lambda}_k)\\
&\le \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \frac{1}{2}\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|\|\mathcal{H}_k\|\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\\
&\quad+\frac{1}{2}\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|(\|I+\eta_{2,k}H_k\|\,\|(H_k-B_k)\Delta\boldsymbol{x}_k\| + \|\eta_{2,k}G_k\|\,\|(H_k-B_k)\Delta\boldsymbol{x}_k\|) + o\left(\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\right)\\
&\le \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \frac{1}{2}\|\mathcal{H}_k\|\left(2\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\right)\left(\delta_k\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|\right)\\
&\quad+(1+2\eta_{2,k}\Upsilon_k)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\| o(\|\Delta\boldsymbol{x}_k\|) + o\left(\left\|\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\right)\\
&\le \mathcal{L}_{\boldsymbol{\eta}}^k + \frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 + o\left(\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\right).
\end{aligned}\tag{B.17}$$

Using the fact that for any $k\ge 0$, $\delta_k \le \delta_k^{\text{trial}} = \left(\frac{1}{2}-\beta\right)\frac{\eta_{2,k}}{2\Psi_k^2(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)}$ and Lemma 10, we have that for any $k\ge 0$,

$$\begin{aligned}
&\delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2) \le \left(\frac{1}{2}-\beta\right)\frac{\eta_{2,k}}{2\Psi_k^2}\\
&\Rightarrow \delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 \le \left(\frac{1}{2}-\beta\right)\frac{\eta_{2,k}}{2\Psi_k^2}\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\\
&\Rightarrow \delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 \le \left(\frac{1}{2}-\beta\right)\frac{\eta_{2,k}}{2}\|\nabla\mathcal{L}_k\|^2\\
&\Rightarrow \delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 \le -\left(\frac{1}{2}-\beta\right)\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\\
&\Rightarrow \frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \delta(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 \le \beta\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}.
\end{aligned}$$

We let $K_1\ge 0$ be the outer iteration such that for any $k\ge K_1$,

$$\frac{1}{2}\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix} + \delta_k(3\Upsilon_k+4\eta_{2,k}\Upsilon_k^2+\eta_{1,k}\Upsilon_k^2)\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2 + o\left(\left\|\begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\right\|^2\right) \le \beta\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}.$$

Plugging the above inequality back into (B.17), we have that for any $k\ge K_1$,

$$\mathcal{L}_\eta(\boldsymbol{x}^k+\tilde{\Delta}\boldsymbol{x}_k,\boldsymbol{\lambda}^k+\tilde{\Delta}\boldsymbol{\lambda}_k) \le \mathcal{L}_{\boldsymbol{\eta}}^k + \beta\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}.$$

Using (B.13), the event $\cap_{k=0}^{\infty}(\mathcal{A}_k \cap \mathcal{B}_k)$ happens with probability 1, hence,

$$\mathcal{P}\left(\cap_{k=K_1}^{\infty}\left\{\mathcal{L}_\eta(\boldsymbol{x}^k+\tilde{\Delta}\boldsymbol{x}_k, \boldsymbol{\lambda}^k+\tilde{\Delta}\boldsymbol{\lambda}_k) \le \mathcal{L}_{\boldsymbol{\eta}}^k + \beta\begin{pmatrix}\nabla_{\boldsymbol{x}}\mathcal{L}_{\boldsymbol{\eta}}^k\\ \nabla_{\boldsymbol{\lambda}}\mathcal{L}_{\boldsymbol{\eta}}^k\end{pmatrix}^T\begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k\end{pmatrix}\right\}\right) = 1.$$

Next, we show for all sufficiently large k,

$$\left\|\begin{pmatrix}\boldsymbol{x}_k+\tilde{\Delta}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\tilde{\Delta}\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}\right\| \le \delta^\star\left\|\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}\right\| \quad \textit{almost surely},$$

where $\delta^\star$ be the stabilized value of $\delta \in (0,1)$. We start from dividing $\begin{pmatrix}\boldsymbol{x}_k+\tilde{\Delta}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\tilde{\Delta}\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}$ into two terms as follows:

$$\begin{pmatrix}\boldsymbol{x}_k+\tilde{\Delta}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\tilde{\Delta}\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} = \begin{pmatrix}\boldsymbol{x}_k+\Delta\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\Delta\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} + \begin{pmatrix}\tilde{\Delta}\boldsymbol{x}_k-\Delta\boldsymbol{x}_k\\ \tilde{\Delta}\boldsymbol{\lambda}_k-\Delta\boldsymbol{\lambda}_k\end{pmatrix}. \tag{B.18}$$

First we develop the first term in (B.18). Using Assumption 1– 2 together with $\nabla\mathcal{L}_\star = \mathbf{0}$, we obtain for any $k \ge 0$,

$$\begin{aligned}
\begin{pmatrix}\boldsymbol{x}_k+\Delta\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\Delta\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} &= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} + \begin{pmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{\lambda}_k\end{pmatrix}\\
&= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} - \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\nabla\mathcal{L}_k\\
&= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\left(\begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} - \nabla\mathcal{L}_k\right)\\
&= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\left(\begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} - (\nabla\mathcal{L}_k - \nabla\mathcal{L}_\star)\right).
\end{aligned} \tag{B.19}$$

Using Assumption 1, we know $\nabla^2\mathcal{L}$ is continuous over $\mathcal{X}$. Using this fact, we apply Taylor's theorem and obtain

$$\begin{aligned}
\nabla\mathcal{L}_k - \nabla\mathcal{L}_\star &= \int_0^1 \nabla^2\mathcal{L}\left(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k), \boldsymbol{\lambda}_k+t(\boldsymbol{\lambda}^\star-\boldsymbol{\lambda}_k)\right)\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}dt\\
&= \int_0^1 \begin{pmatrix}H\left(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k), \boldsymbol{\lambda}_k+t(\boldsymbol{\lambda}^\star-\boldsymbol{\lambda}_k)\right) & G^T(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k))\\ G(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k)) & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}dt.
\end{aligned}$$

Now we let $H(t) = H\left(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k), \boldsymbol{\lambda}_k+t(\boldsymbol{\lambda}^\star-\boldsymbol{\lambda}_k)\right)$ and $G(t) = G(\boldsymbol{x}_k+t(\boldsymbol{x}^\star-\boldsymbol{x}_k))$. Then we rewrite the above display as

$$\nabla\mathcal{L}_k - \nabla\mathcal{L}_\star = \int_0^1 \begin{pmatrix}H(t) & G^T(t)\\ G(t) & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}dt.$$

Plugging the above display back into (B.19), we obtain for any $k \ge 0$,

$$\begin{aligned}
\begin{pmatrix}\boldsymbol{x}_k+\Delta\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k+\Delta\boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} &= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\left(\begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix} - (\nabla\mathcal{L}_k - \nabla\mathcal{L}_\star)\right)\\
&= \begin{pmatrix}B_k & G_k^T\\ G_k & \mathbf{0}\end{pmatrix}^{-1}\left(\int_0^1\begin{pmatrix}B_k-H(t) & G_k^T-G(t)^T\\ G_k-G(t) & \mathbf{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{x}_k-\boldsymbol{x}^\star\\ \boldsymbol{\lambda}_k-\boldsymbol{\lambda}^\star\end{pmatrix}dt\right)
\end{aligned}$$

Using Assumption 1, we know H and G are Lipschitz continuous over $\mathcal{X}$. Let $\Gamma_1, \Gamma_2 > 0$ be the Lipschitz constants for H and G respectively. Using this fact together with Lemma 6 and Assumption 5, and taking ℓ_2

norm on both sides, we have that for any $k \geq 0$,

$$\begin{aligned}
\left\| \begin{pmatrix} \boldsymbol{x}_k + \Delta \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| &\leq \left\| \begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0} \end{pmatrix}^{-1} \right\| \left\| \int_0^1 \begin{pmatrix} B_k - H(t) & G_k^T - G(t)^T \\ G_k - G(t) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} dt \right\| \\
&\leq \left\| \begin{pmatrix} B_k & G_k^T \\ G_k & \mathbf{0} \end{pmatrix}^{-1} \right\| \int_0^1 \left\| \begin{pmatrix} B_k - H(t) & G_k^T - G(t)^T \\ G_k - G(t) & \mathbf{0} \end{pmatrix} \right\| \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| dt \\
&\leq \Psi \int_0^1 \left(\|B_k - H_k\| + \|H_k - H(t)\| + 2\|G_k - G(t)\| \right) \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| dt \\
&\leq \Psi \int_0^1 \left(o(1) + \Gamma_1 t \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| + 2\Gamma_2 t \left\| \boldsymbol{x}_k - \boldsymbol{x}^\star \right\| \right) \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| dt \\
&\leq \Psi \int_0^1 \left(o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right) + \Gamma_1 t \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 + 2\Gamma_2 t \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 \right) dt \\
&\leq \Psi o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right) + \Psi \Gamma_1 \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 \int_0^1 t dt + 2\Psi\Gamma_2 \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 \int_0^1 t dt \\
&\leq o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right) + \frac{\Psi\Gamma_1}{2} \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 + \Psi\Gamma_2 \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 \\
&\leq o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right) + O\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|^2 \right) \\
&\leq o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right). \qquad \text{(B.20)}
\end{aligned}$$

Furthermore, using (B.20) we have

$$\left\| \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \right\| \leq \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| + \left\| \begin{pmatrix} \boldsymbol{x}_k + \Delta \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \leq \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| + o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right).$$

Using the above inequality together with (A.3) and (B.20), and taking ℓ_2 norm on both sides of (B.18), we have that for any $k \geq 0$,

$$\begin{aligned}
\left\| \begin{pmatrix} \boldsymbol{x}_k + \tilde{\Delta} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \tilde{\Delta} \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| &\leq \left\| \begin{pmatrix} \tilde{\Delta} \boldsymbol{x}_k - \Delta \boldsymbol{x}_k \\ \tilde{\Delta} \boldsymbol{\lambda}_k - \Delta \boldsymbol{\lambda}_k \end{pmatrix} \right\| + \left\| \begin{pmatrix} \boldsymbol{x}_k + \Delta \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \Delta \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \\
&\leq \delta_k \left\| \begin{pmatrix} \Delta \boldsymbol{x}_k \\ \Delta \boldsymbol{\lambda}_k \end{pmatrix} \right\| + o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right) \\
&\leq \delta_k \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| + o\left(\left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right).
\end{aligned}$$

We let $\bar{K} \geq 0$ be the outer iteration such that $\left\| \begin{pmatrix} \boldsymbol{x}_k + \tilde{\Delta} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \tilde{\Delta} \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \leq \delta^\star \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|$ holds where $\delta^\star$ be the stabilized value of $\delta \in (0, 1)$. Now we let $K_2 = \bar{K} \vee K_1$. Then, we have for any $k \geq K_2$,

$$\left\| \begin{pmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| = \left\| \begin{pmatrix} \boldsymbol{x}_k + \tilde{\Delta} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k + \tilde{\Delta} \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \leq \delta^\star \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\|.$$

Using (B.13), the event $\cap_{k=0}^{\infty} (\mathcal{A}_k \cap \mathcal{B}_k)$ happens with probability 1, hence,

$$\mathcal{P}\left(\cap_{k=K_2}^{\infty} \left\{ \left\| \begin{pmatrix} \boldsymbol{x}_{k+1} - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \leq \delta^\star \left\| \begin{pmatrix} \boldsymbol{x}_k - \boldsymbol{x}^\star \\ \boldsymbol{\lambda}_k - \boldsymbol{\lambda}^\star \end{pmatrix} \right\| \right\} \right) = 1.$$

This ends proof of Theorem 2. ■

Appendix C. Algorithms

Algorithm 1: Adaptive Inexact SQP via Iterative Randomized Sketching

Input: Initial iterate $(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)$;
Scalars $\eta_{1,0}$, $\eta_{2,0}$, $\delta_0 \in (0,1)$; $\xi_B \in (0,1]$, $\beta \in (0,1/2)$; $\nu > 1$;
for $k = 0, 1, 2, \ldots$ **do**
  Compute f_k, ∇f_k, c_k, G_k, H_k, and generate B_k
  Compute δ_k^{trial} by (2.6)
  Set $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k) \leftarrow (\mathbf{0}, \mathbf{0})$ and compute $\boldsymbol{r}_k$ by (2.4)
  Set $\delta_k \leftarrow (\delta_k \wedge \delta_k^{\text{trial}})$
  while *Step Acceptance Condition does not hold* **do**
    while $\|\boldsymbol{r}_k\| > \delta_k \dfrac{\|\nabla \mathcal{L}_k\|}{\|\Gamma_k\| \, \Psi_k}$ **do**
      Generate $S \sim \mathcal{P}$, update $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ by (2.5), and compute $\boldsymbol{r}_k$ by (2.4)
    end
    if $\begin{pmatrix} \nabla_{\boldsymbol{x}} \mathcal{L}_{\boldsymbol{\eta}}^k \\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}_{\boldsymbol{\eta}}^k \end{pmatrix}^T \begin{pmatrix} \tilde{\Delta}\boldsymbol{x}_k \\ \tilde{\Delta}\boldsymbol{\lambda}_k \end{pmatrix} > -\dfrac{\eta_{2,k}}{2} \|\nabla \mathcal{L}_k\|^2$ **then**
      Set $\eta_{1,k} \leftarrow \eta_{1,k}\nu^2$ and $\eta_{2,k} \leftarrow \eta_{2,k}/\nu$
      Update δ_k^{trial} by (2.6) and set $\delta_k \leftarrow (\delta_k/\nu^4 \wedge \delta_k^{\text{trial}})$
    end
  end
  Select α_k to satisfy (2.10) using backtracking
  Update iterate by (2.11)
  Set $\eta_{1,k+1} \leftarrow \eta_{1,k}$, $\eta_{2,k+1} \leftarrow \eta_{2,k}$, and $\delta_{k+1} \leftarrow \delta_k$
end

Algorithm 2 and 3 use the ℓ_1 penalized merit function of the form $\phi_\pi(\boldsymbol{x}) = f(\boldsymbol{x}) + \pi\|c(\boldsymbol{x})\|_1$. Since $\phi_\pi(\boldsymbol{x})$ is not differentiable and its directional derivative is hard to compute, we use the upper bound of the directional derivative of the merit function ϕ_π along a step $\tilde{\Delta}\boldsymbol{x}_k$,

$$\tilde{D}_\phi(\tilde{\Delta}\boldsymbol{x}_k; \pi_k) \leq \nabla f_k{}^T \tilde{\Delta}\boldsymbol{x}_k - \pi_k \left(\|c_k\|_1 - \|\boldsymbol{r}_k\|_1\right),$$

when we check if $\tilde{\Delta}\boldsymbol{x}_k$ is a descent direction of ϕ_π. *Termination Test 1*, *Termination Test 2*, *Model Reduction Condition*, and π_k^{trial} are referred to in [4].

Algorithm 2: [4] with ℓ_1 penalized merit function

Input: Initial iterate $(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)$; Scalars $\kappa_1, \epsilon, \tau, \sigma, \eta \in (0,1)$; $\xi_B \in (0,1]$; $\pi_0, \beta, \kappa, \kappa_2 > 0$;
for $k = 0, 1, 2, \ldots$ **do**
  Compute f_k, ∇f_k, c_k, G_k, H_k, and generate B_k
  Set $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k) \leftarrow (\mathbf{0}, \mathbf{0})$ and compute $\boldsymbol{r}_k$ by (2.4)
  while *Termination Test 1 AND Termination Test 2 are not satisfied* **do**
    Generate $S \sim \mathcal{P}$, update $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ by (2.5), and compute $\boldsymbol{r}_k$ by (2.4)
  end
  if *Termination Test 2 is satisfied and Model Reduction Condition does not hold* **then**
    Set $\pi_k \leftarrow \pi_k^{\text{trial}} + 10^{-4}$
  end
  Select α_k to satisfy (2.10) using backtracking
  Update iterate by (2.11)
  Set $\pi_{k+1} \leftarrow \pi_k$
end

Algorithm 3: Adaptive version of Algorithm 2

Input: Initial iterate $(\boldsymbol{x}_0, \boldsymbol{\lambda}_0)$; Scalars $\kappa_0, \eta \in (0,1)$, $\xi_B \in (0,1]$, $\pi_0 > 0$; $\nu > 1$;
for $k = 0, 1, 2, \ldots$ **do**
  Compute f_k, ∇f_k, c_k, G_k, H_k, and generate B_k
  Set $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k) \leftarrow (\mathbf{0}, \mathbf{0})$ and compute $\boldsymbol{r}_k$ by (2.4)
  while *Termination Test 1 is not satisfied* **do**
    while $\|\boldsymbol{r}_k\|_1 > \kappa \|\nabla \mathcal{L}_k\|_1$ **do**
      Generate $S \sim \mathcal{P}$, update $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ by (2.5), and compute $\boldsymbol{r}_k$ by (2.4)
    end
    if *Model Reduction Condition does not hold* **then**
      Set $\pi_k \leftarrow \pi_k \nu$ and $\kappa_k \leftarrow \kappa_k / \nu^2$
    end
  end
  Select α_k to satisfy (2.10) using backtracking
  Update iterate by (2.11)
  Set $\pi_{k+1} \leftarrow \pi_k$
end

Appendix D. Further Experiments

Implementation Details.

1. Algorithm 1: The proposed algorithm. The parameters are set as $\eta_{1,0} = 1$, $\eta_{2,0} = 0.1$, $\delta_0 = 0.1$, $\xi_B = 0.1$, $\beta = 0.1$, $\nu = 1.4$, $\rho = 0.5$, $\theta_k = 1$.

2. Algorithm 2: [4] with the ℓ_1 penalized merit function. We follow the parameter setup used in [4]. The parameters are set as $\pi_0 = 1$, $\kappa = 1$, $\kappa_1 = 0.1$, $\epsilon = 0.1$, $\tau = 0.1$, $\eta = 10^{-8}$. Likewise, the remaining parameters are set as $\xi_B = 0.1$, $\sigma = \tau(1-\epsilon)$, and $\kappa_2 = \beta = \dfrac{\|\nabla \mathcal{L}^0\|_1}{\|c^0\|_1 + 1} \vee 1$.

3. Algorithm 3: Adaptive version of Algorithm 2. The parameters are set as $\pi_0 = 1$, $\kappa_0 = 0.1$, $\eta = 10^{-8}$, $\xi_B = 0.1$, $\beta = 0.1$, $\nu = 1.4$.

For the Hessian modification, we regularize the Hessian H_k by $B_k = H_k + (\xi_B + \|H_k\|)I_n$ whenever H_k does not satisfy Assumption 2. The stopping criterion is set as:

$$\|\nabla \mathcal{L}_k\| \leq 10^{-4} \quad \text{OR} \quad k \geq 10^4.$$

If the algorithm terminates by the former stopping criterion, we say the algorithm converges, otherwise the latter stopping criterion would be satisfied.

D.1. CUTEst

Among all the problems in the CUTEst test set, we selected the problems for which f is not a constant objective with $n < 1000$, containing only equality constraints, positive definiteness of $G_k G_k^T$ at all iterates of all algorithms that we ran. This selection scheme yields a total of 47 problems. Throughout the experiments, we use the initial value of primal-dual variables which are provided by the CUTEst package. For each algorithm, we average over 10 independent runs.

We compare Algorithms 2 and 3. Remark that *Step acceptance condition* of Algorithm 1 is similar with the *Termination Test 1* in Algorithm 2, in that both conditions require an inexact step $(\tilde{\Delta}\boldsymbol{x}_k, \tilde{\Delta}\boldsymbol{\lambda}_k)$ to acheive certain accuracy and ensure a step to be a descent direction of the merit function. However, Algorithm 1 adaptively controls the accuracy of an inexact solution of (2.3), on the other hand, Algorithm 2 uses a consistent bound to the accuracy throughout all iterations. Since Algorithm 3 adaptively controls the accuracy of an inexact search direction, but relies on *Termination Test 1* in Algorithm 2, we can view Algorithm 3 as an adaptive version of Algorithm 2.

We present the comparison between Algorithms 2 and 3 on CUTEst set in Figure 2. From Figure 2, we observe that Algorithm 3 is superior than Algorithm 2 in all three criteria. This is because Algorithm 3 adaptively controls the accuracy of the inexact solution and this adaptive scheme yields tighter bounds on the residuals of the iterative solver. This results in steeper decrease in the merit function at each iteration and smaller number of outer iterations. Since both algorithms use the ℓ_1 penalized merit function, both algorithm do not involve gradient and Jacobian evaluations when we find a stepsize α_k.

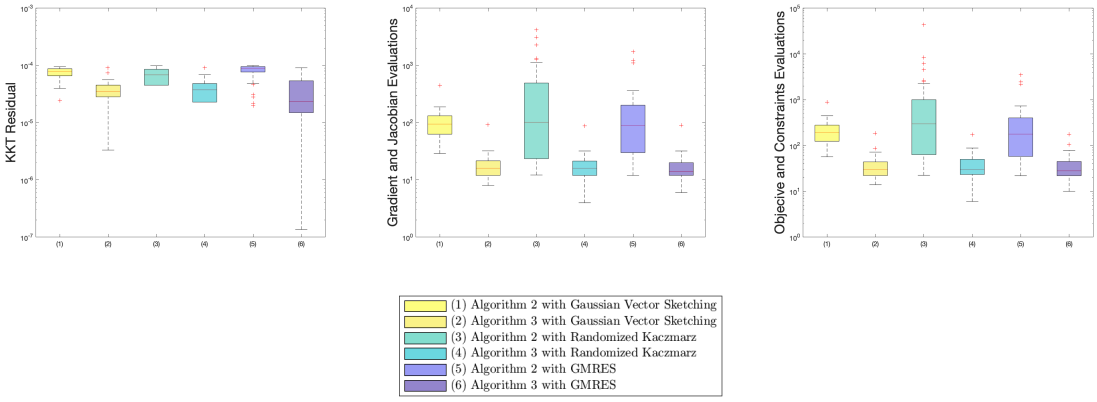

Figure 2: KKT residual, number of gradient and Jacobian evaluations, and number of objective and constraints evaluations boxplots for Algorithm 2 and Algorithm 3 on CUTEst problems.

D.2. Constrained Logistic Regression

We consider equality-constrained logistic regression problems of the form

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} \log\left(1+\exp(-y_i \cdot < X_{i,:}, \boldsymbol{x} >)\right) \quad \text{s.t.} \quad A\boldsymbol{x} = \boldsymbol{b}, \quad \|\boldsymbol{x}\|^2 = 1,$$

where $X \in \mathbb{R}^{N\times n}$ is a feature matrix with n feature dimensions and N data points, $\boldsymbol{y} \in \{-1,1\}^N$ contains corresponding label data, $A \in \mathbb{R}^{m\times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$. We follow the experiment details in [1]. Among all datasets in the LIBSVM collection, we consider 7 binary classification datasets for which $12 \le n \le 1000$, $256 \le N \le 100000$, and positive definiteness of $G_k G_k^T$ at all iterates of all algorithms we ran. For the linear constraints, we fix $m = 10$ and randomly generate each entry of A and $\boldsymbol{b}$ from a standard normal distribution for each problem. Combining with the norm constraint, we use total of 11 number of constraints. For all problems and algorithms, we set the initial primal and dual iterates as the vector of all ones. For each algorithm, we average over 5 independent runs. Details of the datasets are given in Table 1.

Table 1: Dataset Statistics.

Dataset	feature dimension	# data points
a9a	123	32,561
ionosphere	34	351
mushrooms	112	8,124
phishing	68	11,055
sonar	60	208
splice	60	1,000
w8a	300	49,749

We evaluate Algorithm 1 and Algorithm 3 with three criteria on the LIBSVM datasets. The boxplots for the criteria are shown in Figure 3. From Figure 3, we observe that Algorithm 1 outperforms

Algorithm 3 in terms of the KKT residual and number of objective and constraints evaluations, but Algorithm 3 has lower number of gradient and Jacobian evaluations than Algorithm 1 as we observed in Subsection D.1.

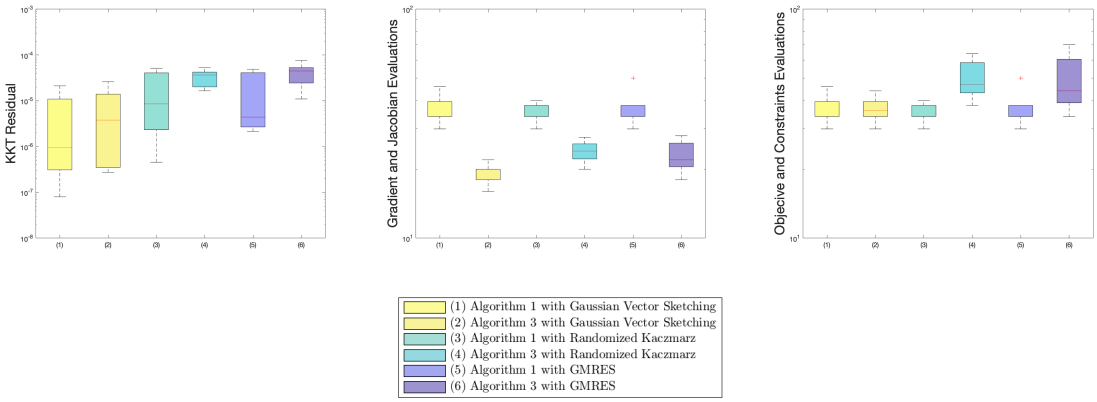

Figure 3: KKT residual, number of gradient and Jacobian evaluations, and number of objective and constraints evaluations boxplots for Algorithm 1 and Algorithm 3 on the LIBSVM datasets.