# PREDICTION-CORRECTION FOR NONSMOOTH TIME-VARYING OPTIMIZATION VIA FORWARD-BACKWARD ENVELOPES

*Nicola Bastianello*[*], *Andrea Simonetto*[§], *Ruggero Carli*[*]

[*]Department of Information Engineering, University of Padova
[§]IBM Research Ireland

## ABSTRACT

We present an algorithm for minimizing the sum of a strongly convex time-varying function with a time-invariant, convex, and nonsmooth function. The proposed algorithm employs the prediction-correction scheme alongside the forward-backward envelope, and we are able to prove the convergence of the solutions to a neighborhood of the optimizer that depends on the sampling time. Numerical simulations for a time-varying regression problem with elastic net regularization highlight the effectiveness of the algorithm.

***Index Terms***— time-varying optimization, prediction-correction methods, forward-backward envelope, convex optimization

## 1. INTRODUCTION

In this work, we are interested in the solution of time-varying optimization problems in the form

$$\boldsymbol{x}^*(t) = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n}\{f(\boldsymbol{x};t) + g(\boldsymbol{x})\} \tag{1}$$

where $f : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ is smooth and strongly convex, and $g : \mathbb{R}^n \to \mathbb{R}$ is proper, closed and convex, but possibly non-differentiable. Since the solution $\boldsymbol{x}^*(t)$ – the trajectory – changes over time, the objective is to track it up to a bounded error ball.

In particular, we are interested in solving problem (1) in a discrete-time framework, in order to directly implement the solution on digital hardware. Hence we discretize the problem with a sampling period $T_{\mathrm{s}} := t_{k+1} - t_k$, which yields the sequence of time-invariant problems

$$\boldsymbol{x}^*(t_{k+1}) = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n}\{f(\boldsymbol{x};t_{k+1}) + g(\boldsymbol{x})\}, \quad k \in \mathbb{N}. \tag{2}$$

The smaller the sampling time is, the higher the accuracy of the trajectory composed of the solutions to (2) will be. However, we need to account for the time required to solve the problems which might exceed some values of $T_{\mathrm{s}}$, and therefore there is a trade-off between precision and practical implementation constraints.

There are many applications in which problems in the form (1) arise. For instance, in signal processing the reconstruction of time-varying signals on the basis of (noisy) observations gathered online can be cast as a sequence of optimization problems [1–6]. In control, the *model predictive control* (MPC) requires that we solve an optimization problem which varies over time [7–9] in order to design a control action. In robotics, path tracking and leader following problems can be cast in the framework of (1), see for example [10–12].

In this paper, we are interested in the solution of (2) using a *prediction-correction* scheme. Time-varying optimization algorithms based on the prediction-correction scheme have been proposed for both the discrete-time framework that we employ [13–15] and in a continuous-time setup [16–18].

These works, however, are designed to solve smooth optimization problems only; here, our aim is to tackle non-smooth optimization problems by employing the recent results on *envelope functions*, and in particular the *forward-backward envelope* (FBE) first introduced in [19], in conjunction with the prediction-correction scheme.

The main contributions of this paper are: *(i)* a prediction-correction algorithm to solve the time-varying optimization problem (1) by using the envelope functions in both the prediction and correction step; *(ii)* a detailed convergence and convergence rate analysis of the above that show global convergence to an error bound of $O(T_{\mathrm{s}})$ and local convergence to an error bound of $O(T_{\mathrm{s}}^2)$, under additional assumptions.

**Remark 1** *The forward-backward envelope is a powerful tool that has recently gained momentum, especially in the context of solving certain classes on non-convex optimization problems. In this sense, this paper can be seen as a first step towards a more general theory of time-varying optimization algorithms. We remark also that the FBE has been advocated as a way to derive Newton-like methods for $\ell_1$ regularized problems, showing improved results in comparison to more traditional approaches, such as FISTA [6], at lower computational cost. In Sec. 2, we will report some results about the FBE, and we refer the reader to [20–22] for an in-depth treatment of the subject.*

**Remark 2** *For the relationship of the FBE with the forward-backward splitting (also known as proximal gradient method) [23, 24], see e.g. [20].*

**Organization** The paper is organized as follows. Sec. 2 introduces the prediction-correction scheme, the forward-backward envelope, and then the proposed algorithm. Sec. 3 presents the convergence results for the algorithm and a sketch of the proof. Sec. 4 describes the results of the numerical simulations and Sec. 5 some concluding remarks.

**Basic definitions** We say that a function $\varphi : \mathbb{R}^n \to \mathbb{R}$ is *m-strongly convex* for a constant $m \in \mathbb{R}_+$ iff $\varphi(\boldsymbol{x}) - \frac{m}{2}\|\boldsymbol{x}\|^2$ is convex. The function $\varphi$ is said to be *L-smooth* if its gradient is $L$-Lipschitz continuous, or equivalently $\varphi(\boldsymbol{x}) - \frac{L}{2}\|\boldsymbol{x}\|^2$ is concave. We denote the class of $m$-strongly convex and $L$-smooth functions with $\mathcal{S}_{m,L}(\mathbb{R}^n)$.

A function is said to be *closed* if for any $a \in \mathbb{R}$ the set $\{\boldsymbol{x} \in \mathrm{dom}(f) \mid \varphi(\boldsymbol{x}) \leqslant a\}$ is closed. A function is said to be *proper* if it does not attain $-\infty$. We denote the class of closed, convex and proper functions with $\Gamma_0(\mathbb{R}^n)$.

Given $\varphi \in \Gamma_0(\mathbb{R}^n)$ we define its *subdifferential* as the set-valued operator $\partial\varphi : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ such that

$$\boldsymbol{x} \mapsto \{\boldsymbol{z} \in \mathbb{R}^n \mid \forall\boldsymbol{y}\in\mathbb{R}^n : \langle\boldsymbol{y} - \boldsymbol{x}, \boldsymbol{z}\rangle + \varphi(\boldsymbol{x}) \leqslant \varphi(\boldsymbol{y})\}.$$

## 2. PREDICTION-CORRECTION WITH ENVELOPES

In this section, we introduce the prediction-correction scheme for time-varying optimization alongside with the forward-backward envelope function. In the remainder of this paper we make use of the following assumptions.

**Assumption 1** *The function $f : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$ belongs to $\mathcal{S}_{m,L}(\mathbb{R}^n)$ uniformly in time. The function $g : \mathbb{R}^n \to \mathbb{R}$ belongs to $\Gamma_0(\mathbb{R}^n)$ and is in general nonsmooth.*

**Assumption 2** *The function $f$ has bounded time derivative of its gradient derivative as: $\|\nabla_{tx} f(\boldsymbol{x}; t)\| \leqslant C_0$.*

**Assumption 3** *The function $f$ is at least three time differentiable and has bounded derivatives w.r.t. $\boldsymbol{x} \in \mathbb{R}^n$ and $t \in \mathbb{R}_+$ as:*

$$\|\nabla_{\boldsymbol{xxx}} f(\boldsymbol{x}; t)\| \leqslant C_1, \quad \|\nabla_{\boldsymbol{x}t\boldsymbol{x}} f(\boldsymbol{x}; t)\| \leqslant C_2,$$
$$\|\nabla_{tt\boldsymbol{x}} f(\boldsymbol{x}; t)\| \leqslant C_3.$$

In the analysis of time-varying problems, Assumption 1 is common, see *e.g.* [13,25,26]. This assumption ensures by strong convexity that the solution to the problem is unique at each time, and that the gradient of $f$ is Lipschitz continuous. Moreover, Assumption 2 guarantees that the gradient of $f$ has a variability over time that is bounded, thus enabling the computation of reliable predictions. Assumption 3 imposes instead boundedness of the tensor $\nabla_{\boldsymbol{xxx}} f(\boldsymbol{x}; t)$, which is typical when analyzing the convergence of second-order algorithms. Moreover, it bounds the variability of the Hessian of $f$ over time, which makes it possible to carry out even more precise predictions of the optimal trajectory.

### 2.1. Prediction-correction

Prediction-correction algorithms have appeared as a computational-light way to solve time-varying optimization problems. The main idea is to compute approximate optimizers for the sequence of time-invariant problems (2), such that eventually one converges on the time-varying optimizer trajectory $\boldsymbol{x}^*(t)$. More formally, let $\boldsymbol{x}_k$ be the approximate optimizer for (2) at $k$. Then we want to design methods to determine the sequence $\{\boldsymbol{x}_k\}_{k \in \mathbb{N}}$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}^*(t_k)\|$ goes eventually to a bounded error term.

Prediction-correction algorithms determine each $\boldsymbol{x}_{k+1}$ by first predicting (at $t_k$) how the optimizer will change in time, and then by correcting (at $t_{k+1}$) based on the new acquired sampled cost function. Both prediction and correction are here based on a few descent iterations on the envelope functions. The more iterations one performs, the smaller the asymptotical tracking error, however the greater the computational time is.

The *prediction* step has the aim of computing an approximation of the optimal solution at time $t_{k+1}$, $\boldsymbol{x}^*(t_{k+1}) =: \boldsymbol{x}_{k+1}^*$, by using only the information available at time $t_k$, that is $f(\boldsymbol{x}; t_k)$ and the previous solution $\boldsymbol{x}_k$ computed by the algorithm. Once the new cost function $f(\boldsymbol{x}; t_{k+1})$ is observed at time $t_{k+1}$, we perform the *correction* step, that is we solve problem (2) approximately, using as initial condition the prediction computed at time $t_k$.

In order to use the forward-backward envelope framework, it is useful to reformulate the minimization problem (2) as the following generalized equation

$$\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}; t_{k+1}) + \partial g(\boldsymbol{x}_{k+1}) \ni 0. \tag{3}$$

During the prediction step at time $t_k$, we cannot solve (3) to predict how the optimizer will change at $t_{k+1}$; instead, we make use

of the available information at time $t_k$ to approximate $\nabla_{\boldsymbol{x}} f(\boldsymbol{x}; t_{k+1})$ with the following Taylor expansion

$$\nabla h_k(\boldsymbol{x}) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k) + \\ + \nabla_{\boldsymbol{xx}} f(\boldsymbol{x}_k; t_k)(\boldsymbol{x} - \boldsymbol{x}_k) + T_s \nabla_{t\boldsymbol{x}} f(\boldsymbol{x}_k; t_k). \tag{4}$$

Therefore during the prediction step we want to solve the approximated generalized equation

$$\nabla h_k(\boldsymbol{x}_{k+1|k}) + \partial g(\boldsymbol{x}_{k+1|k}) \ni 0 \tag{5}$$

derived from (3) substituting $h_k(\boldsymbol{x})$ to the (as yet unknown) $f(\boldsymbol{x}; t_{k+1})$; notice that $\boldsymbol{x}_{k+1|k}$ will denote the computed prediction.

During the correction step at time $t_{k+1}$, we can now solve (approximately) (3), which is what we will do.

**Remark 3** *From Assumption 1 follows that $h_k \in \mathcal{S}_{m,L}(\mathbb{R}^n)$, and by definition we can write it explicitly as*

$$h_k(\boldsymbol{x}) = \frac{1}{2} \boldsymbol{x}^\top \nabla_{\boldsymbol{xx}} f(\boldsymbol{x}_k; t_k) \boldsymbol{x} + \\ + \left( \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k) - \nabla_{\boldsymbol{xx}} f(\boldsymbol{x}_k; t_k) \boldsymbol{x}_k + T_s \nabla_{t\boldsymbol{x}} f(\boldsymbol{x}_k; t_k) \right)^\top \boldsymbol{x}.$$

### 2.2. Forward-backward envelope

Notice that both the prediction and correction problems, (5) and (3), are of the form

$$\nabla \varphi(\boldsymbol{x}^*) + \partial g(\boldsymbol{x}^*) \ni 0 \tag{6}$$

with $\varphi$ that is $m$-strongly convex and $L$-smooth. Therefore we can apply the recently proposed *forward-backward envelope* (FBE) to solve them.

The FBE for a problem (6) is defined as

$$\mathrm{M}(\boldsymbol{x}) = \min_{\boldsymbol{y}} \left\{ \varphi(\boldsymbol{x}) + \langle \nabla \varphi(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + g(\boldsymbol{y}) + \frac{\|\boldsymbol{y} - \boldsymbol{x}\|^2}{2\gamma} \right\} \tag{7}$$

where $\gamma \in (0, 1/L)$.

Under Assumption 1 it holds that

$$\arg\min(\varphi + g)(\boldsymbol{x}) = \arg\min \mathrm{M}(\boldsymbol{x}),$$

and therefore minimizing the FBE is equivalent to solving problem (6). Moreover, the envelope is continuously differentiable on $\mathbb{R}^n$ and twice continuously differentiable at the unique solution $\boldsymbol{x}^*$, with positive definite Hessian.

In general, the FBE is however nonconvex, and hence in order to minimize it a *quasi-Newton* scheme with line search has been proposed in [20], that estimates the Hessian of the FBE using the BFGS method. In our framework, it is possible to prove that the quasi-Newton method applied to the FBE has global linear convergence, that is

$$\left\| \boldsymbol{x}^{l+1} - \boldsymbol{x}^* \right\| \leqslant \zeta \left\| \boldsymbol{x}^l - \boldsymbol{x}^* \right\|, \quad l \in \mathbb{N}$$

with

$$\zeta = \sqrt{\max \left\{ \frac{1}{2}, 1 - \frac{m}{4} \min \left\{ \gamma, \frac{1}{4L} \right\} \right\}} < 1, \tag{8}$$

a result that will be instrumental in proving convergence of our prediction-correction algorithm.

**Remark 4** *The recent work [22] proved that if $\varphi$ is convex quadratic, then the FBE is strongly convex and smooth, and notice that this is exactly the case of $h_k$ in the prediction step. Therefore we can minimize the FBE at the prediction step using a Newton method with BFGS scheme, without the need for the line search that requires a larger number of iterations. The numerical results presented in Sec. 4 exploit this.*

**Remark 5** *An alternative minimization strategy for the FBE is proposed in [21].*

### 2.3. Proposed algorithm

The previous section introduced the forward-backward envelope, that is suited to solving the prediction and correction problems. However, the convergence of the quasi-Newton method is guaranteed only asymptotically. For practical reasons, namely the finite length of each sampling period, we choose to perform only a fixed number of iterations of the solution algorithm: $P$ for the prediction step, $C$ for the correction.

We are now ready to describe the proposed prediction-correction algorithm with the FBE, which is reported in Algorithm 1: at every time $t_k$, we perform $P$ steps of the quasi-Newton method for the FBE,

$$\tilde{\mathrm{M}}(\boldsymbol{x}) = \min_{\boldsymbol{y}} \left\{ h_k(\boldsymbol{x}) + \langle \nabla h_k(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + g(\boldsymbol{y}) + \frac{\|\boldsymbol{y} - \boldsymbol{x}\|^2}{2\gamma} \right\},$$

constructed for the prediction problem (5) [cf. line 3]; this yields an approximate predictor $\tilde{\boldsymbol{x}}_{k+1|k}$.

At time $t_{k+1}$, we observe the new cost function $f(\cdot; t_{k+1})$ [cf. line 6], and we perform $C$ steps of the quasi-Newton method for the FBE,

$$\mathrm{M}(\boldsymbol{x}) = \min_{\boldsymbol{y}} \left\{ f(\boldsymbol{x}; t_{k+1}) + \langle \nabla_{\boldsymbol{x}} f(\boldsymbol{x}; t_{k+1}), \boldsymbol{y} - \boldsymbol{x} \rangle + \right.$$
$$\left. + g(\boldsymbol{y}) + \frac{\|\boldsymbol{y} - \boldsymbol{x}\|^2}{2\gamma} \right\},$$

constructed for the correction problem [cf. line 7]; this yields the approximate optimizer $\boldsymbol{x}_{k+1}$.

---

**Algorithm 1** Prediction-correction algorithm with the FBE.

---

**Input:** $\boldsymbol{x}_0$, parameter $\gamma$, horizons $P$ and $C$.
1: **for** $k = 0, 1, \ldots$ **do**
2:      // time $t_k$
3:      perform $P$ steps of the quasi-Newton method for the FBE with initial condition $\boldsymbol{x}_k$
4:      set $\tilde{\boldsymbol{x}}_{k+1|k}$ equal to the last iterate produced by the quasi-Newton
5:      // time $t_{k+1}$
6:      observe the cost function $f(\cdot; t_{k+1})$
7:      perform $C$ steps of the quasi-Newton method for the FBE with initial condition the prediction $\tilde{\boldsymbol{x}}_{k+1|k}$
8:      set $\boldsymbol{x}_{k+1}$ equal to the last iterate produced by the quasi-Newton
9: **end for**

---

**Remark 6** *In general we could use two different $\gamma$ parameters for the prediction and correction steps, but for simplicity we use a single one.*

## 3. CONVERGENCE ANALYSIS

In this section, we prove that the sequence $\{\boldsymbol{x}_k\}_{k \in \mathbb{N}}$ generated by Algorithm 1 converges to a neighborhood of the optimal trajectory, which is characterized in terms of the sampling period $T_{\mathrm{s}}$. We divide the result in two theorems. The first is a global convergence result with standard assumptions; the second is a local enhanced convergence result with additional assumptions. Such results are typical in prediction-correction time-varying optimization and they extend the ones in [14] for non-smooth cost functions.

**Theorem 1** *Let Assumptions 1-2 hold, and choose the parameters $P$ and $C$ in such a way that*

$$\zeta^C \left[ \zeta^P + (\zeta^P + 1) \frac{2L}{m} \frac{1 - \gamma m}{1 - \gamma L} \right] < 1.$$

*Then the trajectory $\{\boldsymbol{x}_k\}_{k \in \mathbb{N}}$ generated by Algorithm 1 converges to a neighborhood of the optimal trajectory $\{\boldsymbol{x}_k^*\}_{k \in \mathbb{N}}$ as*

$$\limsup_{k \to \infty} \|\boldsymbol{x}_k - \boldsymbol{x}_k^*\| = O(\zeta^C T_{\mathrm{s}}).$$

        □

**Theorem 2** *Let Assumptions 1, 2, and 3 hold, and choose the parameters $P$ and $C$, and $\tau \in (0, 1)$ in such a way that $\zeta^{P+C} < \tau$. Then there exist an upper bound for the sampling time $\bar{T}_{\mathrm{s}}$ and a convergence region $\bar{R}$ such that if $T_{\mathrm{s}} < \bar{T}_{\mathrm{s}}$ and $\|\boldsymbol{x}_0 - \boldsymbol{x}_0^*\| < \bar{R}$, then*

$$\limsup_{k \to \infty} \|\boldsymbol{x}_k - \boldsymbol{x}_k^*\| = O(\zeta^C T_{\mathrm{s}}^2) + O(\zeta^{P+C} T_{\mathrm{s}}).$$

*In particular, the bound for the sampling time and the convergence region are characterized by*

$$\bar{T}_{\mathrm{s}} = \frac{\tau - \zeta^{P+C}}{\zeta^C(\zeta^P + 1)} \frac{1}{\kappa(\kappa C_0 C_1 + C_2)}$$

$$\bar{R} = \frac{2}{C_1}(\kappa C_0 C_1 + C_2) \left( \bar{T}_{\mathrm{s}} - \frac{m}{\zeta^C} T_{\mathrm{s}} \right).$$

*with $\kappa = (1 - \gamma m)/[m(1 - \gamma L)]$.*     □

    The two theorems guarantee that, under suitable regularity conditions of the problem in hand, the trajectory generated by Algorithm 1 converges asymptotically to a neighborhood of the optimal trajectory. Moreover, the size of this neighborhood depends on $T_{\mathrm{s}}$ for Theorem 1 and on $T_{\mathrm{s}}^2$ for Theorem 2, in accordance with the fact that the smaller the sampling time is, the better the sequence of problems (2) approximates the original problem (1).

    The neighborhoods depend also on the convergence rate $\zeta$, which in turn depends on the convexity and smoothness moduli of the function $f$; thus the structure of the problem influences the accuracy of the proposed algorithm.

    The proof of both results can be found in the Appendix, along with the exact expression for the asymptotic error. Here we mention only some facts. The idea behind the proof is to compute an upper bound to the error $\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\|$, and to do so we need to account for two sources of error: the approximation error introduced during the prediction step, and the early termination error due to the finite number of minimization steps in both prediction and correction. The approximation error depends (among other things) on how fast the cost function is changing and a bound on such error can be derived

based on implicit function mapping theorems. In particular, we make use of Dini's theorem, see *e.g.*, [27, Th. 1B.1], and the algebraic properties of the envelope function. The early termination errors are instead bounded based solely on the properties of the envelope function.

Once the bound for the errors is derived, we combine them and provide a bound for the error $\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\|$ based on the parameters of Algorithm 1 (*i.e.*, the step-size $\gamma$, and the horizons $P$ and $C$). Then, we choose such parameters in order to guarantee a finite error. The error bound available is in general not tight, and therefore it might be possible to relax the conditions on the parameters while still ensuring the convergence; which we will explore in future research.

## 4. SIMULATIONS

Inspired by [6], and only as a proof of concept of our algorithm, we consider a regression problem, where we are interested in reconstructing a sparse time-varying signal $\boldsymbol{y}_k$ from the noisy measurements $\mathbf{b}_k = \mathbf{A}\boldsymbol{y}_k + \mathbf{e}_k$ where the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, the measurement vector $\mathbf{b}_k \in \mathbb{R}^m$ with $m < n$ – in particular $m = 25$ and $n = 50$ – and the components of the error vector $\mathbf{e}_k$ are drawn from the normal distribution $\mathcal{N}(0, 10^{-3})$. We apply an *elastic net* to solve the problem, *i.e.*, we define $g(\boldsymbol{x}) = \alpha \|\boldsymbol{x}\|_1$ and $f(\boldsymbol{x}; t_k) = (1/2)\|\mathbf{A}\boldsymbol{x} - \mathbf{b}_k\|_2^2 + (1-\alpha)\|\boldsymbol{x}\|_2^2/2$ for $\alpha \in [0, 1]$, and we formulate the sequence of time-invariant problems

$$\boldsymbol{x}^*(t_k) = \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{ \frac{1}{2}\|\mathbf{A}\boldsymbol{x} - \mathbf{b}_k\|_2^2 + \frac{(1-\alpha)}{2}\|\boldsymbol{x}\|_2^2 + \alpha\|\boldsymbol{x}\|_1 \right\}.$$
(9)

Each component of the signal to be reconstructed is either of the form $y_k^i = c\sin(\omega t_k + \phi^i)$ where $c, \phi^i$ are random, or it is 0; $\omega$ is set as $1/20$, so that we do half-a-turn every minute. The number and index of the "active" components is fixed at 6.

In Fig. 1(a) we present the evolution of the error

$$E_{\mathrm{r}} = \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\|/6,$$

(the error divided by the number of non-zero components), labeled as "Tracking error", for different values of the prediction horizon $P$, and $\alpha = 0.8$, obtained with $T_{\mathrm{s}} = 0.1\mathrm{s}$, $C = 5$, $\gamma = 0.8/L$ and minimizing the FBE with the line-search quasi-Newton. Notice that a larger number of prediction steps yields a faster convergence rate and a lower error, which justifies the use of the prediction-correction scheme. Indeed in case we perform only a correction ($P = 0$) we obtain the worst performance, which means that the ability to predict the future solution enhances the performance of the optimization algorithm. Note that, even with $P = 5$, the performance is better. The quasi-cyclic nature of the error is due to the sinusoidal reference signal.

As noted in Remark 4, a convex quadratic cost function $f$ guarantees that the FBE is strongly convex. In this scenario, we can compute a descent direction for the FBE without the need for the line-search procedure included in the quasi-Newton algorithm of [20]; moreover, the quasi-Newton algorithms with and without line-search yield the same results. We can also think of applying the gradient method, which has good convergence properties for strongly and well-conditioned convex functions and which does not require the computation of the Hessian of the FBE.
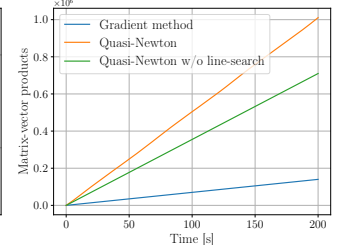
Fig. 1(b) depicts the evolution of the error for the quasi-Newton and the gradient methods (with the parameter $P = 10, C = 5$), while Fig. 1(c) the number of matrix-vector products required by the quasi-Newton with and without line-search, and the gradient.
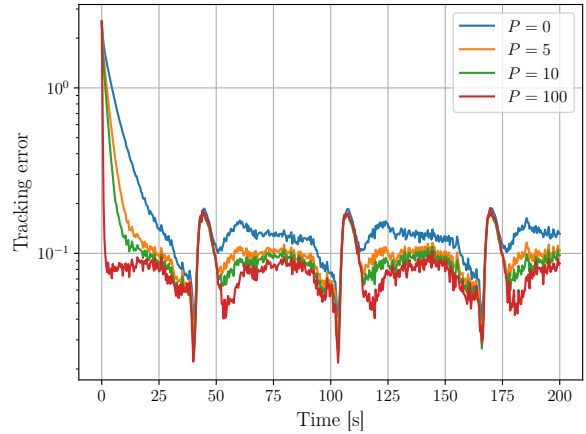


(a) Tracking error for different prediction horizons.
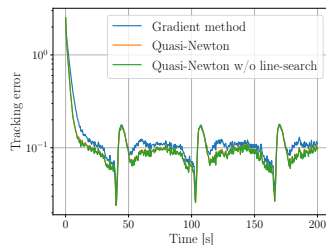


(b) Gradient vs quasi-Newton.  (c) Matrix-vector products.

**Fig. 1:** Experimental results.

Therefore we can choose between a more computationally demanding quasi-Newton, or a gradient method that is simpler to implement but obtains results close to the former.

## 5. CONCLUSION

In this paper, we presented a prediction-correction scheme for time-varying optimization that employs the forward-backward envelope. We described two Theorems that guarantee the convergence of the solution computed by the algorithm to a neighborhood of the optimal solution that depends on the sampling time. Finally we validated the proposed algorithm with some numerical results. Future works will address the problem of relaxing the requirements on the cost functions, especially strong convexity, and perhaps convexity altogether, together with extensive numerical validation and comparison with state-of-the-art methods in specific applications, *e.g.*, dynamic $\ell_1$ reconstruction.

## 6. REFERENCES

[1] M Salman Asif and Justin Romberg, "Dynamic updating for $\ell_1$ minimization," *IEEE Journal of selected topics in signal processing*, vol. 4, no. 2, pp. 421–434, 2010.

[2] M Salman Asif and Justin Romberg, "Sparse recovery of streaming signals using $\ell_1$-homotopy," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4209–4223, 2014.

[3] N. Vaswani and J. Zhan, "Recursive recovery of sparse signal sequences from compressive measurements: A review," *IEEE Transactions on Signal Processing*, vol. 64, no. 13, pp. 3523 – 3549, 2016.

[4] Y. Yang, M. Zhang, M. Pesavento, and D. P. Palomar, "An online parallel and distributed algorithm for recursive estimation of sparse signals," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 3, pp. 290 – 305, 2016.

[5] Adam S Charles, Aurele Balavoine, and Christopher J Rozell, "Dynamic filtering of time-varying sparse signals via $\ell_1$ minimization," *IEEE Transactions on Signal Processing*, vol. 64, no. 21, pp. 5644–5656, 2016.

[6] P. Sopasakis, N. Freris, and P. Patrinos, "Accelerated reconstruction of a compressively sampled data stream," in *Proceedings of the 24th EUSIPCO*, Budapest, Hungary, September 2016, pp. 1078 – 1082.

[7] Juan L Jerez, Paul J Goulart, Stefan Richter, George A Constantinides, Eric C Kerrigan, and Manfred Morari, "Embedded online optimization for model predictive control at megahertz rates," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3238–3251, 2014.

[8] Jean-Hubert Hours and Colin N Jones, "A parametric nonconvex decomposition algorithm for real-time and distributed NMPC," *IEEE Transactions on Automatic Control*, vol. 61, no. 2, pp. 287–302, 2016.

[9] Benjamin Gutjahr, Lutz Gröll, and Moritz Werling, "Lateral vehicle trajectory optimization using constrained linear time-varying MPC," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1586–1595, 2017.

[10] Diederik Verscheure, Bram Demeulenaere, Jan Swevers, Joris De Schutter, and Moritz Diehl, "Time-optimal path tracking for robots: A convex optimization approach," *IEEE Transactions on Automatic Control*, vol. 54, no. 10, pp. 2318–2327, 2009.

[11] Tohid Ardeshiri, Mikael Norrlöf, Johan Löfberg, and Anders Hansson, "Convex optimization approach for time-optimal path tracking of robots with speed dependent constraints," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 14648–14653, 2011.

[12] Rishabh Dixit, Amrit Singh Bedi, Ruchi Tripathi, and Ketan Rajawat, "Online learning with inexact proximal online gradient descent algorithms," *arXiv preprint arXiv:1806.00202*, 2018.

[13] Andrea Simonetto, Aryan Mokhtari, Alec Koppel, Geert Leus, and Alejandro Ribeiro, "A class of prediction-correction methods for time-varying convex optimization.," *IEEE Trans. Signal Processing*, vol. 64, no. 17, pp. 4576–4591, 2016.

[14] Andrea Simonetto and Emiliano Dall'Anese, "Prediction-correction algorithms for time-varying constrained optimization," *IEEE Transactions on Signal Processing*, vol. 65, no. 20, pp. 5481–5494, 2017.

[15] A. Simonetto, "Dual prediction-correction methods for linearly constrained time-varying convex programs," *IEEE Transactions on Automatic Control (to appear)*, 2018.

[16] Mahyar Fazlyab, Santiago Paternain, Victor M. Preciado, and Alejandro Ribeiro, "Prediction-correction interior-point method for time-varying convex optimization," *IEEE Transactions on Automatic Control*, 2017.

[17] S. Rahili and W. Ren, "Distributed convex optimization for continuous-time dynamics with time-varying cost functions," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1590 – 1605, 2017.

[18] Mahyar Fazlyab, Cameron Nowzari, George J. Pappas, Alejandro Ribeiro, and Victor M. Preciado, "Self-triggered time-varying convex optimization," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 3090–3097.

[19] Panagiotis Patrinos and Alberto Bemporad, "Proximal newton methods for convex composite optimization," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*. IEEE, 2013, pp. 2358–2363.

[20] Lorenzo Stella, Andreas Themelis, and Panagiotis Patrinos, "Forward–backward quasi-newton methods for nonsmooth optimization problems," *Computational Optimization and Applications*, vol. 67, no. 3, pp. 443–487, 2017.

[21] Andreas Themelis, Lorenzo Stella, and Panagiotis Patrinos, "Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone linesearch algorithms," *SIAM Journal on Optimization*, vol. 28, no. 3, pp. 2274–2303, 2018.

[22] Pontus Giselsson and Mattias Fält, "Envelope functions: Unifications and further properties," *Journal of Optimization Theory and Applications*, vol. 178, no. 3, pp. 673–698, 2018.

[23] Neal Parikh and Stephen Boyd, "Proximal Algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[24] Patrick L. Combettes and Jean-Christophe Pesquet, "Proximal Splitting Methods in Signal Processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, vol. 49, pp. 185–212. Springer New York, New York, NY, 2011.

[25] A Yu Popkov, "Gradient methods for nonstationary unconstrained optimization problems," *Automation and Remote Control*, vol. 66, no. 6, pp. 883–891, 2005.

[26] Asen L Dontchev, MI Krastanov, R Tyrrell Rockafellar, and Vladimir M Veliov, "An Euler–Newton continuation method for tracking solution trajectories of parametric variational inequalities," *SIAM Journal on Control and Optimization*, vol. 51, no. 3, pp. 1823–1840, 2013.

[27] Asen L Dontchev and R Tyrrell Rockafellar, *Implicit Functions and Solution Mappings: A View from Variational Analysis*, Springer, 2014.

[28] Carl D. Meyer, *Matrix Analysis and Applied Linear Algebra*, vol. 71, SIAM, 2000.

## A. PROOFS

The sources of error for the proposed algorithm are the Taylor approximation in the prediction step, and the early termination of the quasi-Newton minimization of the FBE.

### A.1. Approximation error

Let $\tilde{M}(\boldsymbol{x})$ be the FBE computed for $h_k$ and $g$, explicitly defined as

$$\tilde{M}(\boldsymbol{x}) = \min_{\boldsymbol{y}} \left\{ h_k(\boldsymbol{x}) + \langle \nabla h_k(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle + g(\boldsymbol{y}) + \frac{\|\boldsymbol{y} - \boldsymbol{x}\|^2}{2\gamma} \right\},$$

and thus that solves the prediction problem $(5)^1$.
Minimizing this FBE requires that we solve the generalized equation $\nabla \tilde{M}(\boldsymbol{x}) \ni 0$, and we consider its parametrized version, defined as

$$\nabla \tilde{M}(\boldsymbol{x}) \ni \boldsymbol{p}, \tag{10}$$

where $\boldsymbol{p} \in \mathbb{R}^n$.
Therefore the problem becomes that of finding the solution mapping

$$S(\boldsymbol{p}) = \{\boldsymbol{x} \in \mathbb{R}^n \mid F(\boldsymbol{p}, \boldsymbol{x}) = 0\}, \ \boldsymbol{p} \in \mathbb{R}^d$$

where $F : \mathbb{R}^d \times \mathbb{R}^n \to \mathbb{R}^n$ is defined as $F(\boldsymbol{p}, \boldsymbol{x}) = \nabla \tilde{M}(\boldsymbol{x}) - \boldsymbol{p}$. Reformulating the problem in this fashion we can make use of Dini's theorem [27, Th. 1B.1], reported here for convenience.

**Theorem 3 (Dini [27, Th. 1B.1])** *Let $F : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ be continuously differentiable in a neighborhood of $(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}})$ and such that $F(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}}) = 0$ and $\nabla_{\boldsymbol{x}} F(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}})$ is nonsingular. Then the solution mapping $S(\boldsymbol{p})$ has a single-valued localization $s$ around $\bar{\boldsymbol{p}}$ for $\bar{\boldsymbol{x}}$ which is continuously differentiable in a neighborhood of $\bar{\boldsymbol{p}}$ with Jacobian satisfying*

$$\nabla s(\boldsymbol{p}) = -\nabla_{\boldsymbol{x}} F(\boldsymbol{p}, s(\boldsymbol{p}))^{-1} \nabla_p F(\boldsymbol{p}, s(\boldsymbol{p})).$$

□

Assume now that $\bar{\boldsymbol{x}}$ is a solution to (10) with $\bar{\boldsymbol{p}}$ such that $s(\bar{\boldsymbol{p}}) = \boldsymbol{x}$. Then Theorem 3 holds for $F(\boldsymbol{p}, \boldsymbol{x}) = \nabla \tilde{M}(\boldsymbol{x}) - \boldsymbol{p}$ in a neighborhood of $(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}})$ if $F$ is continuously differentiable and has nonsingular Jacobian. These conditions are analyzed in the following:

- $F$ is differentiable in $\boldsymbol{p}$ everywhere, while to be differentiable in $\boldsymbol{x}$ it must be that $\nabla_{\boldsymbol{x}} \tilde{M}$ is differentiable, which is guaranteed in a neighborhood of $\bar{\boldsymbol{x}}$ by [21, Theorem 4.7] under Assumption 1.

- By [20, Theorem 2.11] the Hessian of the FBE in a *strong local minimum*$^2$ is positive definite, therefore $\nabla_{\boldsymbol{x}} F(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}}) = \nabla^2 \tilde{M}(\bar{\boldsymbol{x}})$ is nonsingular if $\bar{\boldsymbol{x}}$ is a strong local minimum.

Hence as long as $\bar{\boldsymbol{x}}$ is a strong local minimum it is possible to prove that Theorem 3 holds for the problem at hand.

The next step is to apply Dini's theorem to provide an upper bound to the error. First notice that a function that is continuously differentiable in a point $\boldsymbol{y}$ has Lipschitz constant the norm of its

---

$^1$Notice that since $h_k$ depends only on $\boldsymbol{x}$ we omit to specify that the gradient is computed w.r.t. $\boldsymbol{x}$.

$^2$A minimum $x^*$ of function $h$ is said to be *locally strong* if there exists $\alpha > 0$ such that $h(x) - h(x^*) \geq \alpha \|x - x^*\|^2$ for any $x$ in a neighborhood of $x^*$.

gradient in $\boldsymbol{y}$ [27, p. 30]. Therefore, since $\nabla_p F(\boldsymbol{p}, s(\boldsymbol{p})) = I$, then $s(\boldsymbol{p})$ is Lipschitz continuous in a neighborhood of $\bar{\boldsymbol{p}}$ with constant

$$\|\nabla s(\bar{\boldsymbol{p}})\| = \left\| \nabla_{\boldsymbol{x}} F(\bar{\boldsymbol{p}}, \bar{\boldsymbol{x}})^{-1} \right\| = \left\| \nabla^2 \tilde{M}(\bar{\boldsymbol{x}})^{-1} \right\| =: K.$$

As mentioned at the beginning of this proof, the aim is to define an upper bound for the error introduced by the approximation of the cost function, that is $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*(t_{k+1})\|$. This is accomplished by using the Lipschitz continuity of $s$

$$\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*(t_{k+1})\| = \|s(\bar{\boldsymbol{p}}) - s(\mathbf{q})\| \leq K \|\bar{\boldsymbol{p}} - \mathbf{q}\|$$

where it is necessary to find $\mathbf{q}$ such that $s(\mathbf{q}) = \boldsymbol{x}^*(t_{k+1})$. Notice that $\boldsymbol{x}^*_{k+1} := \boldsymbol{x}^*(t_{k+1})$ must satisfy $\nabla M(\boldsymbol{x}^*_{k+1}) \ni 0$ where $M(\boldsymbol{x}^*_{k+1})$ is the forward-backward envelope defined in (7). Therefore by choosing

$$\mathbf{q} = \nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) - \nabla M(\boldsymbol{x}^*_{k+1})$$

we have $s(\mathbf{q}) = \boldsymbol{x}^*_{k+1}$, since

$$\nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) \ni \mathbf{q} = \nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) - \nabla M(x^*_{k+1})$$

is verified because by definition $\nabla M(\boldsymbol{x}^*_{k+1}) \ni 0$.
Finally it follows that

$$\left\| \bar{\boldsymbol{x}} - \boldsymbol{x}^*_{k+1} \right\| \leq K \left\| \nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) - \nabla M(\boldsymbol{x}^*_{k+1}) \right\|.$$

By [20, Theorem 2.6] it follows that

$$\nabla M(\boldsymbol{x}^*_{k+1}) = (I - \gamma \nabla_{\boldsymbol{xx}} f(\boldsymbol{x}^*_{k+1}; t_{k+1})) R_\gamma(\boldsymbol{x}^*_{k+1})$$
$$=: S_\gamma(\boldsymbol{x}^*_{k+1}) R_\gamma(\boldsymbol{x}^*_{k+1}) \tag{11}$$

where $R_\gamma(\boldsymbol{y}) = \gamma^{-1}(\boldsymbol{y} - \text{prox}_{\gamma g}(\boldsymbol{y} - \gamma \nabla_{\boldsymbol{x}} f(\boldsymbol{y}; t_{k+1})))$ is the *residual*. However, by the definition of residual it holds $R_\gamma(\boldsymbol{x}^*_{k+1}) = 0$ which actually implies that $\nabla M(\boldsymbol{x}^*_{k+1}) = 0$.
Similarly for $\tilde{M}(\boldsymbol{x})$ it holds

$$\nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) = \tilde{S}_\gamma(\boldsymbol{x}^*_{k+1}) \tilde{R}_\gamma(\boldsymbol{x}^*_{k+1})$$

with $\tilde{S}_\gamma(\boldsymbol{x}^*_{k+1})$ and $\tilde{R}_\gamma(\boldsymbol{x}^*_{k+1})$ defined substituting $h_k(\cdot)$ to $f(\cdot; t_{k+1})$ in $S_\gamma(\boldsymbol{x}^*_{k+1})$ and $R_\gamma(\boldsymbol{x}^*_{k+1})$.
Therefore the approximation error can be upper bounded as

$$\left\| \bar{\boldsymbol{x}} - \boldsymbol{x}^*_{k+1} \right\| \leq K \left\| \nabla \tilde{M}(\boldsymbol{x}^*_{k+1}) \right\|. \tag{12}$$

We now proceed to bound $K = \|\nabla^2 \tilde{M}(\bar{\boldsymbol{x}})^{-1}\|$ and $J := \|\nabla \tilde{M}(\boldsymbol{x}^*_{k+1})\|$.

First of all, using the fact that $\nabla^2 h_k(\boldsymbol{y}) = \nabla_{\boldsymbol{xx}} f(\boldsymbol{x}_k; t_k) =: \boldsymbol{Q}_k$ it follows $\tilde{S}_\gamma(\boldsymbol{x}^*_{k+1}) = I - \gamma \boldsymbol{Q}_k$ which is a positive definite matrix. Indeed observe that by Assumption 1 the maximum and minimum eigenvalues of $\boldsymbol{Q}_k$ are, respectively: $\lambda_M(\boldsymbol{Q}_k) = L$ and $\lambda_m(\boldsymbol{Q}_k) = m$, with $L \geq m > 0$. Therefore since the eigenvalues of $\tilde{S}_\gamma(\boldsymbol{x})$ are of the form $\lambda_i(\tilde{S}_\gamma(\boldsymbol{x})) = 1 - \gamma \lambda_i(\boldsymbol{Q}_k)$ it follows that

$$\lambda_M(\tilde{S}_\gamma(\boldsymbol{x})) \leq 1 - \gamma m \quad \text{and} \quad \lambda_m(\tilde{S}_\gamma(\boldsymbol{x})) \geq 1 - \gamma L, \tag{13}$$

with $1 - \gamma L > 0$ because $\gamma < 1/L$.
Furthermore it holds by [20, Theorem 2.10]

$$\nabla^2 \tilde{M}(\bar{\boldsymbol{x}}) = \gamma^{-1} \tilde{S}_\gamma(\bar{\boldsymbol{x}})(I - \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}}))$$

where $\tilde{P}_\gamma(\bar{\boldsymbol{x}})$ is semi-definite positive and $\|\tilde{P}_\gamma(\boldsymbol{x})\| \leq 1$.

*A.1.1. Computation of J*

By the formula for the gradient of $\tilde{\mathrm{M}}(\boldsymbol{x})$ it holds

$$\|\nabla\tilde{\mathrm{M}}(\boldsymbol{x}_{k+1}^*)\| \leqslant \|\tilde{S}_\gamma(\boldsymbol{x}_{k+1}^*)\|\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\| \qquad (14)$$

where by (13) it holds $\|\tilde{S}_\gamma(\boldsymbol{x}_{k+1}^*)\| \leqslant 1 - \gamma m$ . Therefore to compute $J$ it is necessary to provide an upper bound to the norm of the residual $\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\|$.

Recalling the definition of the residual and by the fact that $R_\gamma(\boldsymbol{x}_{k+1}^*) = 0$ it holds

$$\begin{aligned}
\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\| &= \|R_\gamma(\boldsymbol{x}_{k+1}^*) - \tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\| \\
&= \gamma^{-1}\| \operatorname{prox}_{\gamma g}(\boldsymbol{x}_{k+1}^* - \gamma\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1})) \\
&\qquad - \operatorname{prox}_{\gamma g}(\boldsymbol{x}_{k+1}^* - \gamma\nabla h_k(\boldsymbol{x}_{k+1}^*))\| \\
&\leqslant \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla h_k(\boldsymbol{x}_{k+1}^*))\right\|
\end{aligned}$$

where the nonexpansiveness of the proximal operator was used to derive the last inequality.

By using the definition of $h_k$, it follows

$$\begin{aligned}
&\left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla h_k(\boldsymbol{x}_{k+1}^*))\right\| \\
&\leqslant \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| + \\
&\qquad + \|\boldsymbol{Q}_k\| \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + T_{\mathrm{s}}\left\|\nabla_{t\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| \\
&\leqslant \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| + L\left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + T_{\mathrm{s}}C_0
\end{aligned}$$

where the upper bounds in Assumptions 1-2 on the derivatives of $f$ were used.

The first term on the right-hand side of the inequality remains now to be computed. It holds

$$\begin{aligned}
&\left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| \\
&\leqslant \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_{k+1})\right\| + \\
&\qquad + \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\|
\end{aligned}$$

where by Lipschitz continuity

$$\left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_{k+1})\right\| \leqslant L\left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\|$$

and $\left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_{k+1}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| \leqslant T_{\mathrm{s}}C_0$ (see [14, Appendix A]).

Finally the results above yield

$$\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\| \leqslant 2(L\left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + T_{\mathrm{s}}C_0) \qquad (15)$$

and therefore

$$\left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \leqslant 2K(1 - \gamma m)(L\left\|\boldsymbol{x}_k - \boldsymbol{x}_{k+1}^*\right\| + T_{\mathrm{s}}C_0).$$

The problem now is to bound $\left\|\boldsymbol{x}_k - \boldsymbol{x}_{k+1}^*\right\|$ in terms of $\left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\|$; first of all, it holds $\left\|\boldsymbol{x}_k - \boldsymbol{x}_{k+1}^*\right\| \leqslant \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k^*\right\| + \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\|$ and so an upper bound for $\left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k^*\right\|$ must be found.

Recall that with $\boldsymbol{x}^*(t)$ we denote the optimal solution of the original time-varying problem (1). Notice that Dini's theorem holds for the corresponding FBE at time $t$: $\nabla_{\boldsymbol{x}} \mathrm{M}(\boldsymbol{x}; t)$ with $t \in \mathbb{R}_+$ around $\boldsymbol{x}^*(t)$. Therefore the solution mapping $\boldsymbol{x}^*(t)$ has a Lipschitz constant upper bounded by

$$\left\|\nabla_{\boldsymbol{x}\boldsymbol{x}} \mathrm{M}(\boldsymbol{x}^*(t); t)^{-1}\right\| \left\|\nabla_{t\boldsymbol{x}} \mathrm{M}(\boldsymbol{x}^*(t); t)\right\| \\ =: \bar{K} \left\|\nabla_{t\boldsymbol{x}} \mathrm{M}(\boldsymbol{x}^*(t); t)\right\|.$$

The term $\bar{K}$ will be computed in the next section alongside $K$, while the second term requires evaluating the time derivative of the gradient of the FBE. In particular, by derivative rules, it holds that

$$\nabla_{t\boldsymbol{x}} \mathrm{M}(\boldsymbol{x}; t) = \nabla_t S_\gamma(\boldsymbol{x}; t)R_\gamma(\boldsymbol{x}; t) + S_\gamma(\boldsymbol{x}; t)\nabla_t R_\gamma(\boldsymbol{x}; t)$$

where the first term on the right-hand side can be ignored, since at $(\boldsymbol{x}^*(t); t)$ it is zero. By using the definition (11), it holds that $S_\gamma(\boldsymbol{x}; t) = I - \gamma\nabla_{\boldsymbol{x}\boldsymbol{x}} f(\boldsymbol{x}; t)$, whose norm is upper bounded by $1 - \gamma m$; hence it remains only to compute an upper bound to the norm of $\nabla_t R_\gamma(\boldsymbol{x}; t)$.

By computing the time derivative of the residual, it follows that

$$\begin{aligned}
\nabla_t R_\gamma(\boldsymbol{x}; t) &= -\gamma^{-1}\nabla_t[\operatorname{prox}_{\gamma g}(\boldsymbol{x} - \gamma\nabla_{\boldsymbol{x}} f(\boldsymbol{x}; t))] \\
&= \nabla_{t\boldsymbol{x}} f(\boldsymbol{x}; t)J \operatorname{prox}_{\gamma g}(\boldsymbol{x} - \gamma\nabla_{\boldsymbol{x}} f(\boldsymbol{x}; t)) \\
&= \nabla_{t\boldsymbol{x}} f(\boldsymbol{x}; t)P_\gamma(\boldsymbol{x}; t)
\end{aligned}$$

where the second inequality is derived using the chain rule, and the third by using the definition of the semi-definite positive matrix $P_\gamma(\boldsymbol{x}; t)$ reported in [20].

Therefore it holds that

$$\left\|\nabla_t R_\gamma(\boldsymbol{x}^*(t); t)\right\| \leqslant \left\|\nabla_{t\boldsymbol{x}} f(\boldsymbol{x}^*(t); t)\right\| \left\|P_\gamma(\boldsymbol{x}^*(t); t)\right\| \leqslant C_0$$

since $\|P_\gamma(\boldsymbol{x}^*(t); t)\| \leqslant 1$.

Finally, the solution mapping $\boldsymbol{x}^*(t)$ has Lipschitz constant $\bar{K}(1 - \gamma m)C_0$ and so

$$\begin{aligned}
\left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k^*\right\| &= \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k^*\right\| \qquad (16) \\
&\leqslant \bar{K}(1 - \gamma m)C_0|t_{k+1} - t_k| = \bar{K}(1 - \gamma m)C_0 T_{\mathrm{s}}.
\end{aligned}$$

*A.1.2. Computation of K*

Let $A, B \in \mathbb{R}^{n\times n}$ with $A$ nonsingular, and denote with $\sigma_m(\cdot)$ and $\sigma_M(\cdot)$ the minimum and maximum singular values of a matrix, respectively. The following facts hold true for symmetric nonsingular matrices [28, Ch. 5].

1. $\left\|A^{-1}\right\| = \sigma_M\left(A^{-1}\right) = 1/\sigma_m(A)$;
2. $\sigma_m(AB) \geqslant \sigma_m(A)\sigma_m(B)$.

Therefore it follows

$$K = \|\nabla^2\tilde{\mathrm{M}}(\bar{\boldsymbol{x}})^{-1}\| = 1/\sigma_m\left(\nabla^2\tilde{\mathrm{M}}(\bar{\boldsymbol{x}})\right)$$

which can be bounded using lower bounds for $\sigma_m\left(\nabla^2\tilde{\mathrm{M}}(\bar{\boldsymbol{x}})\right)$. In particular,

$$\sigma_m\left(\nabla^2\tilde{\mathrm{M}}(\bar{\boldsymbol{x}})\right) \geqslant \gamma^{-1}\sigma_m\left(\tilde{S}_\gamma(\bar{\boldsymbol{x}})\right)\sigma_m\left((I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))\right) \qquad (17)$$

and the problem is to compute the minimum singular values of $\tilde{S}_\gamma(\bar{\boldsymbol{x}})$ and $(I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))$.

First of all, for a symmetric matrix $A \in \mathbb{R}^{n\times n}$ it holds $\sigma_m(A) = |\lambda_m(A)|$. Recalling (13) we have that $\sigma_m\left(\tilde{S}_\gamma(\bar{\boldsymbol{x}})\right) \geqslant 1 - \gamma L > 0$. Since $I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}})$ is symmetric and positive definite, then

$$\sigma_m\left((I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))\right) = \lambda_m\left((I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))\right)$$

and it is necessary to lower bound the spectrum of $(I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))$.

It holds that

$$\lambda_m\left((I - \tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}}))\right) = 1 - \lambda_M\left(\tilde{P}_\gamma(\bar{\boldsymbol{x}})\tilde{S}_\gamma(\bar{\boldsymbol{x}})\right)$$

and an upper bound for $\lambda_M \left( \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right)$ must be found. Notice that $\tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}})$ is symmetric and positive definite [20, Appendix B]. The following inequalities hold

$$\lambda_M \left( \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right) = \left\| \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right\|$$
$$\leqslant \left\| \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \right\| \left\| \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right\| \leqslant \left\| \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right\| \leqslant 1 - \gamma m$$

where the fact that $\left\| \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \right\| \leqslant 1$ and the result (13) were used.

Finally, these results yield

$$\sigma_m \left( (I - \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}})) \right) = \lambda_m \left( (I - \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}})) \right)$$
$$\geqslant 1 - \lambda_M \left( \tilde{P}_\gamma(\bar{\boldsymbol{x}}) \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right) \geqslant 1 - (1 - \gamma m) = \gamma m. \quad (18)$$

Finally, substituting (18) and $\sigma_m \left( \tilde{S}_\gamma(\bar{\boldsymbol{x}}) \right) \geqslant 1 - \gamma L$ into (17), it follows

$$\sigma_m \left( \nabla^2 \tilde{M}(\bar{\boldsymbol{x}}) \right) \geqslant \gamma^{-1}(1 - \gamma L)\gamma m = m(1 - \gamma L)$$

and thus

$$K \leqslant \frac{1}{m(1 - \gamma L)}. \quad (19)$$

Notice that the same bound holds for $\bar{K}$ as well since, like $f$, it is $h_k \in \mathcal{S}_{m,L}(\mathbb{R}^n)$.

*A.1.3. Approximation error bound*

The results derived in the previous sections can be used to bound the approximation error as follows

$$\left\| \bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^* \right\| \leqslant \frac{2(1 - \gamma m)}{m(1 - \gamma L)} \left( L \left\| \boldsymbol{x}_k - \boldsymbol{x}_k^* \right\| \right.$$
$$\left. + \frac{L(1 - \gamma m)}{m(1 - \gamma L)} T_s C_0 + T_s C_0 \right)$$
$$\leqslant a_1 \left\| \boldsymbol{x}_k - \boldsymbol{x}_k^* \right\| + a_0 \quad (20)$$

where the coefficients are defined as

$$a_1 = \frac{2L(1 - \gamma m)}{m(1 - \gamma L)}$$
$$a_0 = 2C_0 T_s \frac{1 - \gamma m}{m(1 - \gamma L)} \left[ \frac{L(1 - \gamma m)}{m(1 - \gamma L)} + 1 \right]$$

with $a_0$ that linearly depends on the sampling time $T_s$.

**A.2. Early termination errors**

The previous section derived an upper bound to the approximation error introduced by the approximate FBE used in the prediction step instead of the correct FBE. However there are two other sources of errors, namely the early termination of the minimisation algorithms applied to the $\tilde{M}(\boldsymbol{x})$ and the FBE in the prediction and correction steps, respectively. These will be the focus of the current section.

Assume that the minimisations are carried out using the *quasi-Newton* Algorithm 2 in [20] applied to the $\tilde{M}$ or the $M$. This method is described in Algorithm 2 in the general case.

**Remark 7** *In case $L$ is not known, it is possible to apply Algorithm 1 of [20] which includes a line-search procedure for choosing $\gamma$.*

---

**Algorithm 2** Quasi-Newton method for the FBE.

**Input:** $\hat{\boldsymbol{x}}^0 \in \mathbb{R}^n$, envelope parameter $\gamma \in (0, 1/L)$, maximum number of iterations $I$.
1: $i \leftarrow 0$
2: **while** $i < I$ **do**
3:     compute the nonsingular matrix $B^i$ using the BFGS method
4:     compute the descent direction
$$\mathbf{d}^i = -(B^i)^{-1} \nabla M(\hat{\boldsymbol{x}}^i)$$
5:     select the step-size $\tau^i \geqslant 0$ s.t. $M(\boldsymbol{w}^i) \leqslant M(\hat{\boldsymbol{x}}^i)$ where $\boldsymbol{w}^i = \hat{\boldsymbol{x}}^i + \tau^i \mathbf{d}^i$
6:     $\hat{\boldsymbol{x}}^{i+1} \leftarrow \text{prox}_{\gamma g}(\boldsymbol{x} - \gamma \nabla_{\boldsymbol{x}} f(\boldsymbol{x}; t_k))$
7:     $i \leftarrow i + 1$
8: **end while**

---

The sequence $\{\hat{\boldsymbol{x}}^m\}_{m \in \mathbb{N}}$ produced by Algorithm 2 can be proved to converge to a critical point with a *super-linear* rate, by a combination of Theorems 3.6 and 4.3 in [20]. Moreover the result holds globally by strong convexity.

Since super-linear convergence implies linear convergence as well, then there exist $\zeta \in (0, 1)$ (see (8) for an estimate of its value) such that

$$\left\| \hat{\boldsymbol{x}}^{p+1} - \bar{\boldsymbol{x}} \right\| \leqslant \zeta \left\| \hat{\boldsymbol{x}}^p - \bar{\boldsymbol{x}} \right\| \quad p = 1, 2, \dots, P$$

and

$$\left\| \hat{\boldsymbol{x}}^{c+1} - \boldsymbol{x}_{k+1}^* \right\| \leqslant \zeta \left\| \hat{\boldsymbol{x}}^c - \boldsymbol{x}_{k+1}^* \right\| \quad c = 1, 2, \dots, C$$

where $\hat{\boldsymbol{x}}^p$ and $\hat{\boldsymbol{x}}^c$ are the dummy variables used during prediction and correction.

The dummy variables are initialized as follows: for the prediction $\hat{\boldsymbol{x}}^0 = \boldsymbol{x}_k$, for the correction $\hat{\boldsymbol{x}}^0 = \tilde{\boldsymbol{x}}_{k+1|k} = \hat{\boldsymbol{x}}^P$. Therefore iterating

$$\left\| \tilde{\boldsymbol{x}}_{k+1|k} - \bar{\boldsymbol{x}} \right\| = \left\| \tilde{\boldsymbol{x}}_{k+1|k} - \boldsymbol{x}_{k+1|k} \right\| \leqslant \zeta^P \left\| \boldsymbol{x}_k - \boldsymbol{x}_{k+1|k} \right\|$$
$$\left\| \boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^* \right\| \leqslant \zeta^C \left\| \tilde{\boldsymbol{x}}_{k+1|k} - \boldsymbol{x}_{k+1}^* \right\|.$$

**A.3. Overall error bound**

During the previous sections the following bounds have been derived

$$\left\| \bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^* \right\| \leqslant a_1 \left\| \boldsymbol{x}_k - \boldsymbol{x}^*(t_k) \right\| + a_0 \quad (21)$$
$$\left\| \tilde{\boldsymbol{x}}_{k+1|k} - \bar{\boldsymbol{x}} \right\| \leqslant \zeta^P \left\| \boldsymbol{x}_k - \boldsymbol{x}_{k+1|k} \right\| \quad (22)$$
$$\left\| \boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^* \right\| \leqslant \zeta^C \left\| \tilde{\boldsymbol{x}}_{k+1|k} - \boldsymbol{x}_{k+1}^* \right\| \quad (23)$$

and therefore the last thing to do is to combine them to derive a bound for the error $\left\| \boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^* \right\|$. Following the same steps of [14, Appendix B] from inequalities (21), (22) and the results above

it is possible to compute

$$\begin{aligned}
\left\|\tilde{\boldsymbol{x}}_{k+1|k} - \boldsymbol{x}_{k+1}^*\right\| &\leqslant \left\|\tilde{\boldsymbol{x}}_{k+1|k} - \bar{\boldsymbol{x}}\right\| + \left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \\
&\leqslant \zeta^P \left\|\boldsymbol{x}_k - \bar{\boldsymbol{x}}\right\| + \left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \\
&\leqslant \zeta^P (\left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| + \left\|\boldsymbol{x}_k^* - \boldsymbol{x}_{k+1}^*\right\| + \left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\|) + \\
&\qquad\qquad \left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \\
&\leqslant \zeta^P \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| + \zeta^P \frac{(1-\gamma m)T_{\mathrm{s}} C_0}{m(1-\gamma L)} + \\
&\qquad\qquad (\zeta^P + 1)\left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \\
&\leqslant [\zeta^P + a_1(\zeta^P + 1)] \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| \\
&\qquad + \zeta^P \frac{(1-\gamma m)T_{\mathrm{s}} C_0}{m(1-\gamma L)} + (\zeta^P + 1)a_0.
\end{aligned}$$

Using now inequality (23) it follows

$$\left\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^*\right\| \leqslant A_1 \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| + A_0 \qquad (24)$$

where

$$\begin{aligned}
A_1 &= \zeta^C[\zeta^P + a_1(\zeta^P + 1)] \\
&= \zeta^C \left[\zeta^P + (\zeta^P + 1)\frac{2L(1-\gamma m)}{m(1-\gamma L)}\right] \\
A_0 &= \zeta^C \left[\zeta^P \frac{(1-\gamma m)T_{\mathrm{s}} C_0}{m(1-\gamma L)} + (\zeta^P + 1)a_0\right].
\end{aligned}$$

Therefore, for the algorithm to converge to a bounded error it is necessary that $A_1 < 1$, which must be guaranteed by choosing suitable prediction and correction horizons. Notice that $A_0$ depends linearly on the sampling time $T_{\mathrm{s}}$, therefore it holds that

$$\limsup_{k\to\infty} \left\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^*\right\| = O(\zeta^C T_{\mathrm{s}}).$$

We have thus proved Theorem 1. ∎

### A.3.1. $O(T_{\mathrm{s}}^2)$ convergence

Suppose now that Assumption 3 holds, it is possible to show that the error bound tends to $O(T_{\mathrm{s}}^2)$.

Consider that

$$\left\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\right\| \leqslant \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{x}_{k+1}^*; t_{k+1}) - \nabla h_k(\boldsymbol{x}_{k+1}^*)\right\| \leqslant \|\boldsymbol{\epsilon}\|, \qquad (25)$$

where $\boldsymbol{\epsilon}$ is the residual of the Taylor expansion of $\nabla_{\boldsymbol{x}} f(\cdot; t_{k+1})$. Under Assumption 3, the Taylor residual can be bound as follows

$$\begin{aligned}
\|\boldsymbol{\epsilon}\| &\leqslant \frac{1}{2}\Big(\left\|\nabla_{\boldsymbol{xxx}} f(\boldsymbol{x}_k; t_k)\right\| \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\|^2 + \\
&\quad + T_{\mathrm{s}} \left\|\nabla_{t\boldsymbol{xx}} f(\boldsymbol{x}_k; t_k)\right\| \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + \\
&\quad + T_{\mathrm{s}} \left\|\nabla_{\boldsymbol{x}t\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\| \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + \\
&\qquad\qquad + T_{\mathrm{s}}^2 \left\|\nabla_{tt\boldsymbol{x}} f(\boldsymbol{x}_k; t_k)\right\|\Big) \\
&\leqslant \frac{C_1}{2} \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\|^2 + T_{\mathrm{s}} C_2 \left\|\boldsymbol{x}_{k+1}^* - \boldsymbol{x}_k\right\| + \frac{1}{2}T_{\mathrm{s}}^2 C_3.
\end{aligned} \qquad (26)$$

We want now to use this result to compute an upper bound to $\left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\|$ that is stricter than (21). First of all, substituting (25) into the bound (14) we obtain

$$\|\nabla\tilde{\mathrm{M}}(\boldsymbol{x}_{k+1}^*)\| \leqslant \left\|\tilde{S}_\gamma(\boldsymbol{x}_{k+1}^*)\right\| \left\|\tilde{R}_\gamma(\boldsymbol{x}_{k+1}^*)\right\| \leqslant (1-\gamma m)\|\boldsymbol{\epsilon}\| \qquad (27)$$

where we used the bound $\|\tilde{S}_\gamma(\boldsymbol{x}_{k+1}^*)\| \leqslant 1 - \gamma m$. Therefore applying the bound (27) into (12) yields

$$\begin{aligned}
\left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| &\leqslant K \left\|\nabla\tilde{\mathrm{M}}(\boldsymbol{x}_{k+1}^*)\right\| \\
&\leqslant K(1-\gamma m)\|\boldsymbol{\epsilon}\| \leqslant \frac{1-\gamma m}{m(1-\gamma L)}\|\boldsymbol{\epsilon}\|
\end{aligned}$$

where the bound (19) for $K$ was used. Substituting (26) finally we get

$$\left\|\bar{\boldsymbol{x}} - \boldsymbol{x}_{k+1}^*\right\| \leqslant a_2 \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\|^2 + a_1 \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| + a_0 \qquad (28)$$

where

$$\begin{aligned}
a_2 &= \frac{1-\gamma m}{m(1-\gamma L)}\frac{C_1}{2} \\
a_1 &= T_{\mathrm{s}}\frac{1-\gamma m}{m(1-\gamma L)}\left[\frac{1-\gamma m}{m(1-\gamma L)}C_0 C_1 + C_2\right] \\
a_0 &= T_{\mathrm{s}}^2 \frac{1-\gamma m}{m(1-\gamma L)}\left[\left(\frac{1-\gamma m}{m(1-\gamma L)}\right)^2 \frac{C_1 C_0^2}{2} + \right. \\
&\qquad\left. + \frac{1-\gamma m}{m(1-\gamma L)}C_0 C_2 + \frac{1}{2}C_3\right].
\end{aligned}$$

Notice that $a_1$ and $a_0$ linearly depend on $T_{\mathrm{s}}$ and $T_{\mathrm{s}}^2$, respectively.

With computations very similar to those carried out during the previous section, exchanging (28) for (21), it is then possible to compute the bound

$$\left\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^*\right\| \leqslant A_2 \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\|^2 + A_1 \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| + A_0 \qquad (29)$$

where

$$\begin{aligned}
A_2 &= \zeta^C(\zeta^P + 1)a_2 \\
A_1 &= \zeta^C[\zeta^P + a_1(\zeta^P + 1)] \\
A_0 &= \zeta^C \left[\zeta^P \frac{1-\gamma m}{m(1-\gamma L)}C_0 T_{\mathrm{s}} + (\zeta^P + 1)a_0\right].
\end{aligned}$$

In order to prove convergence, it is now possible to use the argument presented in [14, Appendix B], which guarantees convergence if $\tau > \zeta^P \zeta^C$,

$$\begin{aligned}
T_{\mathrm{s}} &< \frac{\tau - \zeta^P \zeta^C}{\zeta^C(\zeta^P + 1)} \times \\
&\quad \times \left\{\frac{1-\gamma m}{m(1-\gamma L)}\left[\frac{1-\gamma m}{m(1-\gamma L)}C_0 C_1 + C_2\right]\right\}^{-1} =: \bar{T}_{\mathrm{s}},
\end{aligned}$$

and

$$\left\|\boldsymbol{x}_0 - \boldsymbol{x}_0^*\right\| \leqslant \frac{\tau - A_1}{A_2} =: \bar{R}.$$

Therefore the asymptotic error satisfies

$$\limsup_{k\to\infty} \left\|\boldsymbol{x}_k - \boldsymbol{x}_k^*\right\| \leqslant O(T_{\mathrm{s}}^2 \zeta^C) + O(T_{\mathrm{s}} \zeta^C \zeta^P),$$

which proves Theorem 2. ∎