A Novel Algorithm for GARCH Model Estimation

Chenyu Gao
Sch. of Inf. Sci. & Tech.
ShanghaiTech University
Shanghai, China
gaochy@shanghaitech.edu.cn

Ziping Zhao
Sch. of Inf. Sci. & Tech.
ShanghaiTech University
Shanghai, China
zipingzhao@shanghaitech.edu.cn

Daniel P. Palomar
Dept. of ECE & Dept. of IEDA
The Hong Kong University of Science & Technology
Hong Kong, China
palomar@ust.hk

2023 IEEE Statistical Signal Processing Workshop (SSP) | 978-1-6654-5245-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/SSP53291.2023.10208065

***Abstract*—Generalized autoregressive conditional heteroskedasticity (GARCH) is a popular model to describe the time-varying conditional volatility of a time series, which is widely used in signal processing and machine learning. In this paper, we focus on the model parameter estimation of GARCH based on the Gaussian maximum likelihood estimation method. Due to the recursively coupling nature of parameters in GARCH, the optimization problem is highly non-convex. In this paper, we propose a novel algorithm based on the block majorization-minimization algorithmic framework, which can take care of the per-block variable structures for efficient problem solving. Numerical experiments demonstrate that the proposed algorithm can achieve comparable and even better performance in terms of parameter estimation errors. More importantly, estimated parameters from our algorithm always guarantee a stationary model, which is a desirable property in time series volatility modeling.**

***Index Terms*—GARCH, volatility modeling, maximum likelihood, non-convex optimization, majorization-minimization.**

I. Introduction

In time series analysis, classical models commonly assume that the conditional volatility (i.e., the conditional standard deviation) of time series residuals is a constant over time, i.e., homoskedastic; however, many real applications have shown large deviation from this idealized assumption, exhibiting a heteroskedastic property [1]. Engle [2] observed that the volatility of an economic time series actually is time-varying which depends on the past information; then he proposed an autoregressive conditional heteroskedastic (ARCH) model to characterize such a phenomenon, which explains the volatility of a time series based on its past residuals. After that, the generalized autoregressive conditional heteroskedasticity (GARCH) model [3] was proposed. GARCH extends ARCH to describe time series whose volatility is serially autocorrelated following an autoregressive moving average process. More specifically, GARCH model uses both past residuals and past volatility to forecast the future volatility. Later, numerous GARCH-type models have been proposed [4], [5]; for more details on GARCH, readers can refer to [6], [7], [8].

The GARCH model has been widely used in many applications for modeling time series with time-varying volatility due to its desirable property in modeling volatility clustering and heavy-tailed distributions. GARCH was initially used in econometrics to model the volatility of financial returns [9], [10], [11], [12], [13]. Later on, its application was extended to many other fields. For example, in array signal processing, radar clustering can be described by GARCH process due to its volatility clustering characteristics [14]; in speech signal processing, GARCH is used for speech enhancement since speech signals in the short-time Fourier transform domain demonstrate both volatility clustering and heavy-tailed behavior [15]; in biomedical signal processing, GARCH is used for variance modeling in brain MRI classification [16]. Besides, there are many other applications such as traffic prediction [17] and anomaly detection [18], [19].

Despite of its powerful capacity and wide applications, the estimation of a GARCH model is difficult. One simple approach is through ordinary least squares estimation. However, since the residuals in GARCH are heteroskedastic, the ordinary least squares estimation is inefficient. In literature, a common estimation method is the Gaussian maximum likelihood estimation (MLE). Solving the MLE problem is nontrivial due to the recursively coupling nature of model parameters in GARCH. Besides that, to guarantee the positivity of the volatility and the stationarity of the model, positivity constraints and stationarity constraints on model parameters should be considered, respectively [20]. In the seminal work [3], a quasi-Newton method called Berndt–Hall–Hall–Hausman (BHHH) was proposed. Packages `tseries` [21], `fGarch` [22], and `rugarch` [23] in R also consider other types of quasi-Newton methods, like Broyden-Fletcher-Goldfarb-Shanno (BFGS) in `tseries` and limit-memory BFGS (L-BFGS) method in `fGarch` and `rugarch`. A drawback of all these quasi-Newton methods is that they drop the necessary constraints and the MLE problem is taken as an unconstrained one [24], [20], [25]. Besides these methods, the R package `rugarch` [23] and the `Econometrics Toolbox` in the commercial software MATLAB [26] also apply the sequential quadratic programming (SQP) and interior-point method (IPM) to solve the MLE problem as a constrained optimization problem. However, since both SQP and IPM reply on existing optimization subroutines, they have multiple loops in nature and the convergence of the algorithms are not always observed in practice.

In this paper, we propose a novel single-loop algorithm with guaranteed convergence for Gaussian MLE of GARCH based on the block majorization-minimization (BMM) algorithmic framework [27], [28]. The algorithm has lower per-iteration

This work was supported in part by the National Nature Science Foundation of China (NSFC) under Grant 62001295 and in part by the Shanghai Sailing Program under Grant 20YF1430800.

computational complexity and can take care of the per-block variable structures for efficient problem solving. Based on the general idea of BMM, we divide the MLE problem into several variable blocks in which each block-wise sub-problem can be solved with cheap solution. Numerical experiments demonstrate that the proposed algorithm can achieve comparable and even better performance in terms of parameter estimation errors. More importantly, the estimated GARCH model can always be guaranteed to be stationary.

II. Preliminaries

Considering a time series r_t for $t = 1, \ldots, N$, which we denote as $\{r_t\}$ hereafter, we define its residual as

$$a_t = r_t - \mu_t, \tag{1}$$

where $\mu_t = \mathsf{E}(r_t|\mathcal{F}_{t-1})$ is the conditional mean with $\mathcal{F}_{t-1}$ denoting the information available up to time $t-1$. The residual series $\{a_t\}$ is said to follow a GARCH model [3] if its conditional variance, i.e., the squared conditional volatility, $\sigma_t^2 = \mathsf{Var}(a_t|\mathcal{F}_{t-1})$ satisfies the following recursion relation

$$\sigma_t^2 = \omega + \sum_{i=1}^{q} \alpha_i a_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2, \tag{2}$$

where ω, $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_q]^\top$, and $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_p]^\top$ are the model parameters. The integers $q \geq 0$ and $p \geq 0$ specify the order of a GARCH model and Eq. (2) is referred to as GARCH(q, p). In the seminal paper [3], several conditions are enforced on the GARCH parameters. To ensure the positivity of σ_t^2 , it is required $\omega > 0$, $\boldsymbol{\alpha} \geq \mathbf{0}$, and $\boldsymbol{\beta} \geq \mathbf{0}$. The recursive relation (2) is an autoregressive moving average process; and for it to be stationary, i.e., the unconditional variance of $\{a_t\}$ is finite, constraint $\mathbf{1}^\top\boldsymbol{\alpha} + \mathbf{1}^\top\boldsymbol{\beta} < 1$ is also necessary.

By Eq. (2), the residual series $\{a_t\}$ is modeled as

$$a_t = \sigma_t \epsilon_t, \tag{3}$$

where $\{\epsilon_t\}$ is a white noise series with zero mean and unit variance. In GARCH modeling, $\{\epsilon_t\}$ could be assumed to follow a specific distribution. In this paper, we study a special case where $\{\epsilon_t\}$ follows a Gaussian distribution [3].

III. Problem description

The Gaussian MLE problem for GARCH is as follows [3]:

$$\begin{array}{ll} \underset{\omega, \boldsymbol{\alpha}, \boldsymbol{\beta}, \{\sigma_t^2\}}{\mathsf{minimize}} & \displaystyle\sum_{t=1}^{N} \log \sigma_t^2 + \sum_{t=1}^{N} \frac{a_t^2}{\sigma_t^2} \\ \mathsf{subject\ to} & \text{equation (2) for } t = 1, \ldots, N \\ & \omega > 0,\ \boldsymbol{\alpha} \geq \mathbf{0},\ \boldsymbol{\beta} \geq \mathbf{0},\ \mathbf{1}^\top\boldsymbol{\alpha} + \mathbf{1}^\top\boldsymbol{\beta} < 1, \end{array} \tag{4}$$

where the initial values σ_t^2 with $t \leq 0$ are given values. Equation (2) makes the optimization variables recursively coupled. Hence, problem (4) is a non-convex constrained multi-block optimization problem, which is hard to solve analytically. For the two open convex constraints, i.e., $\omega > 0$ and $\mathbf{1}^\top\boldsymbol{\alpha} + \mathbf{1}^\top\boldsymbol{\beta} < 1$, we will relax them to be closed convex ones $\omega \geq 0_\varepsilon$ and $\mathbf{1}^\top\boldsymbol{\alpha} + \mathbf{1}^\top\boldsymbol{\beta} \leq 1_\varepsilon$, respectively, where $1_\varepsilon = 1 - \varepsilon$ and $0_\varepsilon = 0 + \varepsilon$ with ε a given small constant. This relaxation is a common practice in all existing GARCH solvers like the Econometrics Toolbox [26] in MATLAB and package tseries [21], fGarch [22], and rugarch [23] in R, which is mainly to ease the numerical algorithm design.

In this paper, we propose to solve a penalized MLE formulation by relaxing the equality constraints (2) as a regularization term in the objective:

$$\begin{array}{ll} \underset{\omega, \boldsymbol{\gamma}, \{\sigma_t^2\}}{\mathsf{minimize}} & \displaystyle\sum_{t=1}^{N} \log \sigma_t^2 + \sum_{t=1}^{N} \frac{a_t^2}{\sigma_t^2} + \frac{\eta}{2} \sum_{t=1}^{N} \left(\sigma_t^2 - \omega - \boldsymbol{\gamma}^\top \mathbf{c}_t^2\right)^2 \\ \mathsf{subject\ to} & \omega \geq 0_\varepsilon,\ \boldsymbol{\gamma} \geq \mathbf{0},\ \mathbf{1}^\top\boldsymbol{\gamma} \leq 1_\varepsilon, \end{array} \tag{5}$$

where η is a tuning hyperparameter, and we define $\boldsymbol{\gamma} = \left[\boldsymbol{\alpha}^\top, \boldsymbol{\beta}^\top\right]^\top$ and $\mathbf{c}_t^2 = [a_{t-1}^2, \ldots, a_{t-q}^2, \sigma_{t-1}^2, \ldots, \sigma_{t-p}^2]^\top$ for notational simplicity. This penalty problem formulation will help to decouple the variables in the constraints (2) and to design a single-loop algorithm with cheap variable updates.

IV. MLE for GARCH based on BMM

In this section, we develop an algorithm for problem (5) based on BMM [27], [28] which can take the per-block variable structures into consideration. In a BMM algorithm, each of the variable blocks will be updated successively by optimizing an upper-bound surrogate function. For problem (5), we divide the optimization variables into three blocks, namely ω, $\boldsymbol{\gamma}$, and $\{\sigma_t^2\}$, and through carefully constructing surrogate functions, we are able to derive closed-form updating equations for all variable blocks.

A. Solving the ω-block sub-problem

Since only the last term in the objective of (5) is related to ω, the sub-problem for ω is given by[1]

$$\begin{array}{ll} \underset{\omega}{\mathsf{minimize}} & \displaystyle\frac{1}{2} \sum_{t=1}^{N} \left(\underline{\sigma_t^2} - \omega - \underline{\boldsymbol{\gamma}}^\top \underline{\mathbf{c}_t^2}\right)^2 \\ \mathsf{subject\ to} & \omega \geq 0_\varepsilon, \end{array} \tag{6}$$

where $\underline{\mathbf{c}_t^2} = [a_{t-1}^2, \ldots, a_{t-q}^2, \underline{\sigma_{t-1}^2}, \ldots, \underline{\sigma_{t-p}^2}]^\top$. Problem (6) is a convex scalar quadratic optimization problem which has an analytical solution

$$\omega^\star = \max\left\{ \frac{\sum_{t=1}^{N} \left(\underline{\sigma_t^2} - \underline{\boldsymbol{\gamma}}^\top \underline{\mathbf{c}_t^2}\right)}{N}, 0_\varepsilon \right\}. \tag{7}$$

B. Solving the γ-block sub-problem

The sub-problem with respect to $\boldsymbol{\gamma}$ is given by

$$\begin{array}{ll} \underset{\boldsymbol{\gamma}}{\mathsf{minimize}} & \displaystyle\frac{1}{2} \sum_{t=1}^{N} \left(\underline{\sigma_t^2} - \underline{\omega} - \boldsymbol{\gamma}^\top \underline{\mathbf{c}_t^2}\right)^2 \\ \mathsf{subject\ to} & \boldsymbol{\gamma} \geq \mathbf{0},\ \mathbf{1}^\top\boldsymbol{\gamma} \leq 1_\varepsilon. \end{array} \tag{8}$$

Problem (8) is a convex quadratic optimization problem. Although it can be solved by many standard solvers, in the

[1]Throughout this paper, underlined variables denote those whose values are given as constants.

following, we aim to obtain a closed-form update by finding proper surrogate functions. We first introduce a useful result.

Lemma 1 ([28])**.** *The quadratic form* $\mathbf{x}^\top \mathbf{D}\mathbf{x}$*, where* $\mathbf{D}$ *is a symmetric matrix, can be upper-bounded as*

$$\mathbf{x}^\top \mathbf{D}\mathbf{x} \leq \mathbf{x}^\top \mathbf{L}\mathbf{x} + 2\mathbf{x}^\top(\mathbf{D}-\mathbf{L})\mathbf{y} + \mathbf{y}^\top(\mathbf{L}-\mathbf{D})\mathbf{y},$$

with $\mathbf{L} \succeq \mathbf{D}$*, where the equality is attained when* $\mathbf{x} = \mathbf{y}$.

Based on Lemma 1, an upper-bound surrogate for the objective in (8) can be computed in the following way

$$\begin{aligned}
&\frac{1}{2}\sum_{t=1}^{N}\left(\underline{\sigma_t^2} - \underline{\omega} - \boldsymbol{\gamma}^\top \underline{\mathbf{c}_t^2}\right)^2 \\
&= \frac{1}{2}\boldsymbol{\gamma}^\top \underline{\mathbf{C}}\boldsymbol{\gamma} - \sum_{t=1}^{N}\left(\underline{\sigma_t^2} - \underline{\omega}\right)\boldsymbol{\gamma}^\top \underline{\mathbf{c}_t^2} + \mathsf{const.} \\
&\leq \frac{1}{2}\varphi\boldsymbol{\gamma}^\top\boldsymbol{\gamma} + \boldsymbol{\gamma}^\top\mathbf{v} + \mathsf{const.},
\end{aligned}$$

where $\underline{\mathbf{C}} = \sum_{t=1}^{N}\underline{\mathbf{c}_t^2}\left(\underline{\mathbf{c}_t^2}\right)^\top$, φ is any number satisfying $\varphi \geq \lambda_{\max}(\underline{\mathbf{C}})$ with $\lambda_{\max}(\underline{\mathbf{C}})$ the maximal eigenvalue of $\underline{\mathbf{C}}$, and $\mathbf{v} = -\varphi\underline{\boldsymbol{\gamma}} + \sum_{t=1}^{N}\left(\underline{\boldsymbol{\gamma}}^\top\underline{\mathbf{c}_t^2} - \underline{\sigma_t^2} + \underline{\omega}\right)\underline{\mathbf{c}_t^2}$. With this surrogate function, the sub-problem for $\boldsymbol{\gamma}$ is given by

$$\begin{aligned}
\underset{\boldsymbol{\gamma}}{\mathsf{minimize}} \quad & \frac{1}{2}\varphi\boldsymbol{\gamma}^\top\boldsymbol{\gamma} + \boldsymbol{\gamma}^\top\mathbf{v} \\
\mathsf{subject\ to} \quad & \boldsymbol{\gamma} \geq \mathbf{0},\ \mathbf{1}^\top\boldsymbol{\gamma} \leq 1_\varepsilon.
\end{aligned} \tag{9}$$

The optimal solution to problem (9) can be computed based on the following result.

Proposition 2 ([29])**.** *The solution* $\boldsymbol{\gamma}^\star$ *to problem* (9) *is*

$$\gamma_i^\star = \begin{cases}
-\frac{v_i}{w}, & \sum_{j\in\mathcal{V}} v_j + w1_\epsilon > 0, i \in \mathcal{V} \\
0, & \sum_{j\in\mathcal{V}} v_j + w1_\epsilon > 0, i \notin \mathcal{V} \\
\frac{\sum_{j\notin\mathcal{I}} v_j + w1_\varepsilon}{(q+p-\mathsf{Card}(\mathcal{I}))w} - \frac{v_i}{w}, & \sum_{j\in\mathcal{V}} v_j + w1_\varepsilon \leq 0, i \notin \mathcal{I} \\
0, & \sum_{j\in\mathcal{V}} v_j + w1_\varepsilon \leq 0, i \in \mathcal{I},
\end{cases} \tag{10}$$

where $w = \lambda_{\max}(\underline{\mathbf{C}})$, $\mathcal{V} = \{i \,|v_i < 0,\ i = 1, \ldots, q+p\}$*, and* $\mathcal{I} = \left\{i \,\middle|\, -\frac{\sum_{j\notin\mathcal{I}} v_j + w1_\varepsilon}{q+p-\mathsf{Card}(\mathcal{I})} + v_i \geq 0,\ i = 1, \ldots, q+p\right\}$.

Based on Proposition 2, an efficient computational procedure to find the solution $\boldsymbol{\gamma}^\star$ is given in Algorithm 1.

Algorithm 1 Find the solution $\boldsymbol{\gamma}^\star$

```
if ∑_{i∉V} v_i + w1_ε > 0 then
    γ_i = −v_i/w with i ∈ V and γ_i = 0 with i ∉ V
else
    build a set I, let i ∉ V be in the set I
    start a loop
    find the max element v_i where i ∉ I
    put i into the set I
    if −(∑_{j∉I} v_j + w1_ε)/(q+p−Card(I)) + v_i ≥ 0 then
    else
        remove i from the set I
        γ_i = 0 with i ∈ I
        γ_i = (∑_{j∉I} v_j + w1_ε)/((q+p−Card(I))w) − v_i/w with i ∉ I
        terminate the loop
    end if
end if
```

C. Solving the $\{\sigma_t^2\}$-block sub-problem

Define $\underline{o_t} = \underline{\omega} + \sum_{i=1}^{q}\underline{\alpha_i}a_{t-i}^2$. The sub-problem for the $\{\sigma_t^2\}$-block is

$$\begin{aligned}
\underset{\{\sigma_t^2\}}{\mathsf{minimize}} \quad & \sum_{t=1}^{N}\log\sigma_t^2 + \sum_{t=1}^{N}\frac{a_t^2}{\sigma_t^2} \\
& + \frac{\eta}{2}\sum_{t=1}^{N}\left(\sigma_t^2 - \sum_{j=1}^{p}\underline{\beta_j}\sigma_{t-j}^2 - \underline{o_t}\right)^2,
\end{aligned} \tag{11}$$

which is an unconstrained non-convex optimization problem. Denote the objective in problem (11) as g. The optimal solutions are the first-order stationary points of g, i.e., solutions to a cubic system of equations $\frac{\partial g(\{\sigma_t^2\})}{\partial \sigma_t^2} = 0$ with $t = 1, \ldots, N$. Since optimization variables in the third term of g are coupled, solving the cubic system is generally non-trivial. In the following, we aim to obtain a cheap update for $\{\sigma_t^2\}$ via finding a proper surrogate function for $g(\{\sigma_t^2\})$.

Define $\underline{\tilde{\boldsymbol{\beta}}} = \left[1, -\underline{\boldsymbol{\beta}}^\top\right]^\top$ and $\boldsymbol{\sigma}_t^2 = [\sigma_t^2, \ldots, \sigma_{t-p}^2]^\top$. Applying Lemma 1 to the third term in g, we obtain an upper-bound surrogate function as follows:

$$\begin{aligned}
\bar{g}(\{\sigma_t^2\}) = &\sum_{t=1}^{N}\log\sigma_t^2 + \sum_{t=1}^{N}\frac{a_t^2}{\sigma_t^2} \\
&+ \frac{\eta}{2}\left\|\underline{\tilde{\boldsymbol{\beta}}}\right\|_2^2\sum_{t=1}^{N}(\boldsymbol{\sigma}_t^2)^\top\boldsymbol{\sigma}_t^2 + \eta\sum_{t=1}^{N}\mathbf{z}_t^\top\boldsymbol{\sigma}_t^2,
\end{aligned} \tag{12}$$

where $\mathbf{z}_t = \left(\underline{\tilde{\boldsymbol{\beta}}}^\top\underline{\boldsymbol{\sigma}_t^2} - \underline{o_t}\right)\underline{\tilde{\boldsymbol{\beta}}} - \left\|\underline{\tilde{\boldsymbol{\beta}}}\right\|_2^2\underline{\boldsymbol{\sigma}_t^2}$. Now, the variables $\{\sigma_t^2\}$ in $\bar{g}$ become separable. In the following, we solve the sub-problems for $\{\sigma_t^2\}$ with the surrogate function $\bar{g}$. Computing the partial derivatives of $\bar{g}$ with respect to σ_t^2 for $t = 1, \ldots, N$, multiplying the functions by $(\sigma_t^2)^2$, and setting them equal to $\mathbf{0}$ leads to N separable cubic equations

$$\eta\kappa_t\left\|\underline{\tilde{\boldsymbol{\beta}}}\right\|_2^2(\sigma_t^2)^3 + \eta\sum_{j=0}^{p}\left[\mathbf{z}_{t+j}\right]_{j+1}(\sigma_t^2)^2 + \sigma_t^2 - a_t^2 = 0, \quad \text{for } t = 1, \ldots, N, \tag{13}$$

where $\kappa_t = p+1$ for $1 \leq t \leq N-p$ and $\kappa_t = N-t+1$ for $N-p < t \leq N$. For each cubic equation, its solutions can be obtained via the analytical "cubic formula" and then we can choose the one achieving the minimal value for $\bar{g}$.

Algorithm 2 summarizes the overall algorithm solving the Gaussian MLE for GARCH. We also summarize the convergence property of the BMM method in the following.

Theorem 3. *Denote the iterates from the* k*-th iteration of Alg. 2 by* $\omega^{(k)}$*,* $\boldsymbol{\gamma}^{(k)}$*, and* $\{(\sigma_t^2)^{(k)}\}$*. The sequence*

Table I
PERFORMANCE COMPARISONS OF DIFFERENT ALGORITHMS FOR ESTIMATION OF A GARCH(2,3) MODEL.

Method		MSE(ω) (SE(ω))	MSE($\boldsymbol{\alpha}$) (SE($\boldsymbol{\alpha}$))	MSE($\boldsymbol{\beta}$) (SE($\boldsymbol{\beta}$))	MSE(σ_t)	Constraint violation	Avg. objective	Avg. Time (sec.)
BHHH	[3]	3.8341×10^{-4} (1.6480×10^{-3})	0.0199 (0.0116)	0.0296 (0.0159)	0.4554	44/100	−147.9819	0.3221
BFGS	'tseries' [21]	1.3422×10^{-2} (1.0187×10^{-2})	0.0193 (0.0122)	0.0277 (0.0114)	0.9587	6/100	−138.4176	0.0077
L-BFGS	'fGarch' [22]	2.1770×10^{-4} (1.2895×10^{-3})	0.0217 (0.0149)	0.0396 (0.0214)	0.5114	15/100	−152.2049	0.6934
SQP	'rugarch' [23]	1.8615×10^{-4} (1.3370×10^{-3})	0.0246 (0.0165)	0.0420 (0.0221)	0.6213	0/100	−125.5769	0.1573
L-BFGS		1.5185×10^{-4} (1.1970×10^{-3})	0.0261 (0.0179)	0.0486 (0.0254)	0.6655	27/100	−133.4514	0.1375
SQP	'Econometrics Toolbox' [26]	1.4723×10^{-4} (1.0909×10^{-3})	0.0235 (0.0158)	0.0464 (0.0255)	0.4723	0/100	−153.3256	0.0410
IPM		1.1938×10^{-4} (9.8895×10^{-4})	0.0229 (0.0152)	0.0454 (0.0248)	0.4629	0/100	−153.3228	0.0878
SQP		1.2631×10^{-3} (1.0041×10^{-3})	0.0235 (0.0158)	0.0456 (0.0253)	0.4743	0/100	−153.3022	0.3128
BFGS		1.3165×10^{-3} (2.0715×10^{-3})	0.0240 (0.0119)	0.0263 (0.0129)	0.7535	4/100	−135.5398	0.7194
BMM (proposed)		1.1643×10^{-4} (1.0825×10^{-3})	0.0172 (9.0007×10^{-3})	0.0243 (0.0131)	0.3035	0/100	−153.6315	0.2453

Algorithm 2 BMM for Gaussian MLE of GARCH

Input: Initialization ω, $\boldsymbol{\gamma}$, and $\{\sigma_t^2\}$ for $t = 1, \ldots, N$;
$\{a_t^2\}$ for $t = 1 - q, \ldots, N$,
$\{\sigma_t^2\}$ for $t = 1 - p, \ldots, 0$;
while stopping criteria not met **do**
  update $\omega^\star$ by Eq. (7);
  update $\boldsymbol{\gamma}^\star$ by Algorithm 1;
  update $\left(\sigma_t^2\right)^\star$ for $t = 1, \ldots, N$ by the cubic formula;
end while
Output: ω and $\boldsymbol{\gamma}$

$\left(\omega^{(k)}, \boldsymbol{\gamma}^{(k)}, \{(\sigma_t^2)^{(k)}\}\right)$ *generated by Algorithm 2 converges to the set of stationary points of prob.* (5).

V. NUMERICAL EXPERIMENTS

In this section, we conduct numerical simulations on synthetic data to study the estimation performance of the proposed BMM method with comparison to existing methods. The numerical simulations are performed in both R and MATLAB on a personal computer with a 3.3 GHz Intel Xeon W CPU. The methods in comparison are BHHH in [3], BFGS in tseries package and also implemented by ourselves, L-BFGS in fGarch and rugarch packages, SQP in rugarch package, Econometrics Toolbox and also implemented by ourselves and IPM in Econometrics Toolbox.

We generate a residual series $\{a_t\}$ with $N = 100$ following a GARCH$(2, 3)$ with parameters $\omega_{\mathsf{true}} = 0.01$, $\boldsymbol{\alpha}_{\mathsf{true}} = [0.1, 0.3]^\top$, and $\boldsymbol{\beta}_{\mathsf{true}} = [0.2, 0.25, 0.1]^\top$, where the white noise series $\{\epsilon_t\}$ follows a standardized Gaussian distribution. In total, $M = 100$ datasets are generated. The estimated parameters are compared with true parameters based on the mean square error (MSE) defined as $\mathsf{MSE}(x) = \frac{1}{M}\sum_{i=1}^{M}(x_i - x_{\mathsf{true}})^2$, and the standard error (SE) defined as $\mathsf{SE}(x) = \frac{\sqrt{\sum_{i=1}^{M}(x_i - \bar{x})^2}}{M}$. We also report the number of constraint violation cases, i.e., the cases with the estimated parameters not obeying either the positivity constraint or the stationarity constraint as well as the averaged objective value and averaged computational time.

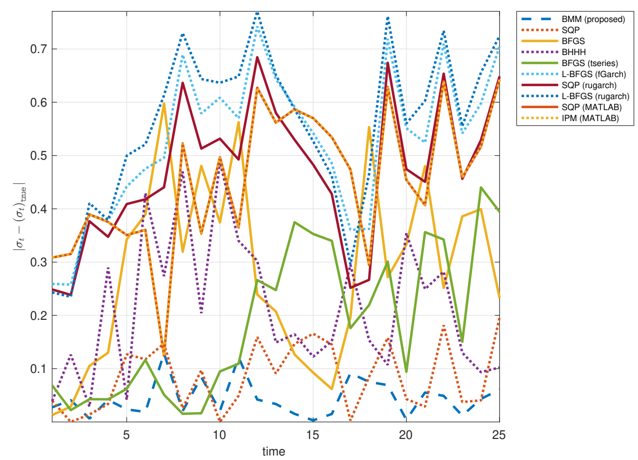

Figure 1. Comparisons of estimated volatility curves with the true volatility.

In Table I, we summarize all the results. It can be seen that the proposed BMM algorithm is able to attain the lowest MSE on the parameters and also volatility σ_t (in this case, $(\sigma_t)_{\mathsf{true}}$ is defined as the volatility obtained by the GARCH model with ω_{true}, α_{true}, and β_{true}). Our algorithm also reaches the lowest objective value among all the methods. This indicates that the efficiency of the proposed algorithm. Meanwhile, we can observe that the quasi-Newton algorithms very often violate the model constraints in the MLE problem. In the following, we visualize the estimated volatility curves from different methods. From Figure 1, the curves describe the distances between σ_t and $(\sigma_t)_{\mathsf{true}}$ from all methods, and the BMM method has a relatively lower $|\sigma_t - (\sigma_t)_{\mathsf{true}}|$ which is also supported by the Table I.

VI. CONCLUSIONS

In this paper, we have proposed a novel algorithm for Gaussian maximum likelihood estimation of a GARCH model. The problem is formulated as a penalized form and solved based on the block majorization-minimization algorithmic framework. Numerical experiments demonstrate the effectiveness of the estimation algorithm. An interesting future direction is to extend the idea of using a penalty form for the parameter estimation of stochastic volatility models.

REFERENCES

[1] R. S. Tsay, *Analysis of Financial Time Series*. John wiley & sons, 2005.

[2] R. F. Engle, "Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation," *Econometrica: Journal of the Econometric Society*, pp. 987–1007, 1982.

[3] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *Journal of Econometrics*, vol. 31, no. 3, pp. 307–327, 1986.

[4] ——, "A conditionally heteroskedastic time series model for speculative prices and rates of return," *The Review of Economics and Statistics*, vol. 69, no. 3, p. 542, 1987.

[5] D. B. Nelson, "Conditional heteroskedasticity in asset returns: A new approach," *Econometrica: Journal of the Econometric Society*, pp. 347–370, 1991.

[6] R. F. Engle, "GARCH 101: The use of ARCH/GARCH models in applied econometrics," *Journal of Economic Perspectives*, vol. 15, no. 4, pp. 157–168, 2001.

[7] T. Bollerslev, "Glossary to ARCH (GARCH)," *CREATES Research Paper*, vol. 49, p. 44, 2008.

[8] Y. Feng and D. P. Palomar, "A signal processing perspective on financial engineering," *Foundations and Trends® in Signal Processing*, vol. 9, no. 1–2, pp. 1–231, 2016.

[9] T. Angelidis, A. Benos, and S. Degiannakis, "The use of GARCH models in VaR estimation," *Statistical Methodology*, vol. 1, no. 1-2, pp. 105–128, 2004.

[10] M. K. P. So and L. H. Y. Philip, "Empirical analysis of GARCH models in value at risk estimation," *Journal of International Financial Markets, Institutions and Money*, no. 2, pp. 180–197, 2006.

[11] Z. Zhao and D. P. Palomar, "Sparse reduced rank regression with nonconvex regularization," in *2018 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE, 2018, pp. 811–815.

[12] ——, "Mean-reverting portfolio with budget constraint," *IEEE Transactions on Signal Processing*, vol. 66, no. 9, pp. 2342–2357, 2018.

[13] Z. Zhao, R. Zhou, and D. P. Palomar, "Optimal mean-reverting portfolio with leverage constraint for statistical arbitrage in finance," *IEEE Transactions on Signal Processing*, vol. 67, no. 7, pp. 1681–1695, 2019.

[14] J. P. Pascual, N. Von Ellenrieder, M. Hurtado, and C. H. Muravchik, "Adaptive radar detection algorithm based on an autoregressive GARCH-2D clutter model," *IEEE Transactions on Signal Processing*, vol. 62, no. 15, pp. 3822–3832, 2014.

[15] I. Cohen, "Modeling speech signals in the time–frequency domain using GARCH," *Signal Processing*, vol. 84, no. 12, pp. 2453–2459, 2004.

[16] H. Kalbkhani, M. G. Shayesteh, and B. Zali-Vargahan, "Robust algorithm for brain magnetic resonance image (MRI) classification based on GARCH variances series," *Biomedical Signal Processing and Control*, p. 11, 2013.

[17] Q. T. Tran, Z. H. Ma, H. C. Li, L. Hao, and Q. K. Trinh, "A multiplicative seasonal ARIMA/GARCH model in EVN traffic prediction," *International Journal of Computer Science and Network Security*, vol. 08, no. 04, pp. 43–49, 2015.

[18] A. Noiboar and I. Cohen, "Two-dimensional GARCH model with application to anomaly detection," in *2005 13th European Signal Processing Conference*. IEEE, 2005, pp. 1–4.

[19] H. T. Pham and B. Yang, "Estimation and forecasting of machine health condition using ARMA/GARCH model," *Mechanical Systems and Signal Processing*, vol. 24, no. 2, pp. 546–558, 2010.

[20] D. B. Nelson and C. Q. Cao, "Inequality constraints in the univariate GARCH model," *Journal of Business & Economic Statistics*, vol. 10, no. 2, pp. 229–235, 1992.

[21] T. Adrian and H. Kurt, *tseries: Time Series Analysis and Computational Finance*, 2020, R package version 0.10.48.

[22] W. Diethelm, *fGarch: Rmetrics - Autoregressive Conditional Heteroskedastic Modelling*, 2017, R package version 3042.83.2.

[23] A. Ghalanos, *rugarch: Univariate GARCH models*, 2020, R package version 1.4.4.

[24] D. B. Nelson, "Stationarity and persistence in the GARCH (1, 1) model," *Econometric Theory*, vol. 6, no. 3, pp. 318–334, 1990.

[25] G. Cornuejols and R. Tütüncü, *Optimization Methods in Finance*. Cambridge University Press, 2006, vol. 5.

[26] MathWorks, "MATLAB Econometrics Toolbox," 2022, the MathWorks, Natick, MA, USA.

[27] M. Razaviyayn, M. Hong, and Z. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013.

[28] Y. Sun, P. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2016.

[29] D. P. Palomar, "Convex primal decomposition for multicarrier linear MIMO transceivers," *IEEE Transactions on Signal Processing*, vol. 53, no. 12, pp. 4661–4674, 2005.