IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 12, 2020, accepted February 25, 2020, date of publication March 9, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2979268

Peak Detection Based on FPGA Using Quasi-Newton Optimization Method for Femtosecond Laser Ranging

YU JIANG[1], QIANG LIU[1], (Member, IEEE), HUI CAO[2], AND YOUJIAN SONG[2]

[1]Tianjin Key Laboratory of Imaging and Sensing Microelectronic Technology, School of Microelectronics, Tianjin University, Tianjin 300072, China
[2]Ultrafast Laser Laboratory, Key Laboratory of Opto-electronic Information Technology, Ministry of Education, School of Precision Instrument and Opto-electronics Engineering, Tianjin University, Tianjin 300072, China

Corresponding author: Qiang Liu (qiangliu@tju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61574099 and Grant 61974102.

ABSTRACT Dual femtosecond laser ranging (DFLR) is an enabling absolute distance measurement technique which is advantageous of high measurement precision, fast update rate, and large unambiguous range. Peak detection, which requires repeated online solution of nonlinear curve fitting, is a key module of the DFLR system and its performance affects the accuracy and real-time of the ranging system. In addition, for long baseline measurements based on satellite-borne platforms, the DFLR system has requirements in high integration, low cost, and small size. This paper presents a peak detection implementation on a field-programmable gate array (FPGA) that employs a Broyden-Fletcher-Goldfarb-Shanno (BFGS) method to handle nonlinear curve fitting. FPGA is used to explore the possibilities of parallel architecture for the acceleration of peak detection, and realize the miniaturization of the system. The detailed architecture design of the peak detection module using BFGS method (PD-BFGS) and two hardware structures are proposed. Then, the PD-BFGS module is applied to a DFLR system and evaluated with experiments on the absolute distance measurement. The experimental results indicate that the PD-BFGS based on FPGA effectively reduces the peak detection error by 42.81%, compared with the peak detection module using Caruana's method. For the DFLR system, the ranging error is reduced by 63.63% and the real-time updating of the ranging results is guaranteed.

INDEX TERMS BFGS-QN method, femtosecond laser ranging, field-programmable gate array (FPGA), nonlinear curve fitting, peak detection.

I. INTRODUCTION

Femtosecond laser pulses greatly improve the precision and speed of time-of-flight (TOF) absolute distance measurement [1], enabling a number of potential applications such as aerospace engineering, large-scale industrial manufacturing and remote sensing [2]. The capability of femtosecond laser ranging can be further advanced by an asynchronous optical sampling (ASOPS) implementation [3]–[6], which uses a pair of femtosecond lasers, signal laser and local oscillator (LO) laser, with an offset repetition frequency and achieves sub-micrometer precision and kHz update rate [7], [8].

In the dual femtosecond laser ranging (DFLR) system, the TOF absolute distance is calculated based on the time interval between the peak temporal positions of the reference pulses and the target pulses originated from the signal laser. To accurately determine the timing of the two peaks, a local oscillator (LO) laser with slightly different repetition frequency has been used to probe the repetitive reference and target pulses. Samples are acquired over many repetitions of the signal, with one sample taken on each repetition, ensembling an sampling oscilloscope. This process equivalently temporally stretch the ultrashort laser pulses, making them accessible to high speed photodetectors. After photodetection, the sampled reference and target pulses need to be reconstructed in electronics [7]. Gaussian fitting, a nonlinear optimization problem, has been used in the pulse reconstruction which has to be finished in the ultrashort

The associate editor coordinating the review of this manuscript and approving it for publication was Jiafeng Xie.

time interval [7]. Currently, there are two solutions to the Gaussian fitting implementation. One solution [9], [10] use Caruana's method, which has low computation complexity and thus fast speed at the cost of accuracy loss. Another solution use complex optimization methods such as Gauss-Newton method [11] and Levenberg-Marquardt method [12] to improve accuracy, but has high computation workload and thus needs high-performance processor to meet the real-time requirement. However, laser ranging systems used on the spaceborne platforms have the cost, size, and energy limits. These require the peak detection module own high accuracy, high performance, low footprint, and low power consumption at the same time. Therefore, this paper proposes an FPGA-based peak detection hardware module using quasi-Newton (QN) method for Gaussian reconstruction.

FPGAs are considered as a promising alternative because of their small size, massive parallelism (compared to digitizing instrument based on central processing units (CPUs)), high reconfigurability (compared to application-specific integrated circuits (ASICs)), and better energy efficiency (compared to graphics processing units (GPUs)) [13]–[18]. By customized design, such as pipelining and parallel computing, FPGA can achieve very high processing speed, providing enough computing power for complex optimization algorithms.

Gauss-Newton method was implemented on FPGA for nonlinear curve fitting in a real-time ranging system [11]. The Gauss-Newton method requires the matrix inversion, which is typically implemented by solving linear equations. Each iteration requires solving linear equations, which becomes a heavy computing burden as the number of iterations increases. The class of QN methods is one of the most effective multi-dimensional unconstrained optimization methods [19]. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) method is one of the most important methods of the QN methods. Among all the other QN variants, BFGS-QN method can effectively avoid the round-off error and the division by zero [20]. BFGS-QN method reduces the computational complexity of the matrix inversion by constructing an approximation of the inverse Hessian matrix. The BFGS-QN method, used to solve the Gaussian fitting problem, has higher fitting accuracy (compared with Caruana's method) and is more convenient to be implemented on FPGA (compared with complex optimization methods). Therefore, the idea of using BFGS-QN method to solve the nonlinear curve fitting problem in the peak detection is promising.

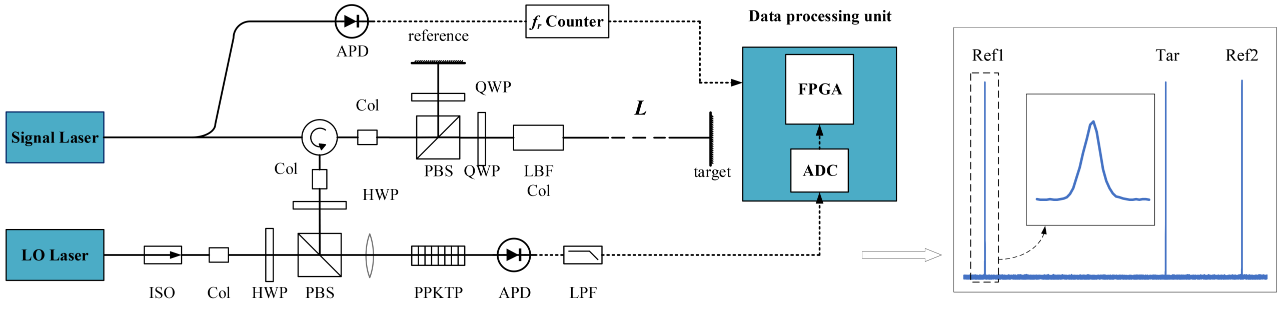
The Gaussian reconstruction using BFGS-QN method is viewed as the most complicated part and an important guarantee for system accuracy and real-time. The main contributions of the presented work are the followings:

1) An online, real-time, and accurate peak detection approach is proposed for the DFLR system to improve the ranging accuracy on FPGA. A pre-processing module is designed to segment the sampled pulse data into noise regions and pulse regions by an adaptive threshold, while BFGS-QN method is used to solve the nonlinear curve fitting problem in the peak detection, to ensure the measurement accuracy.
2) To achieve real-time performance, an inexact line search method is adopted to reduce the computation workload of the BFGS-QN method and customized hardware structure of the BFGS-QN is designed.
3) To trade off hardware resource usage and measurement range, two hardware structures of the peak detection module are proposed. One structure uses two BFGS-QN hardware modules in parallel and has no limitation on the measurement range. The other structure using one BFGS-QN hardware module with properly designed pipeline ensures that interlaced reference pulses and target pulses with interval value, which is greater than the pipeline interval, can be processed in real-time.

The rest of this paper is organized as follows. Section II introduces the principle of DFLR system. The BFGS-QN method used to solve the nonlinear curve fitting problem is introduced in Section III. Section IV presents the hardware implementation details of the proposed PD-BFGS module on FPGA. The experiment results are given in Section V, and conclusions are presented in Section VI.

II. PRINCIPLE OF DUAL FEMTOSECOND LASER RANGING SYSTEM

The DFLR system is composed of three main parts, including two independent and free-running passive mode-locked femtosecond lasers, optical cross-correlation subsystem, and data acquisition and processing subsystem [7]. The DFLR system is shown in Fig. 1 (a). The principle of absolute distance measurement based on dual-comb lasers is to broaden the ranging laser pulse signal through ASPOS and then calculate the distance L from the target-reference TOF (t_{TOF}). As illustrated in Fig. 2, a pair of femtosecond lasers are used as the signal laser and the LO laser, whose repetition frequency are denoted as f_r and $f_r - \Delta f_r$, respectively. There is a repetition frequency difference Δf_r between the signal laser and the LO laser. The signal laser is directed at the reference and the target so that the target TOF is simply determined by the temporal separation of the retro-reflected laser pulses from the reference and the target. The LO is used to sample the retro-reflected signal laser pulse envelopes based on optical cross correlation, which is realized by sum-frequency generation in a PPKTP nonlinear crystal. Considering the slight offset repetition frequency, the LO walks through signal pulses slightly in each repetition period and a full sampling is accomplished in multiple repetition periods. The so-called equivalent sampling process temporally stretch the reference and target pulses by $N = f_r/\Delta f_r$ times. Therefore, the femtosecond pulse timing can be read in a stretched time scale by standard electronics. The temporally-broadened signal is detected by avalanche photo diode (APD) and digitized by analog-to-digital converter (ADC). The equivalent sampling



(a) Dual femtosecond laser ranging system

(b) Equivalent sampling signals and local enlarged figure of *Ref1*

FIGURE 1. Dual femtosecond laser ranging system. ISO (isolator); Col (collimator); LBF Col (large-beam fiber collimator); HWP (half wave plate); QWP (quarter wave plate); PBS (polarization beam splitter); PPKTP (periodically-poled potassium titanyl phosphate); APD (avalanche photo diode); LPF (low pass filter); ADC (analog-to-digital converter).

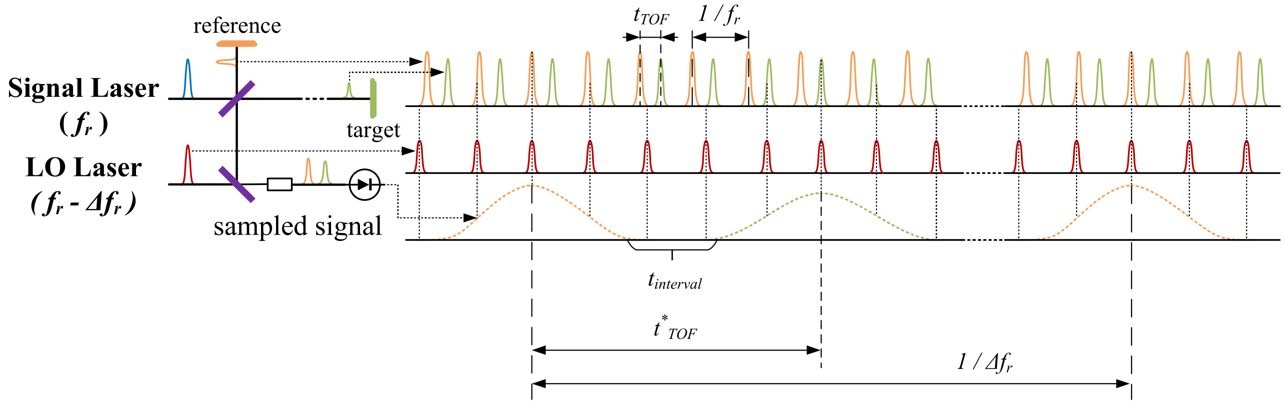


FIGURE 2. Principle of femtosecond laser ranging based on ASOPS.

signals are shown in Fig. 1 (b), where the first pulse is reference pulse (*Ref1*), the second pulse is target pulse (*Tar*) and the third pulse is reference pulse in the next period (*Ref2*). The local enlarged figure of *Ref1* is shown in the middle of Fig. 1 (b), which accords with the Gaussian pulse. In order to calculate the distance between the rangefinder and the target, we only need to measure the target-reference TOF in the stretched time scale (t^*_{TOF}), as well as f_r and Δf_r. The ranging principle can be described by the following equations:

$$L = \frac{c}{2n_g} \cdot t_{TOF} \tag{1a}$$

$$t_{TOF} = \frac{1}{N} \cdot t^*_{TOF} \tag{1b}$$

$$t^*_{TOF} = t_{tar} - t_{ref1} \tag{1c}$$

$$N = \frac{f_r}{\Delta f_r} \tag{1d}$$

$$\Delta f_r = \frac{1}{t_{ref2} - t_{ref1}} \tag{1e}$$

where c is light velocity in a vacuum, n_g is the group refractive index in air, f_r is the repetition frequency of the signal laser, t_{ref1}, t_{tar}, t_{ref2} are the timing of target pulse, reference pulse and the next period reference pulse in the stretched time scale, repectively.

As a result, the distance L to be measured can be calculated by:

$$L = \frac{c}{2n_g} \cdot \frac{1}{f_r} \cdot \frac{t_{tar} - t_{ref1}}{t_{ref2} - t_{ref1}} \tag{2}$$

Equation (2) indicates that, t_{ref1}, t_{tar}, and t_{ref2} (the peak temporal positions of *Ref1*, *Tar*, and *Ref2*) influence the precision of the ranging distance. In order to obtain more accurate peak temporal positions, corresponding to the peak of each Gaussian pulse, it is necessary to fit discrete sampling points based on Gaussian fitting and reconstruct the original Gaussian signal envelope of femtosecond laser pulse. The form of Gaussian function is shown in (3):

$$y(t) = a \cdot exp[-\frac{(t-b)^2}{2c^2}] + d \tag{3}$$

where a, b, c, and d constitute the parameters of the Gaussian function, denoted as $\boldsymbol{p}$, and b is related to the peak temporal position.

$$\boldsymbol{p} = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \tag{4}$$

The Gaussian curve fitting is a nonlinear optimization problem, which is described as follows:

$$\min_{\boldsymbol{p}} E_F(\boldsymbol{p}) \tag{5}$$

with the fitting error function,

$$E_F(\boldsymbol{p}) = \sqrt{\sum_{i=1}^{N_{sp}} \left|\hat{y}(t_i, \boldsymbol{p}) - y_i\right|^2} \tag{6}$$

where N_{sp} is the number of sampling points, (t_i, y_i) represents the i^{th} sampling point. To clearly distinguish between the sampling value and the fitting value, the internal variables of the fitting value are denoted by a hat "ˆ". The purpose of the Gassian curve fitting is to find $\boldsymbol{p}$ such that $\hat{y}(t_i, \boldsymbol{p})$ is close to the real values y_i, and accurately reconstruct the Gaussian envelope.

Caruana's method was used for Gaussian curve fitting [9]. The Gaussian function considered in Caruana's method has no bias, assuming $d = 0$. It calculates the natural logarithm of the data first and then fits the results to a parabola. By doing this, the nonlinear equation is transformed into a linear one. Then linear least square method is used to obtain a, b, and c, thus alleviating its computational complexity. In the targeted DFLR system, the gaussian function is biased and the parameter $d \neq 0$. To enable the linear transformation, parameter d is estimated based on the sampled data. Limited number of sampled data and noises in the data could induce relatively large error in d, leading to inaccurate b. As a result, the accuracy of the peak temporal positions obtained by the Caruana's method still has room for improvement as will be seen in Section IV and Section V. Therefore, a more effective fitting method should be adopted to improve the fitting accuracy. This paper utilizes BFGS-QN method.

III. METHODOLOGY

A. BFGS-QN

The BFGS-QN method can solve the unconstrained nonlinear optimization problem (5). It iteratively updates $\boldsymbol{p}$ ($\boldsymbol{p} \in \mathbb{R}^n$) based on error $E_F(\boldsymbol{p})$ and derivatives $\delta E_F / \delta \boldsymbol{p}$ from an initial point $\boldsymbol{p_0}$ defined as $\left(a^0, b^0, c^0, d^0\right)^{\mathrm{T}}$. The whole process of the BFGS-QN method is summarized in Algorithm 1.

Step 0 estimates the initial point $\boldsymbol{p_0}$. The initial value of a^0 and b^0 correspond to the maximum amplitude of the sampling points on the pulse to be fitted and its temporal position, respectively. The initial value of c^0 is estimated from the width of sampled signal pulses. Since the bias is usually close to 0 but not equal to 0, we set the initial value of d^0 to 0 and correct it through the iterative process. Step 1 computes the search direction $\boldsymbol{d_k}$. Step 2 computes step size λ_k along the search direction by line search methods. Then, Step 3 finds a new $\boldsymbol{p_{k+1}}$ with the search direction and step size, and computes the gradient. In Step 4, if the gradient and the function value satisfy escape conditions, then the iteration process terminates; otherwise, the process continues by updating the approximation of the inverse Hessian matrix $\boldsymbol{B_{k+1}}$ in Step 5. The format of $\boldsymbol{B_{k+1}}$ matrix updating is equivalent to the conventional BFGS equation.

Algorithm 1 BFGS-QN

```
Step 0 Initialization.
    Estimate initial value $p_0 \in \mathbb{R}^n$, $g_0 = \nabla E_F(p_0)$.
    Set $B_0 = I$, $\varepsilon_1 = 1 \times 10^{-7}$, $\varepsilon_2 = 1 \times 10^{-5}$,
    and $k = 0$.
Step 1 Compute search direction $d_k = -B_k g_k$.
Step 2 Compute step size $\lambda_k$ by Armijo-Goldstein
    line search method.
  Step 2.0 Take initial point $\lambda_0$ in $[0, +\infty)$.
      Compare $E_F(p_k)$. Given $\rho \in (0, \frac{1}{2})$, $n > 1$.
      Set $a_0 = 0$, $b_0 = +\infty$, and $t = 0$.
  Step 2.1 If $E_F(p_k + \lambda_t d_k) \leqslant E_F(p_k) + \rho\lambda_t g_k^{\mathrm{T}} d_k$,
      go to Step 2.3.
  Step 2.2 If $E_F(p_k + \lambda_t d_k) \geqslant E_F(p_k) + (1-\rho)\lambda_t g_k^{\mathrm{T}} d_k$,
      stop and output $\lambda_k = \lambda_t$;
      otherwise, set $a_{t+1} = \lambda_t$, $b_{t+1} = b_t$.
      If $b_{t+1} < +\infty$, go to Step 2.3;
      otherwise, set $\lambda_{t+1} = n\lambda_t$, $t = t + 1$,
      go to Step 2.1.
  Step 2.3 Choose a new point.
      Set $\lambda_{t+1} = \frac{a_{t+1} + b_{t+1}}{2}$, $t = t + 1$,
      go to Step 2.1.
Step 3 Update $p_{k+1} = p_k + \lambda_k d_k$, $g_{k+1} = \nabla E_F(p_{k+1})$.
Step 4 Termination test.
    If $\left|E_F(p_{k+1}) - E_F(p_k)\right| \leq \varepsilon_1$
    or $\left\|g_{k+1}\right\| \leq \varepsilon_2$, return $p$.
    otherwise, go to Step 5.
Step 5 Set $s_k = p_{k+1} - p_k$, $z_k = g_{k+1} - g_k$, Update
    $B_{k+1} = B_k - \frac{B_k z_k z_k^{\mathrm{T}} B_k}{z_k^{\mathrm{T}} B_k z_k} + \frac{s_k s_k^{\mathrm{T}}}{z_k^{\mathrm{T}} s_k} + v_k v_k^{\mathrm{T}}$,
    $v_k = \sqrt{z_k^{\mathrm{T}} B_k z_k}\left(\frac{s_k}{z_k^{\mathrm{T}} s_k} - \frac{B_k z_k}{z_k^{\mathrm{T}} B_k z_k}\right)$.
Step 6 Set $k = k + 1$, go to Step 1.
```

B. LINE SEARCH

The line search methods, used to obtain step size λ_k along the search direction, mainly includes exact line search method and inexact line search method. BFGS-QN using Armijo-Goldstein method (BFGS-AG) [21] has been proposed. The Armijo-Goldstein (AG) method is a frequenctly used inexact line search method in practice. The AG method requires less function evaluation, thus greatly reducing the line search time cost to meet the speed requirement of real-time computation. The golden section (GS) method is an exact line search method. BFGS-QN using golden section method is called BFGS-GS in this section.

To compare the AG method with the exact line search method, the BFGS-AG method and the BFGS-GS method are implemented in MATLAB. The data in Table 1 are obtained by the average of 200 independent fitting processes. When the BFGS-QN method is used to solve the problem (5), the fitting performance is mainly influenced by the escape conditions and the number of iterations. As shown in Table 1, when the

TABLE 1. Performance evaluation of BFGS-AG and BFGS-GS. BFGS-AG: BFGS using AG method. BFGS-GS: BFGS using GS method. δ represents the difference of peak temporal positions between BFGS-AG and BFGS-GS. % represents the ratio between δ and the peak temporal position obtained by BFGS-GS.

Number of function evaluations per iteration		Number of iterations		$\delta(\%)$
BFGS_AG	BFGS_GS	BFGS_AG	BFGS_GS	
2.7	16.2	13.6	17.6	0.0002%

escape conditions are set as shown in Step 4 of Algorithm 1, the differences between the peak temporal positions obtained by BFGS-AG and BFGS-GS are about 8.8×10^{-5} ns, which account for about 0.0002% of the peak temporal positions obtained by BFGS-GS. The AG method only needs about 2.7 times evaluations of objective function on average for each line search step, whereas the GS method needs about 16.2 times. At the same time, the number of iterations is approximately 13.6 and 17.6 corresponding to BFGS-AG and BFGS-GS, respectively. Because the peak temporal positions obtained by BFGS-AG and BFGS-GS are almost identical, using the AG method in line search step does not result in additional loss of ranging accuracy. Therefore, the AG method is used as the line search method in this paper as shown in Step 2 of Algorithm 1.

In addition, because the empirical value of the step size λ_k is between 0 and 1, we use 2 to represent $+\infty$ in Step 2.0 by setting $b_0 = 2$ and Step 2.2 by setting $b_{t+1} < 2$ in practice.

C. EVALUATION INDEX

To evalute the peak detection method, we define the peak temporal position error of a single pulse as the peak detection error.

$$R_j = |F_j - A_j| \tag{7}$$

where, R_j denotes the j^{th} peak detection error, F_j represents the j^{th} fitted peak temporal position, and A_j is the j^{th} actual peak temporal position.

Two main indices, the mean peak detection error and the peak detection standard deviation, are used to evaluate performance.

The mean peak detection error is defined as the conformity of the fitted peak temporal positions to the actual value, defined as

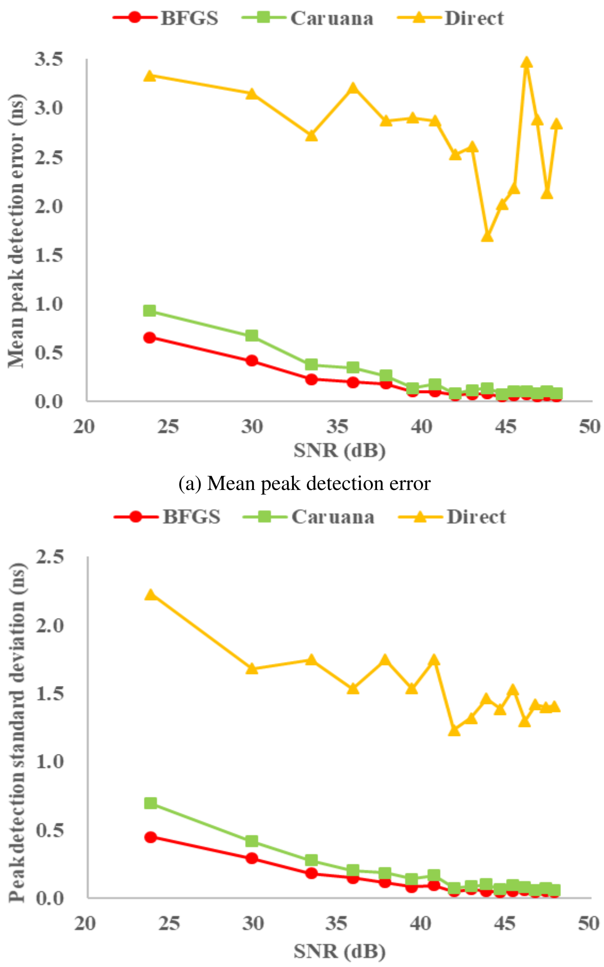
$$\mu_{pd} = \frac{1}{N_p}\sum_{j=1}^{N_p} R_j \tag{8}$$

where, μ_{pd} denotes the mean peak detection error, N_p is the number of detected peaks.

The peak detection standard deviation represents the random error of the peak detection module, calculated from the statistical standard deviation of the peak detection errors as

$$\sigma_{pd} = \sqrt{\frac{1}{N_p}\sum_{j=1}^{N_p}(R_j - \mu_{pd})} \tag{9}$$

where, σ_{pd} denotes the peak detection standard deviation.



(a) Mean peak detection error

(b) Peak detection standard deviation

FIGURE 3. Comparison of the peak detection performance.

D. PERFORMANCE ANALYSIS

To evaluate the performance of the BFGS-QN method, the other two methods are also implemented for peak detection. One is the Caruana's method and the other is the direct method. The direct method uses the temporal position of the maximum sampling point in the sampled data as the peak temporal position without fitting. The three methods are implemented in MATLAB.

In the targeted DFLR system, the amplitudes and signal-to-noise ratios (SNR) of the sampled reference pulses are about 800 mV and 47.8 dB. The sampled target pulses and the sampled reference pulses have similar widths, about 80 ns. The resolution and sample rate of the ADC are 14 bits and

100 MSPS (million samples per second), resulting in approximately 8 sampling points for each pulse fitting [7].

The amplitude range is approximately 50 mV to 800 mV, corresponding to an SNR from 23.8 dB to 47.8 dB. We divide the amplitude range with a step of 50 mV, corresponding to 16 SNRs. According to the characteristics of the sampled signal, 16 waveforms with different SNRs are constructed. Each waveform is randomly sampled 200 times at intervals of 10 ns with different initial sampling positions to produce 200 sets of sampling points. Then the proposed method, the Caruana's method, and the direct method are used to perform the peak detection on each set of the sampling points.

The comparison of the mean peak detection error and the peak detection standard deviation are shown in Fig. 3. While the SNR decreases, the average error and standard deviation of these three methods gradually increase. The mean error and standard deviation of the proposed BFGS method are always lower than the other two methods. As shown in Fig. 3 (a), BFGS method brings up to 42.81% reduction of the mean peak detection error compared with Caruana's method and 97.96% reduction compared with direct method. From Fig. 3 (b), we can see that BFGS method brings up to 44.60% reduction of peak detection standard deviation compared with Caruana's method and 96.83% reduction compared with direct method. The results demonstrate that the BFGS method has good solution performance which will lead to more accurate ranging. Therefore, BFGS method is selected to replace the Caruana's method and the direct method in our design.

In addition, it is worth noting that BFGS is a relatively complex iterative algorithm. While reducing the peak detection error, BFGS increases the execution time. To complete a Gaussian curve fitting on the CPU, the Caruana's method requires 0.18 ms on average, whereas the BFGS method requires 186.14 ms on average. In order to meet the real-time requirement of the DFLR system, we design parallel and pipelined PD-BFGS module on FPGA. In the next, the hardware architecture of the PD-BFGS module is presented.

IV. FPGA HARDWARE IMPLEMENTATION

As shown in Fig. 1, the data processing unit processes the sampled signals in real time, detects the peak temporal positions of each pulse to extract the reference-target TOF, and calculates the distance according to the principle of DFLR, finally sends the distance value.

The FPGA implementation of the data processing unit is shown in Fig. 4. The computation scheduling controller, designed by a finite-state machine (FSM), is used to control the execution sequence of function modules. The phase-locked loop (PLL) module generates the stable clock frequency for other modules. The implementation contains five function modules, including pre-processing (PRE) module, peak detection (PD) module, repetition frequency measurement (RFM) module, distance calculation (DC) module, and universal asynchronous receiver/transmitter (UART) communication interface module.

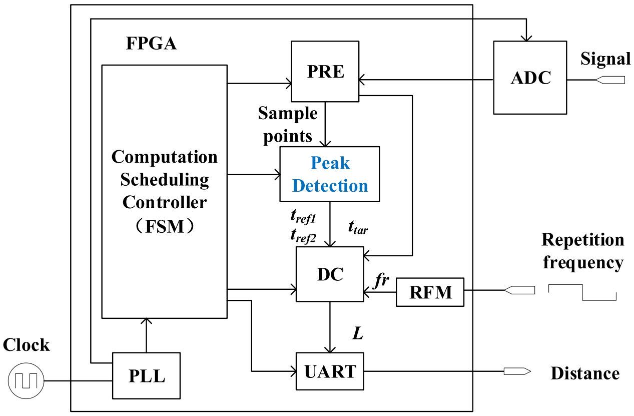


FIGURE 4. Hardware architecture of the data processing unit for DFLR system.

In the targeted DFLR system, most of the sampled data are noise data, and pulse data account for a small proportion. Before peak detection, the PRE module pre-processes the sampled data. The sampled data are segmented into noise regions and pulse regions by an adaptive threshold. A pulse region contains all sampled data of a single pulse higher than the threshold. The data in pulse regions are stored in random access memory (RAM) and used for peak detection. The PD module reads the data in RAM and outputs the peak temporal positions of each pulse. The RFM module updates the current laser repetition frequency (f_r), which is an important parameter in the distance calculation. The DC module calculates the distance results according to (2). The results are sent to the computer through the UART module.

The PRE module contains two blocks. One block dynamically updates the adaptive threshold. The adaptive threshold is updated every 16384 sampled data [22] and calculated as multiplying ξ by the average amplitude of the sampled data. In the case of setting a fixed threshold, if the threshold is set low, noise data will be segmented into pulse regions; if the threshold is set high, most of the effective data in a single pulse will be lost. Both situations will negatively affect the fitting results. The proposed adaptive threshold approach can avoid the above two phenomena by adjusting the threshold according to the noise level, thereby ensuring measurement accuracy. In the experiment, ξ is set to 1.75 based on experimental experience. The other block segments the pulse regions by comparing the threshold with the sampled points. If more than four consecutive sampled data are higher than the threshold, a pulse region is determined, and all consecutive sampled data in this pulse region are stored in RAM. The above two blocks work in parallel. The adaptive threshold is calculated on hardware in pipeline, and the comparison operation needs one clock cycle latency.

In hardware design, the number of sampled data for a single Gaussian curve fitting (η) is fixed. η depends on the sample rate of ADC and the width of the sampled pulses. In the targeted DFLR system, the sample rate of the ADC is 100 MSPS and the width of the sampled pulses is slightly over 80 ns, so the length of the data in a pulse region is about

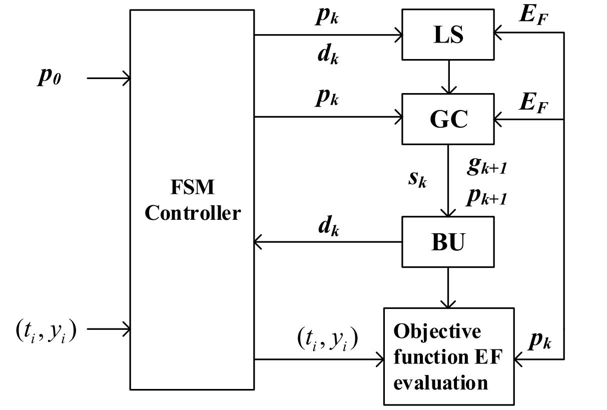


FIGURE 5. Hardware architecture of BFGS-QN solver.

8. As a result, η is set to 8. If the number of sampled data in a pulse region is less than 8, the sampled data in the noise region are selected to make up 8 data. If the number of sampled data in a pulse region is more than 8, the extra effective sampled data, whose amplitude is close to the noise, are discarded.

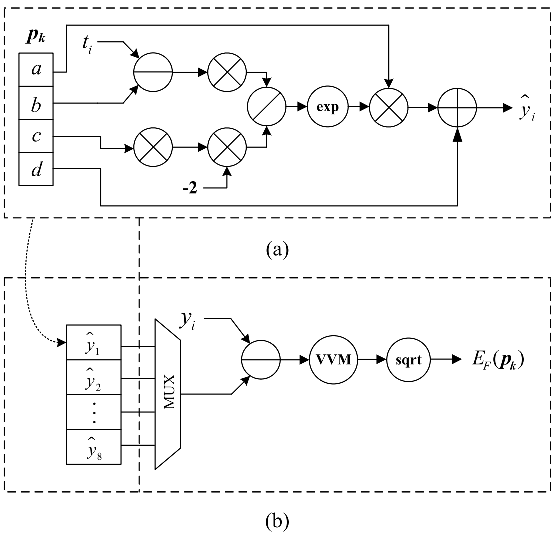
The PD module uses the proposed BFGS-QN method to calculate the peak temporal positions corresponding to the peaks of reference pulses and target pulses. In this section, the proposed BFGS-QN method is implemented as BFGS-QN solver. The details of the BFGS-QN solver and the PD module are described as follows.

A. BFGS-QN SOLVER

According to algorithm 1, the BFGS-QN solver consists of four function modules, which are objective function evaluation module, line search module, gradient calculation module and $\boldsymbol{B_k}$ matrix update module [23]. The FSM controller are designed to control other four function modules in the BFGS-QN solver. The hardware architecture of BFGS-QN solver is shown in Fig. 5. The steps for computing λ_k, $\boldsymbol{g_k}$, and $\boldsymbol{B_k}$ are three computationally intensive parts in the BFGS-QN solver, as well as objective function which is frequently evaluated during λ_k and $\boldsymbol{g_k}$ computation. Therefore, we mainly describe how the four parts are designed and optimized. The overall datapath of the BFGS-QN solver is designed with single-precision floating-point IP cores.

1) OBJECTIVE FUNCTION E_F EVALUATION

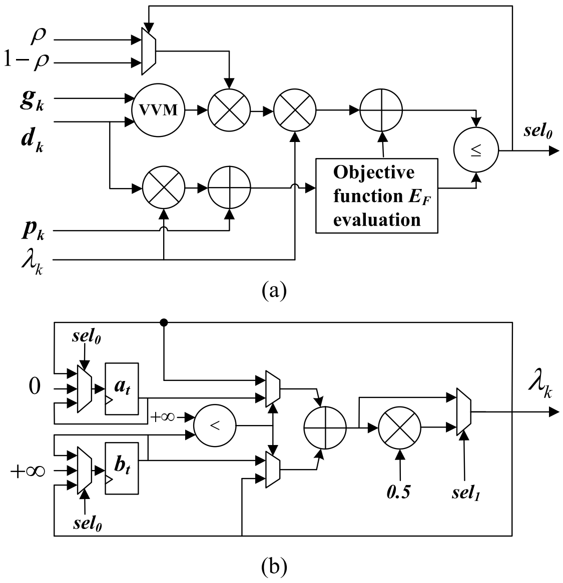
The objective function E_F is evaluated repeatedly during the BFGS-QN solving process. Therefore, we implement the objective function evaluation as a separate block on the hardware. The hardware architecture of objective function E_F is shown in Fig. 6. The block is implemented in two parts: the first part obtains the Gaussian function value according to $\boldsymbol{p_k}$ and the second part computes the fitting error $E_F(\boldsymbol{p_k})$. Fig. 6 (a) shows the design for the first part according to (3). $(t-b)^2$ and $-2c^2$ are calculated in parallel. Fig. 6 (b) shows the second part, which computes the fitting error function according to (6). A deeply pipelined vector-by-vector multiplication (VVM) unit is implemented for the square summation operation. Because the number of sampling points for each fitting operation is limited by the sampling frequency (100 MHz), the first part is duplicated 8 times to increase parallelism, so as to balance the computation time of each part. The function evaluation unit contains 33 multipliers, 9 adders, 9 subtractor, 8 exponents, 8 dividers, and a square rooter operator.



(a)

(b)

FIGURE 6. Hardware architecture of objective function E_F evaluation. (a) Gaussian function $\hat{y}_i$ evaluation; (b) Fitting error $E_F(p_k)$ evaluation.



(a)

(b)

FIGURE 7. Hardware architecture of λ_k computation [21]. (a) Comparison part; (b) λ_k updating part.

2) STEP SIZE λ_k COMPUTATION BY LINE SEARCH (LS) METHOD

As shown in Fig. 7, the design of AG method includes two parts: comparison part and λ_k updating part. Fig. 7 (a) shows the comparison part in Step 2.1 and Step 2.2 in Algorithm 1. After initialization, $f(\boldsymbol{p_k} + \lambda_t \boldsymbol{d_k})$ and $f(\boldsymbol{p_k}) + \rho\lambda_t \boldsymbol{g_k}^{\mathrm{T}} \boldsymbol{d_k}$ are

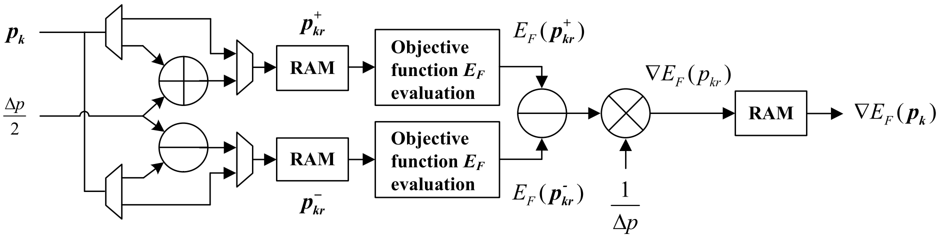


FIGURE 8. Hardware architecture of gradient computation.

calculated in parallel. If sel_0 is 1, $1-\rho$ is selected to participate in the calculation, $f(\boldsymbol{p_k}) + (1-\rho)\lambda_t \boldsymbol{g_k}^{\mathrm{T}} \boldsymbol{d_k}$ is calculated in the same piece of hardware with $f(\boldsymbol{p_k}) + \rho\lambda_t \boldsymbol{g_k}^{\mathrm{T}} \boldsymbol{d_k}$. Fig. 7 (b) shows the λ_k updating part in Step 2.2 and Step 2.3. Different update equations in Steps 2.2 and 2.3 share the same adder and multiplier. All arithmetic operations are designed in pipeline structures to achieve high performance.

3) GRADIENT g_k COMPUTATION (GC)

The gradient of $E_F(\boldsymbol{p_k})$ is calculated by the following equation:

$$\nabla E_F(\boldsymbol{p_k}) = (\nabla E_F(p_{k1}), \nabla E_F(p_{k2}), \nabla E_F(p_{k3}), \nabla E_F(p_{k4}))^{\mathrm{T}} \tag{10}$$

$$\nabla E_F(p_{kr}) = \frac{E_F(\boldsymbol{p^+_{kr}}) - E_F(\boldsymbol{p^-_{kr}})}{\Delta p} \tag{11}$$

$$\boldsymbol{p^+_{kr}} = \left(p_{k1}, p_{kr} + \frac{\Delta p}{2}, p_{k3}, p_{k4}\right)^{\mathrm{T}} \tag{12}$$

$$\boldsymbol{p^-_{kr}} = \left(p_{k1}, p_{kr} - \frac{\Delta p}{2}, p_{k3}, p_{k4}\right)^{\mathrm{T}} \tag{13}$$

where $r = 1, 2, 3, 4$. The pipelined hardware implementation of gradient computation is shown in Fig. 8. $\boldsymbol{p_k}$ is the input vector. Each element p_{kr} of vector $\boldsymbol{p_k}$ add or subtract $\Delta p/2$ respectively, which is controlled by MUXs. $\boldsymbol{p^+_{kr}}$ and $\boldsymbol{p^-_{kr}}$ will be stored in RAMs. $E_F(\boldsymbol{p^+_{kr}})$ and $E_F(\boldsymbol{p^-_{kr}})$ are obtained from the objective function evaluation with concurrent operation, and the results are streamed into a subtractor and a multiplier in sequence. The parallel structure is designed as a tradeoff between speed and resource.

4) B_k MATRIX UPDATING (BU)

The $\boldsymbol{B_k}$ matrix updating (BU) generates an approximate sequence of inverse Hessian by iteration. The block derives an 4×4 matrix, according to Step 5 of BFGS-QN algorithm. The hardware architecture of the BU block is shown in Fig. 9. The VVM unit is heavily used in this module, such as $\boldsymbol{z_k}^{\mathrm{T}}\boldsymbol{s_k}$. The matrix-by-vector multiplication (MVM), which is the most computationally intensive operation, is implemented using multiple VVMs. For ease of illustration, multiplier symbols are used to represent VVMs and MVMs. The intermediate results $\boldsymbol{z_k}$, $\boldsymbol{B_k z_k}$, and $((\boldsymbol{s_k}/\boldsymbol{z_k}^{\mathrm{T}}\boldsymbol{s_k}) - (\boldsymbol{B_k z_k}/\boldsymbol{z_k}^{\mathrm{T}}\boldsymbol{B_k z_k}))$ are buffered in on-chip RAM units for data reuse in subsequent computation steps. At one input of the last adder, a first-input-first-output (FIFO) is used to buffer the result of $\boldsymbol{v_k v_k}^{\mathrm{T}}$ and wait for the results of other sub-items in the update equation. The BU block updates the $\boldsymbol{B_{k+1}}$ matrix row by row.

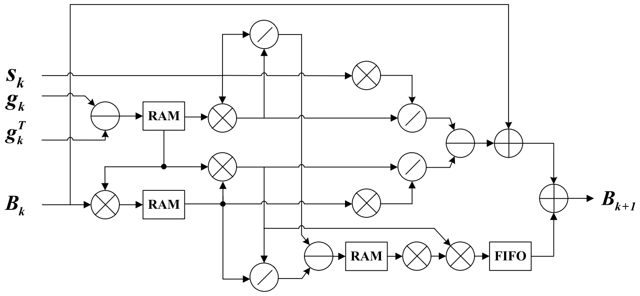
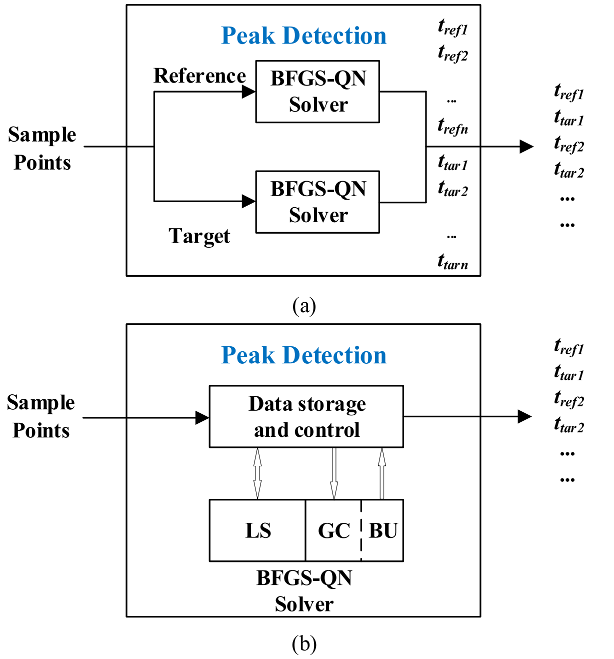


FIGURE 9. Hardware architecture of B_k Matrix Updating [23].



(a)

(b)

FIGURE 10. Two hardware structures of Peak Detection module. (a) Structure 1; (b) Structure 2.

B. TWO HARDWARE STRUCTURES OF THE PEAK DETECTION MODULE

In the targeted DFLR system, the sampled signal is a reference pulse and target pulse interlaced signal with a certain time interval $t_{interval}$, as shown in Fig. 2. Due to the different distance ranges, $t_{interval}$ is not fixed. If $t_{interval}$ is shorter than the time required to fit a curve, BFGS-QN solver cannot

TABLE 2. Clock cycles of BFGS-QN solver.

	#clock cycles per iteration (BFGS)
LS	1460
GC	864
BU	566
BFGS	2890

guarantee to complete the peak detection of the reference pulse (target pulse) before the arrival of the target pulse (reference pulse). Therefore, two hardware structures of the PD-BFGS module are proposed to ensure that interlaced reference and target pulses can be processed in real-time.

Structure 1: Two BFGS-QN solvers process the reference and target signal pulses in parallel. The structure is shown in Fig. 10 (a). The first BFGS-QN solver processes the reference signal pulse, and the second BFGS-QN solver processes the target signal pulse, respectively. This structure can handle any values of the time interval between two pairs of pulses and theoretically does not have limitation on the measurement range of the system. However, the disadvantage is that two BFGS-QN solvers introduce large resource overhead.

Structure 2: One BFGS-QN solver is used to process the reference and target signal pulses in a pipeline. The structure is shown in Fig. 10 (b). From Table 2, we can see that the number of execution clock cycles of the LS block is roughly equal to the sum of the GC and BU blocks. Therefore, the BFGS-QN solver can be executed in a two-stage pipeline. The data storage and control block with simple logics stores intermediate data and controls the pipeline. When $t_{interval}$ is larger than the pipeline interval (the execution time of the LS block), the second stage (GC+BU) processes the reference pulse (target pulse) and the first stage (LS) processes the target pulse (reference pulse) at the same time.

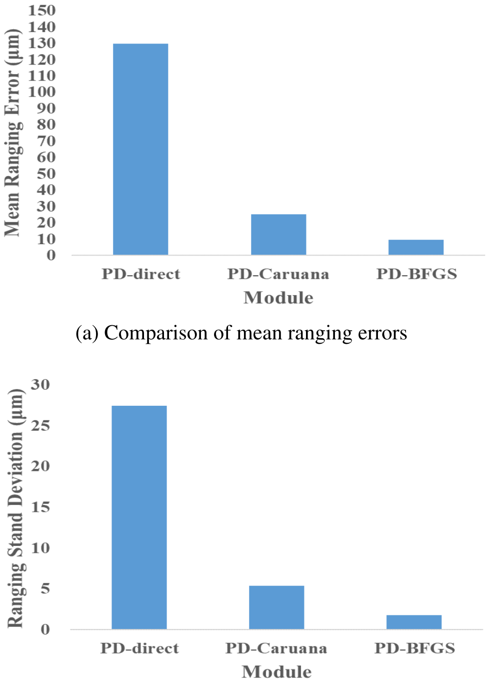
Although Structure 2 reduces hardware resources, it is applicable only when $t_{interval}$ is greater than the pipeline interval. When $t_{interval}$ is less than the pipeline interval, Structure 1 is used. Therefore, which structure to adopt depends on the actual application situation and makes tradeoff between hardware resources and measurement range.

V. EXPERIMENTAL RESULTS

To evaluate the proposed PD-BFGS module, the design is synthesized and implemented on the Net-FPGA SUME (xc7vx690tffg1761-3) board. The repetition frequency (f_r) of the signal laser is about 73.6 MHz, and the repetition frequency difference Δf_r is about 2 KHz [7], [8]. Therefore, the period of the sampled reference pulse and target pulse is 500 μs, respectively.

There is a large-beam fiber collimator placed in the optical unit of the DFLR system, so that the amplitude (and SNR) of the system echo signal at different target distances are approximately equal. In other words, the difference of target distances has little effect on ranging accuracy [24]. Limited by the experimental conditions, we place the target at a fixed position within the measuring range of the DFLR system. The data of this distance is used as a representative for the experiments. The reference distance is 63.294006 m, and sampling points within 300 ms are used in the experiment.

The reference distance is obtained by a calibrated distance ranger. The reference ranger is similar with the proposed DFLR system in Fig. 1. The only difference between the reference ranger and the DFLR system is that the data processing unit of the reference ranger is based on a custom-built digitizer with LabVIEW system. The reference ranger is calibrated by the interferometers at NIM (National Institute of Metrology of China). The measurement residual of the reference ranger is less than 2 μm [24]. In the DFLR system, FPGA is used as the data processing unit with the size much smaller than the custom-built digitizer, so that the system can be integrated in a small volume.



(a) Comparison of mean ranging errors

(b) Comparison of ranging standard deviations

FIGURE 11. The ranging results with three peak detection modules.

A. PERFORMANCE

The performance of the DFLR system based on the proposed PD-BFGS module is evaluated first. The peak detection module using the Caruana's method (PD-Caruana) and the peak detection module using the direct method (PD-direct) are also implemented for comparison.

The ranging results with the three peak detection modules are shown in Fig. 11. The mean value of the ranging results with PD-direct is 63.293876 m, the error is 129.63 μm, and the standard deviation is 27.43 μm; the mean value of the ranging results with PD-Caruana is 63.294031 m, the error is 24.98 μm, and the standard deviation is 5.37 μm; the mean value of the ranging results with PD-BFGS is 63.293997 m, the error is 9.08 μm, and the standard deviation is 1.74 μm.

TABLE 3. Execution time of BFGS-QN solver.

	execution time (μs) per iteration (BFGS) @250 MHz	#iteration (BFGS)	execution time (μs) @250 MHz
LS	5.84	\	\
GC	3.46	\	\
BU	2.26	\	\
BFGS	11.56	15	173.40

TABLE 4. Resource comparison of Structure 1 and Structure 2 of PD-BFGS implementation. (%): Ratio between the used resources and the total available resources.

	LUT(%)	FF(%)	DSP(%)	BRAM(%)
Structure1	159769 (36.88%)	193802 (22.37%)	1210 (33.61%)	57 (3.88%)
Structure2	82661 (19.08%)	99475 (11.48%)	606 (16.83%)	31.5 (2.14%)

PD-BFGS reduces the error of the ranging results by 92.99% compared with the PD-direct, and by 63.63% compared with the PD-Caruana. At the same time, PD-BFGS reduces the standard deviation of the ranging results by 93.66% compared with the PD-direct, and by 67.60% compared with the PD-Caruana. With PD-BFGS module, the ranging performance of the DFLR system is considerably improved.

B. EXECUTION TIME

As shown in Table 3, the designed PD-BFGS module works at 250 MHz and obtains the peak temporal position of the detected pulse within 173.40 μs on average. When $t_{interval}$ is larger than 5.84 μs, Structure 2 is used to save hardware resources; when $t_{interval}$ is less than 5.84 μs, Structure 1 is used to ensure peak detection of reference and target pulses simultaneously. The peak detection is completed in real-time and its update rate is high enough for the upper computer to carry out data post-processing. The update rate of ranging results is 2 KHz, equals to the repetition frequency difference Δf_r just as the ranging principle indicates. The results indicate that the data processing unit based on PD-BFGS can achieve satisfactory performance and obtain the measurement distance in real-time.

In addition, each block of the designed PD-BFGS module uses parallel and pipeline designs to accelerate the execution time. The parallel structure can be appropriately adjusted in exchange for a decline in hardware resource utilization. For example, the Gaussian function $\hat{y}_i$ evaluation part, which is duplicated 8 times in objective function E_F evaluation block, could be duplicated 4 times and executed twice to save hardware resources.

C. HARDWARE RESOURCE UTILIZATION

The PD-BFGS hardware implementation is customized with the proposed optimization structures for the ranging system. The hardware resource utilization results are reported in Table 4, including lookup table (LUT), flip-flop (FF), digital signal processing (DSP) blocks, and block random access memory (BRAM) blocks. From Table 4, we can see that Structure 2 saves nearly half of the hardware resources compared to Structure 1.

VI. CONCLUSION

This paper has demonstrated the successful implementation of an FPGA-based PD-BFGS module. To better fit with the parallel architecture of FPGA, the BFGS-QN method is used as the embedded optimization algorithm. The designed PD-BFGS module is applied to the DFLR system to fulfill its high requirements on accuracy and real-time. The detailed implementation architecture of the PD-BFGS module and two hardware structures are given. With the help of parallel architecture of FPGA and pipeline design, the PD-BFGS module solves quickly and provides the solutions within the available sampling interval. The experimental results show that the PD-BFGS module based on FPGA has good computational performance, achieves high accuracy and real-time measurement for DFLR system. Compared with the peak detection using the Caruana's method, the mean error of ranging results is reduced by 63.63%, the standard deviation of ranging results is reduced by 67.60%. The design PD-BFGS module is equally effective for other nonlinear curve functions with more unknown parameters. It has a broader application prospect.

REFERENCES

[1] J. J. Fontaine, J.-C. Diels, C.-Y. Wang, and H. Sallaba, "Subpicosecond-time-domain reflectometry," *Opt. Lett.*, vol. 6, no. 9, pp. 405–407, Sep. 1981.

[2] M. E. Pritchard and M. Simons, "A satellite geodetic survey of large-scale deformation of volcanic centres in the central andes," *Nature*, vol. 418, no. 6894, pp. 167–171, Jul. 2002.

[3] I. Coddington, W. C. Swann, L. Nenadovic, and N. R. Newbury, "Rapid and precise absolute distance measurements at long range," *Nature Photon.*, vol. 3, no. 6, pp. 351–356, Jun. 2009.

[4] T.-A. Liu, N. R. Newbury, and I. Coddington, "Sub-micron absolute distance measurements in sub-millisecond times with dual free-running femtosecond er fiber-lasers," *Opt. Express*, vol. 19, no. 19, pp. 18501–18509, Sep. 2011.

[5] J. Lee, S. Han, K. Lee, E. Bae, S. Kim, S. Lee, S.-W. Kim, and Y.-J. Kim, "Absolute distance measurement by dual-comb interferometry with adjustable synthetic wavelength," *Meas. Sci. Technol.*, vol. 24, no. 4, Apr. 2013, Art. no. 045201.

[6] H. Zhang, H. Wei, X. Wu, H. Yang, and Y. Li, "Absolute distance measurement by dual-comb nonlinear asynchronous optical sampling," *Opt. Express*, vol. 22, no. 6, pp. 6597–6604, Mar. 2014.

[7] H. Shi, Y. Song, F. Liang, L. Xu, M. Hu, and C. Wang, "Effect of timing jitter on time-of-flight distance measurements using dual femtosecond lasers," *Opt. Express*, vol. 23, no. 11, pp. 14057–14069, Jun. 2015.

[8] H. Cao, Y. Song, Y. Li, R. Li, H. Shi, J. Yu, M. Hu, and C. Wang, "Reduction of moving target Time-of-Flight measurement uncertainty in femtosecond laser ranging by singular spectrum analysis based filtering," *Appl. Sci.*, vol. 8, no. 9, p. 1625, 2018.

[9] R. A. Caruana, R. B. Searle, T. Heller, and S. I. Shupack, "Fast algorithm for the resolution of spectra," *Anal. Chem.*, vol. 58, no. 6, pp. 1162–1167, May 1986.

[10] H. Guo, "A simple algorithm for fitting a Gaussian function [DSP tips and Tricks]," *IEEE Signal Process. Mag.*, vol. 28, no. 5, pp. 134–137, Sep. 2011.

[11] X. Xie, L. Xu, Z. Wang, and X. Li, "Real-time *in situ* laser ranging based on online echo waveform fitting," *IEEE Sensors J.*, vol. 19, no. 20, pp. 9255–9262, Oct. 2019.

[12] F. Song, W. Sun, J. Wei, M. Jiang, L. Zhang, F. Zhang, Q. Sui, and Y. Tian, "The optimization study of FBG Gaussian fitting peak-detection based on Levenberg-Marquardt algorithm," in *Proc. Chin. Autom. Congr. (CAC)*, Oct. 2017, pp. 3723–3727.

[13] K. Ni, M. Xu, Q. Zhou, H. Dong, X. Li, and G. Wu, "Implementation of a data processing platform for real-time distance measurement with dual-comb lasers," in *Proc. Int. Conf. Opt. Instrum. Technol., Optoelectron. Meas. Technol. Syst.*, Aug. 2015, Art. no. 96231H.

[14] K. Ni, L. Wang, Q. Zhou, X. Li, H. Dong, and X. Wang, "Data acquisition and processing platform in the real-time distance measurement system with dual-comb lasers," *Proc. SPIE*, vol. 10026, Nov. 2016, Art. no. 1002619.

[15] H. Guo, H. Chen, F. Xu, F. Wang, and G. Lu, "Implementation of EKF for vehicle velocities estimation on FPGA," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 3823–3835, Sep. 2013.

[16] N. K. Quang, N. T. Hieu, and Q. P. Ha, "FPGA-based sensorless PMSM speed control using reduced-order extended Kalman filters," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 6574–6582, Dec. 2014.

[17] A. Cilardo, "New techniques and tools for application-dependent testing of FPGA-based components," *IEEE Trans Ind. Informat.*, vol. 11, no. 1, pp. 94–103, Feb. 2015.

[18] N. Roshandel Tavana and V. Dinavahi, "A general framework for FPGA-based real-time emulation of electrical machines for HIL applications," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2041–2053, Apr. 2015.

[19] A. Antoniou and W.-S. Lu, *Practical Optimization—Algorithms and Engineering Applications*. Berlin, Germany: Springer, 2007.

[20] I. T. Christou, "A review of optimization methods," in *Quantitative Methods in Supply Chain Management: Models and Algorithms*. Berlin, Germany: Springer, 2012, pp. 1–138.

[21] J. Liu and Q. Liu, "Speed and resource optimization of BFGS quasi-Newton implementation on FPGA using inexact line search method for neural network training," in *Proc. Int. Conf. Field-Program. Technol. (FPT)*, Dec. 2018, pp. 362–365.

[22] K. Ni, H. Dong, Q. Zhou, M. Xu, X. Li, and G. Wu, "Interference peak detection based on FPGA for real-time absolute distance ranging with dual-comb lasers," in *Proc. Int. Conf. Opt. Instrum. Technol., Optoelectron. Meas. Technol. Syst.*, vol. 9623, Aug. 2015, Art. no. 96231G.

[23] Q. Liu, J. Liu, R. Sang, J. Li, T. Zhang, and Q. Zhang, "Fast neural network training on FPGA using quasi-newton optimization method," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 8, pp. 1575–1579, Aug. 2018.

[24] Y. Li, Y. Cai, R. Li, H. Shi, H. Tian, M. He, Y. Song, and M. Hu, "Large-scale absolute distance measurement with dual free-running all-polarization-maintaining femtosecond fiber lasers," *Chin. Opt. Lett.*, vol. 17, no. 9, 2019, Art. no. 091202.

YU JIANG received the B.Eng. degree in electronic science and technology (microelectronics) from the University of Electronic Science and Technology of China, Chengdu, China, in 2016. He is currently pursuing the M.E. degree in integrated circuit engineering with Tianjin University, Tianjin, China. His research interests include digital integrated circuit design and femtosecond laser absolute distance measurement.

QIANG LIU (Member, IEEE) received the Ph.D. degree from the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K., in 2008. From 2009 to 2011, he was a Research Associate with the Department of Computing, Imperial College London. He is currently a Professor with the School of Microelectronics, Tianjin University. His research interests include digital integrated circuit design, high speed and low power circuit system design, VLSI design optimization, and reconfigurable computing.

HUI CAO received the B.S. degree in electronic science and technology (optoelectronics) from Tianjin University, Tianjin, China, in 2012, where he is currently pursuing the Ph.D. degree in optoelectronics technology. His research interests include femtosecond laser absolute distance measurement and singular spectrum analysis.

YOUJIAN SONG received the B.S. and Ph.D. degrees in optical engineering from Tianjin University, Tianjin, China, in 2004 and 2009, respectively. From 2009 to 2011, he was a Postdoctoral Researcher with the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea. He joined the Faculty of the School of Precision Instruments and Opto-electrical Engineering, Tianjin University, in 2011, where he is currently an Associate Professor. His research interests include the pulse dynamics and timing jitters in ultrafast fiber lasers, as well as ultrafast fiber laser-based precision metrology. He is a Senior Member of Optical society of America (OSA).