# Centralized System Identification of Multi-Rail Power Converter Systems using an Iterative Decimation Approach

Jin Xu, Matthew Armstrong, Maher Al-Greer

***Abstract*—This paper presents an iterative decimation approach to significantly alleviate the computational burden of centralized controllers applying real-time recursive system identification algorithms in multi-rail power converters. The proposed approach uses an adaptive update rate as opposed to the fixed update rate used in conventional adaptive filters. Also, the step size/forgetting factors vary at different iteration stages. As a result, a reduced computational burden and faster model update can be achieved. Besides, recursive algorithms, such as Recursive Least Square (RLS), Fast Affine Projection (FAP) and Kalman Filter (KF), contain two important updates per iteration cycle; Covariance Matrix Approximation (CMA) update and Gradient Vector (GV) update. Usually, the CMA update requires the greater computational effort than the GV update. Therefore, in circumstances where the sampled data in the regressor does not experience significant fluctuations, re-using the CMA, calculated from the last iteration cycle for the current update can result in computational cost savings for real-time system identification. In this paper, both iteration rate adjustment and CMA re-cycling are combined and applied to simultaneously identify the power converter models in a three-rail power conversion architecture.**

***Index Terms*—Adaptive Filters, Digital Model of DC-DC Converters, Iteration Frequency, Parameter Estimation, System Identification, Recursive Least Squares**

## I. Introduction

Multi-rail power converter architectures are commonly used in distributed power supply applications to convert a single voltage supply rail to multiple regulated voltage levels via several Point of Load (POL) converters [1, 2]. Although multi-rail DC-DC power converters have been employed in computing and communication equipment [3, 4], electric vehicles [5], and DC micro-grids [6], their main application fields are to provide low voltages with high power density to downstream devices including microprocessors, FPGAs and their peripherals [7]. Fig.1 [8] shows a typical multi-rail power supply product (TPS653850-Q1 from Texas Instruments) applied to power microcontrollers and their peripherals. A similar type of product can be seen in Analog Devices, such as LT8602 [9]. For carrying out these applications in various working conditions of Switch Mode Power Converters (SMPCs), robustness to system changes is important [10-13]; The operation of SMPCs may suffer unexpected or periodic load changes, abrupt disturbances, gradual capacitor degradations, sudden malfunction of circuit components and occasional additions of paralleled output capacitors [14], etc. As these cases may happen concurrently, it is difficult to homogenize parameter change rates to specific numbers and the odds of system parameter changes become highly random. Therefore, a robust control loop, used to cope with these randomly happened system variations, is required; In such cases, control parameters may need to be adjusted in real-time (adaptive control) to minimize the impact of these system variations and achieve optimal regulation. Such controller tuning, for example, adjusting three gains in a PID controller ($kp, ki, kd$) based on information received from a real-time model of the plant, is normally based on online system identification [15].

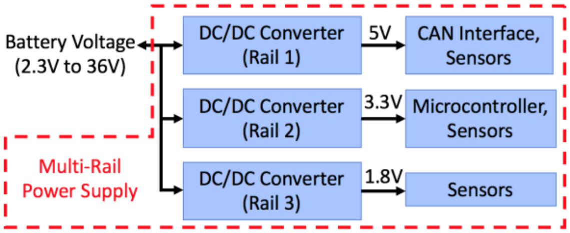


Fig.1. Multi-Rail Power Conversion Architecture for Powering Microcontrollers and Its Peripherals [8].

System identification can be achieved by estimating the model parameters of the power converters (parametric methods) or analyzing the system frequency response (non-parametric methods) [16-21]. Non-parametric methods usually need open-loop control, transient response acquisition, and off-line analyses [22-24], while parametric methods, which use algorithms with particular application in areas such as adaptive control, may be achieved during closed-loop operation. The performance of algorithms can be judged by Convergence Time, Computational Costs, and Estimation Accuracy [25]. Literature shows that variants of the RLS algorithm [16, 26-28] are widely used in power converter applications [29, 30]. For instance, the Dichotomous Coordinate Descent (DCD)-RLS is shown to be more computationally efficient than classical RLS [29, 31]. In [32], a variable-forgetting factor method,

J. Xu, M. Armstrong, are with the School of Engineering, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK (email: j.xu19@ncl.ac.uk; matthew.armstrong@ncl.ac.uk).
M. Al-Greer is with School of Computing, Engineering and Digital Technologies, Teesside University, Tees Valley, TS13BX, UK, (m.al-greer@tees.ac.uk).

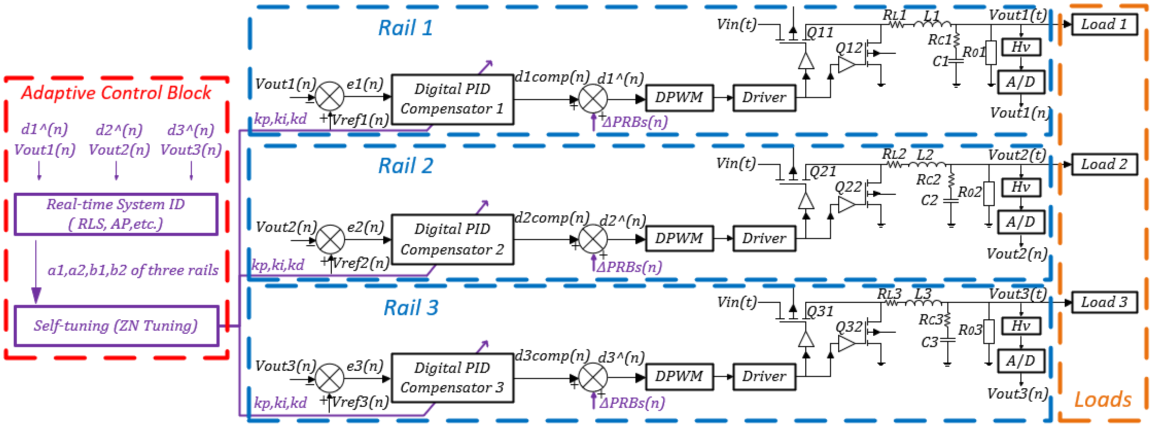


Fig.2. Three-rail Power Converter Architecture (blue, right side) and Real-time System Identification Process (red, left side).

computationally heavier than the classical RLS though, is proposed to improve tracking of real-time parameter variations. A Kalman Filter (KF) approach has also been used in single-rail converter applications [32]. The KF is demonstrated to have advantages in dealing with abrupt load changes, but again computational effort is an issue. Besides, a Fast Affine Projection (FAP) algorithm was proposed [16]. Results show that FAP performs better than RLS in terms of the convergence time, the estimation accuracy, and the computational cost. To further alleviate the computational burden of parameter estimation, recently the authors in [27] proposed a Step-adaptive Approximation Least Squares (SALS) for high-frequency estimation of a single-rail buck converter.

According to pieces of literature, the recursive algorithms perform well in real-time parameter estimation for single power converters. However, in multi-rail architectures with a centralized single controller, the computational burden will become heavy, increasing proportionately with the addition of rails. For example, if the available computation time is 50μs, the employed processor should finish 64 additions, 109 multiplications, and 1 division in 50μs for single-rail parameter estimation by using RLS (see Table II). If three rails are simultaneously identified, the computational burden in the 50μs will be increased to 192 additions, 327 multiplications, and 3 divisions. The significant increase in the computational burden could cause the need for advanced processors more computationally capable particularly, resulting in extra investments. As such, this paper considers two approaches to reduce the computational complexity of multi-rail converters and better facilitate centralized single processor control. These experimentally validated approaches are 1. Iteration frequency reduction. 2. Update frequency reduction of Covariance Matrix Approximation (CMA) by re-using CMA.

The RLS and KF algorithms are employed to experimentally validate the proposed solutions which can be more widely applicable to other recursive algorithms though.

## II. Parameter Estimation of Multi-Rail Power Converters

### A. *Modeling and Parameter Estimation of Buck Converter*

Typically, a DC-DC buck converter can be modeled by a small-signal-average model transfer function [33]. Here, the control (duty cycle, $d$) – to – voltage output ($v$) transfer function is well-reported and can be expressed as follows [34]:

$$\frac{v(s)}{d(s)} = \frac{V_{in}(CR_C s + 1)}{s^2 LC\left(\frac{R + R_C}{R + R_L}\right) + s\left(R_C C + C\left(\frac{RR_L}{R + R_L}\right) + \frac{L}{R + R_L}\right) + 1} \quad (1)$$

Here, $V_{in}$ is the input voltage, $C$ is the output capacitor, $L$ is the inductor, $R$ is the resistance load, $R_C$ is the capacitance Equivalent Series Resistance (ESR), $R_L$ is inductance ESR. Applying zero-order-hold discretization, (1) can be written in the digital form as:

$$\frac{V(z)}{D(z)} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (2)$$

In (2), $a_1, a_2, b_1$, and $b_2$ are four coefficients that should be estimated. $V$ is the voltage output and $D$ the duty cycle.

### B. *RLS for Parameter Estimation*

Fig.2 shows the real-time parameter estimation process of a three-rail power converter. By superimposing a small frequency rich perturbation signal, here Pseudo-Random Binary Sequence (PRBS) [17], the resultant duty cycles, $d^{\wedge}(n)$, will momentarily excite the output voltage of the corresponding power rail. During this time, the applied algorithm samples and processes the duty cycle and the output voltage signals to estimate the parameters of the transfer function in (2) ($a_1$, $a_2$, $b_1$, and $b_2$).

Many adaptive filters used for parameter estimation are based on Regularized Newton's recursions, such as Least Mean Squares (LMS), RLS and Affine Projection (AP), etc., which update New Guess (NG) through adding the Correction Term (CT) calculated in the current iteration cycle to the NG gotten from the last cycle. CT contains a direction part (GV) and a magnitude part (CMA). Therefore, updating an NG typically includes 5 update steps:

1. Regressor: the sampled duty cycle and voltage signals.

2. Covariance Matrix Approximation (CMA): the magnitude part of CT.

3. Gradient Vector (GV): the direction part of CT.

4. Correction Term (CT): the difference between the last NG and the current one.

5. New Guess (NG): updated results.

RLS, as a typical algorithm derived from Regularised Newton's recursion, its 5 steps are presented in Table I and the computational complexity of each step in Table II [35].

According to Fig.2, converters in the multi-rail architecture are independent of each other and there are no master converter or secondary converters, system variations and instabilities (load changes, component failures, etc.) in one rail won't affect other rails at all. Therefore, rails are all estimated separately; they have their individual CMA, GV, CT and NG, the identification-related parameters of one converter won't affect other converters.

TABLE I
THE UPDATE SEQUENCE OF AN ITERATION CYCLE OF RLS

| Step | Updates | RLS Formula |
|---|---|---|
| **1** | Regressor | $u_i \triangleq [-V(n-1)\ -V(n-2)\ D(n-1)\ D(n-2)]$<br>$y(i) \triangleq V(n)$ |
| **2** | CMA | $P_i = \lambda^{-1}\left[P_{i-1} - \frac{\lambda^{-1}P_{i-1}u_i{}^{*}u_iP_{i-1}}{1+\lambda^{-1}u_iP_{i-1}u_i{}^{*}}\right]$ |
| **3** | GV | $e_i = u_i{}^{*}[y(i) - u_i\omega_{i-1}]$ |
| **4** | CT | $P_i \cdot e_i$ |
| **5** | NG | $\omega_i = \omega_{i-1} + P_ie_i$ |

$\lambda$ is forgetting factor, $0 \ll \lambda \leq 1$. $\omega \triangleq [a_1\ a_2\ b_1\ b_2]^T$. $i$ is the current iteration instant and $n$ the current sampling instant.

TABLE II
THE COMPUTATIONAL COST OF RLS PER ITERATION CYCLE

| **Ste p** | **Updates** | **Computational Complexity** | | |
|---|---|---|---|---|
| | | + | × | / |
| **1** | Regressor | | | |
| **2** | CMA | $3M^2 - M, (44)$ | $5M^2 + M + 1, (85)$ | 1 |
| **3** | GV | $^{a}M, (4)$ | $2M, (8)$ | |
| **4** | CT | $M^2 - M, (12)$ | $M^2, (16)$ | |
| **5** | NG | $M, (4)$ | | |
| | **In Total** | $4M^2, (64)$ | $6M^2 + 3M + 1, (109)$ | 1 |

[a] $M$ is the number of transfer function coefficients.

## C. *KF for Parameter Estimation*

Different from RLS, the KF algorithm is not derived from Regularized Newton's recursions, which therefore include an additional update step: Kalman Gain (KG) update (Step 3 in Table III listing the six update steps for acquiring NG). Table IV shows the computational costs of each step.

TABLE III
THE UPDATE SEQUENCE OF AN ITERATION CYCLE OF KF

| Step | Updates | Kalman Filter Formula |
|---|---|---|
| **1** | Regressor | $u_i \triangleq [-V(n-1)\ -V(n-2)\ D(n-1)\ D(n-2)]$<br>$y(i) \triangleq V(n)$ |
| **2** | CMA | $P_i = P_{i-1}(\boldsymbol{I} - G_{i-1}u_i) + \hat{E}_i$ |
| **3** | KG | $G_i = P_iu_i{}^{*}[u_iP_iu_i{}^{*} + r]^{-1}$ |
| **4** | GV | $e_i = [y(i) - u_i\omega_{i-1}]$ |
| **5** | CT | $G_i \cdot e_i$ |
| **6** | NG | $\omega_i = \omega_{i-1} + G_ie_i$ |

$\hat{E}_i = diag\left[[\hat{a}_1(i-1)]^2;\ [\hat{a}_2(i-1)]^2;\ [\hat{b}_1(i-1)]^2;\ [\hat{b}_2(i-1)]^2\right]$. $\boldsymbol{I}$ is an $M \times M$ identity matrix. $r$, a scalar, is the observation noise variance, $r > 0$. $\omega \triangleq [a_1\ a_2\ b_1\ b_2]^T$.

TABLE IV
THE COMPUTATIONAL COST OF KF PER ITERATION CYCLE

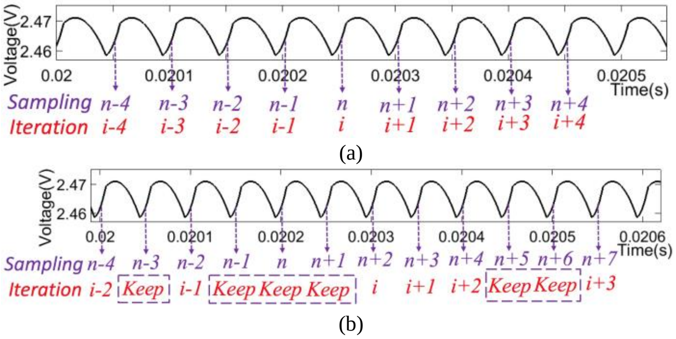
| **Step** | **Updates** | **Computational Complexity** | | |
|---|---|---|---|---|
| | | + | × | / |
| **1** | Regressor | | | |
| **2** | CMA | $M^3 + M^2, (80)$ | $M^3 + M^2, (80)$ | |
| **3** | KG | $M^2, (16)$ | $M^2 + 2M, (24)$ | 1 |
| **4** | GV | $M, (4)$ | $M, (4)$ | |
| **5** | CT | | $M, (4)$ | |
| **6** | NG | $M, (4)$ | | |
| | **In Total** | $M^3 + 2M^2 + 2M, (104)$ | $M^2 + 2M^2 + 4M, (112)$ | 1 |

$M$ is the number of transfer function coefficients.

Both Table II and IV indicate that the cost of the CMA update is higher than the sum of the costs on other steps. Therefore, if reusing CMA, the secondly proposed approach, may be achievable in these adaptive filters, computational burdens in every iteration cycle can be significantly reduced.

# III. THE PROPOSED TECHNIQUES

## A. *Technique 1: Variable Iteration Frequency*



(a)

(b)

Fig. 3. The Comparison of Iteration Frequency between the Conventional Process (a) and Proposal (b).

In the conventional iteration process (Fig.3(a)), the iteration frequency is chosen to equal the sampling frequency, which means iterations, $i, i \pm 1, i \pm 2$ ...., act after every sampling event (dashed purple lines in Fig.3), $n, n \pm 1, n \pm 2$ ....

In this work, the computational cost is reduced by lowering the iteration frequency; The recursive algorithms no longer update NG after every sampling event, instead, there are several intermediary sampling time intervals between iterations (see Fig.3(b)). During these intermediary intervals, Step 2, 3, 4 and 5 in Table I are eliminated/tailored after regressor updates. CMA, GV, CT and NG will simply hold the most recent values until the next iteration phase completes. The regressor vectors, Step 1 shown in Table I, however, need to update at every sampling instant to ensure the same parameter values are identified from the original identification process. For instance, if the sampling frequency is 40kHz, the iteration frequency can be decimated, but the regressor update frequency must be kept at 40kHz. A reduced regressor update frequency will result in estimated model parameters that do not match the 40kHz form (discrete models are sampling-frequency-dependent).

The option to flexibly change the iteration frequency is beneficial when attempting to reduce computational burden and filter the disturbances caused by abrupt system variations in sampled signals. Therefore, a variable $K$ is included in the parameter estimation process to represent the number of sampling events that take place between those samples used to derive the next parameter estimation update. Thus, the iteration frequency can be defined as:

$$Sampling\ Frequency = Iteration\ Frequency \times\ K. \qquad (4)$$

In simple terms, $K$ can be chosen as a constant, serving as a decimating factor. However, to optimize the algorithm further, it is possible to dynamically vary $K$ based on the magnitude of the control error signal. In doing so, it is possible to prioritize the need to update the parameter estimation in the event of substantial system disturbances which are likely to perturb the controller error. While, there is no need to particularly set up which sampled data will be skipped or kept, as long as the iteration frequency is reduced and iteration events of rails are stagger with each other instead of happening together.

The sampling frequency depends on the computational complexity of the estimation algorithm and the processing capability of the employed processor. Once the algorithm and the processor were selected, the time spent for each iteration cycle can be determined. There would be an upper limit of the iteration frequency to ensure that the 'Whole Iteration' cycle can be completed in one iteration interval. Typically, the iteration frequency equals the sampling frequency (i.e. the upper limits of the two frequencies are the same). Now, if the proposed iteration decimation approach is applied, the sampling frequency can be defined as long as the decimation factor ($K$ in (4)), is known. Meanwhile, the limit on the sampling frequency will lead to the same limit on the switching frequency of SMPCs, as the two frequencies are typically configured to be equal. For example, in [36], completing one iteration cycle of KF and RLS on a DSP (TMS320F28335) will respectively take 37μs and 34μs. In [36], the sampling frequency equals the switching frequency; it has to be as low as 20 kHz to guarantee the sampling intervals being longer than 37 μs. If the converter operates at a higher switching frequency, more advanced processors are required to complete one iteration cycle in one sampling interval.

Reducing iteration frequency may achieve computational burden alleviation in every sampling interval, however, may also prolong the time spent on parameter estimation. To solve this, the forgetting factor (λ) in RLS (see Table I) is investigated; As the factor also affects the identification speed, carefully tuning λ may shorten the prolonged estimation time caused by the reduced iteration frequency. Furthermore, to demonstrate the effects brought by the proposed approach on estimation performance, three commonly-used indices (Convergence Time, Estimation Error and Variance of Estimated Results) expressing estimation performance are introduced (see Fig.4). In Fig.4, the estimation process is divided into two stages; In Stage 1, the guesses of the estimated parameters are being iterated to acquire the optimal values, which therefore cannot be used for adaptive controller tuning (recursive curves indicating estimation results have not converged to the true values). In Stage 2, recursive curves have converged to the true values (the optimal guesses have been found) and kept the values for 0.01s to complete the controller parameter update. Based on Fig.4, the three indices describing estimation performance are:

1. *Convergence Time:* the duration of Stage 1 in Fig. 4, starting at the beginning of parameter estimation and ending at the time when the recursive curves have entered and remained within their error bands ±5% of real values [36].

2. *Estimation Error:* Stage 2 (see Fig. 4) begins with the end of Stage 1 and ends with 0.01s after. The average value of the recursive curve in Stage 2 is typically the estimation result taken into adaptive control account. The difference between this average value and True Value is known as the estimation error which implies estimation accuracy.

3. *Variance of Estimated Results:* the variance of Stage 2 (Fig. 4). Variance is another way to reflect the estimation accuracy in case the average is affected by extreme values.

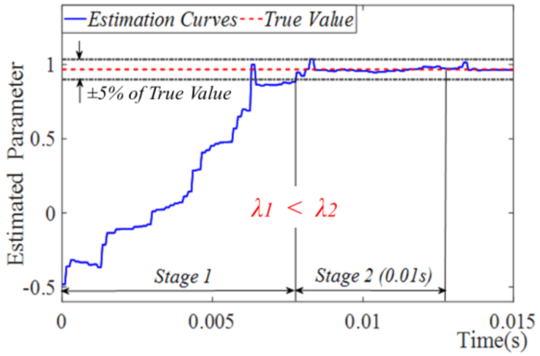
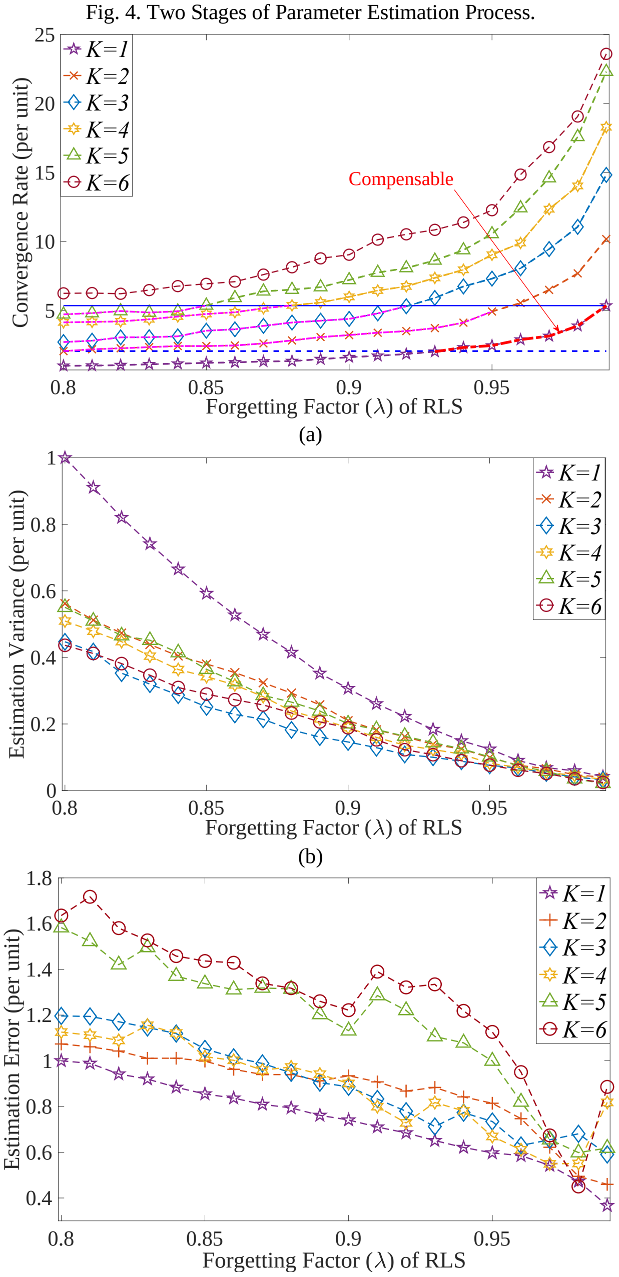


Fig. 4. Two Stages of Parameter Estimation Process.



(a)

(b)

(c)

Fig. 5. Performance Comparison of Parameter Estimation under Different Iteration Frequency and Forgetting Factor for RLS.

To shorten the prolonged convergence time caused by lowering iteration frequency, by manipulating Forgetting

Factor (λ) of RLS, the relation among $K$, λ, and estimation performances are investigated (see Fig.5). Here, Convergence Time is shown in Fig. 5(a), variance in Fig. 5(b), and estimation error in Fig. 5(c). The Y-axis values in Fig 5 are shown on a self-defined "per unit" scale; where the convergence time, estimation error and variance at $\lambda = 0.8$ and $K = 1$ is considered 'unity' or '1'. As such, all other points on the graphs are reference values with respect to the per-unit case. For example, when $\lambda = 0.82$ and $K = 3$ the convergence time is 3 times longer than that when $\lambda = 0.8$ and $K = 1$. From Fig. 5(a), it can be observed that:

1. Increasing $K$ leads to a longer convergence time of RLS, about inversely proportional to the iteration frequency; if the iteration frequency is reduced to a third, its corresponding convergence time will be about three times longer.

2. Decreasing λ will reduce the convergence time, but the larger $K$ would make it more difficult to reach an acceptable convergence time; The solid blue line in Fig. 5(a) shows the convergence time when $\lambda = 0.99$ and $K = 1$ and the dashed blue line is the convergence time when $\lambda = 0.8$ and $K = 2$, so within the two blue lines, when a larger $K$ is adopted for saving computational complexity, a small $\lambda$ (Curves depicted in pink) can then be selected to guarantee the convergence time of the larger $K$ is always the same with that of its $K = 1$ counterpart (Curves depicted in red).

According to Fig. 5(b), the larger λ and $K$ are preferred as they produce less variance; Converged curves have fewer fluctuations. In Fig. 5(c), the estimation error when $K = 1$ is less than that of others, and the errors when $K$ = 5 or 6 are significantly higher; 5 or higher values may not be suitable selections of $K$, but even if $K$ equals 2 or 3, the lower estimation accuracy cannot be ameliorated unless a larger $\lambda$ is applied which negatively influences the convergence time. Therefore, an adaptive $\lambda$ is proposed to be adjustable in different estimation stages; $\lambda$ is configured to be smaller for fast estimation speed in Stage 1 (Fig.4) and larger in Stage 2 to ensure high stability in curves. Theoretically, this approach can be applied in most stochastic-gradient-based algorithms, as typically they always include a factor directly affecting transient behaviors, e.g. step size in AP and Least-mean Square (LMS), forgetting factor in RLS, etc.

### *B. Technique 2: Iteration with Re-using CMA (Covariance Matrix Approximation)*

According to Table I and II, the computational costs of CMA updates in both RLS or KF are higher than the sum of the costs spent on other steps. Therefore, reducing the CMA update frequency is one clear way to reduce computational burdens; Between updates, the same CMA value might be re-used. As such, investigations of CMA are shown below [37].

As shown in Fig. 6, after 200 iterations, the magnitude of CMA reduced by 450, acquired by implementing RLS on practical data for real-time system identification. Then, the difference between two consecutive CMAs in RLS is investigated. Fig. 8(a) indicates the values of $CMA2 - CMA1, CMA3 - CMA2, CMA4 - CMA3, \ldots$, shown in Fig.7.

In Fig.8(a), the smaller $\lambda$ is, the greater difference between two consecutive CMAs is. Besides, the maximum fluctuation amplitude, about 80, in Fig. 8(a) is merely about 17% of the CMA reduction (450 in Fig.6), which suggests two consecutive CMAs in RLS are almost the same. It is, consequently, assumed that the CMA calculated in the last iteration is a reasonable substitution for the current iteration cycle.

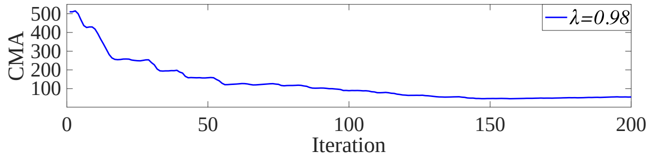


Fig. 6. Reduction of CMA of RLS in First 200 Iteration Cycles.

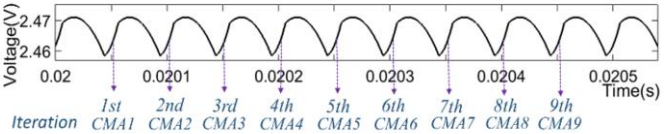


Fig. 7. The relation between CMAs and Iterations in the Conventional Parameter Estimation.

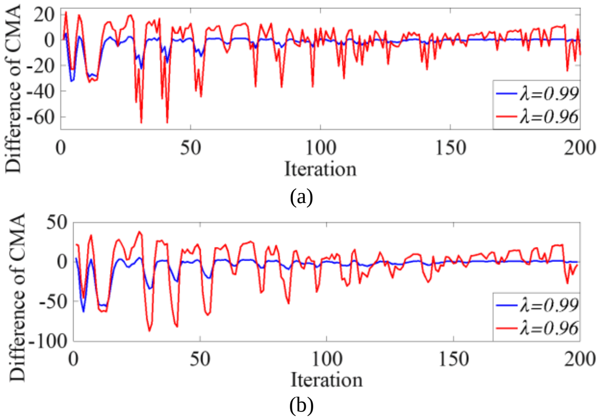


Fig. 8. CMA Differences of RLS when $Q = 2$ (a) and when $Q = 3$ (b)

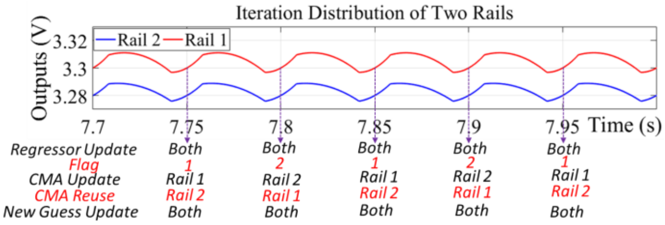


Fig. 9. The Iteration Distribution of Reusing CMA in a Two-Rail Architecture

As such, the proposal of CMA substitution allows a reduction of computational burden per iteration if two rails are identified simultaneously (see Fig.9). A variable $Q$ is introduced to indicate how many times a CMA will be used in the iteration process. In Fig.9, as the CMA calculated from the last iteration is only once reused (into the current one), $Q$ equals 2. 'Flag' indicates which converter/rail should take CMA update after a sampling event. Here, 'Whole Iteration' is stipulated to stand for conducting all steps of an iteration cycle and 'Partial Iteration' doing all steps apart from Step 2 (CMA updates) in Table II and IV. As such, after every sampling event, one rail will conduct 'Whole Iteration' and the other one takes 'Partial Iteration (see Fig. 9). Then the computational consumption is reduced from how much two times a 'Whole Iteration' costs to how much a 'Whole Iteration' and one 'Partial Iteration' cost, which saves more than half of the computational efforts of the conventional estimation way (see Table V).

To alleviate computational burdens further, the CMA calculated at the last iteration is reused twice for both the current and the next iteration cycles. Fig.8(b) shows the difference between the next CMA and the last one，the values of $CMA3 - CMA1, CMA4 - CMA2, CMA5 - CMA3$ ,…, shown in Fig.7. ($Q$ equals 3 now.) As the magnitude differences of CMA when $Q = 3$ are very similar to those when $Q = 2$, it is assumed reusing CMA twice is also a feasible option to further reduce the computational complexity, which may be applied into simultaneously identifying three rails (see Fig.10); After every sampling event, one rail conducts 'Whole Iteration' and the other two take 'Partial Iteration'. The significantly reduced computational consumption is listed in Table V.

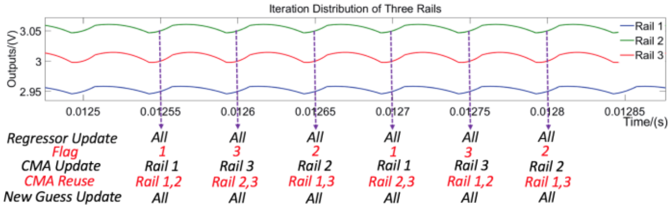


Fig.10. The Iteration Distribution of Reusing CMA in a Three-Rail Architecture.

TABLE V
THE COMPUTATIONAL COST OF DIFFERENT ITERATION STRATEGIES

| **Iteration Event** | **Computational Complexity** | | | | | |
|---|---|---|---|---|---|---|
| | **RLS** | | | **KF** | | |
| | + | × | / | + | × | / |
| *a 'Whole Iteration'* | 64 | 109 | 1 | 104 | 112 | 1 |
| *a 'Partial Iteration'* | 20 | 24 | | 24 | 32 | 1 |
| *$K = 1$ or $Q = 1$ of three rails (Three 'Whole')* | 192 | 327 | 3 | 312 | 336 | 3 |
| *$K$ = 3/2 of three rails (One 'Whole', One 'Partial')* | 84 | 133 | 1 | 128 | 144 | 2 |
| *$K$ = 3 of three rails (One 'Whole')* | 64 | 109 | | | | |
| *$Q$ = 3 of three rails (One 'Whole', Two 'Partial')* | 104 | 157 | 1 | 152 | 176 | 3 |

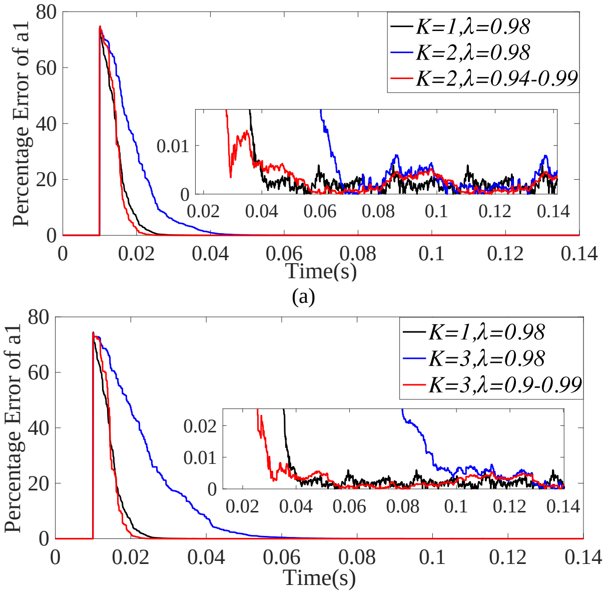
To conclude contributions of the two proposed approaches, computational burdens of parameter estimation of a three-rail power converter in different scenarios, using or not using the proposed approaches, are compared in Table V. As shown in Table V (see the first two rows), the computational costs of finishing a 'Partial Iteration' are even less than half of those of finishing a 'Whole Iteration'. If a three-rail power conversion architecture is being identified in the conventional way ($K = 1$ or $Q = 1$), after every sampling event, all the three rails would conduct 'Whole Iteration'. The corresponding computational costs would be three times the costs of a 'Whole Iteration' (see Table V). However, using the firstly proposed iteration decimation approach ($K = 3$) there is only one rail conducting 'Whole Iteration', whilst one rail conducts 'Whole Iteration' and the other two take 'Partial Iterations' using the secondly proposed CMA substitution approach ($Q = 3$). The computational costs of $K = 3$ and $Q = 3$ are both listed in Table V. Accordingly, if the sampling frequency is 20 kHz, the employed processor should complete 192 additions, 327 multiplications and 3 divisions in every sampling interval (50μs) for not using either of the two proposed approaches. This heavy computational burden may cause that all iteration tasks not to be able to finish in a given sampling event. The solution can be to replace the currently used processor with another more computationally capable one, which, however, could cause extra hardware costs. Nevertheless, the proposed approaches may alleviate the burdens in a cost-exempt way; The computational burden during each sampling interval would be almost halved by using the CMA re-using approach ($Q = 3$), and even be less than halved by staggering iteration actions among the three rails ($K = 3$). As such, neither the extra processor costs nor the reduction of the sampling frequency are needed.

## IV. SIMULATION RESULTS

To verify the performance of the proposed approaches, a single buck converter is simulated using Simulink/Matlab and the RLS algorithm (Table I) is implemented to estimate transfer function coefficients [$a_1$, $a_2$, $b_1$ and $b_2$ in (2)] of the converter. Circuit components of the converter are designed as: $V_{in} = 10\text{V}, L = 220\ \mu\text{H}, R_C = 25\ \text{m}\Omega, R_L = 68\ \text{m}\Omega,\ f_S = 20\ \text{kHz}$.

As initial setups, all the elements of Regressors ($u_{-1}$ and $y(-1)$), Gradient Vector ($e_{-1}$) and New Guess ($\omega_{-1}$) in Table I are 0, apart from CMA ($P_{-1}$) which is $1 \times 10^{-3}\boldsymbol{I}$. ($\boldsymbol{I}$ is a 4 by 4 unity matrix here.)

### A. Technique 1: Variable Iteration Frequency



(a)

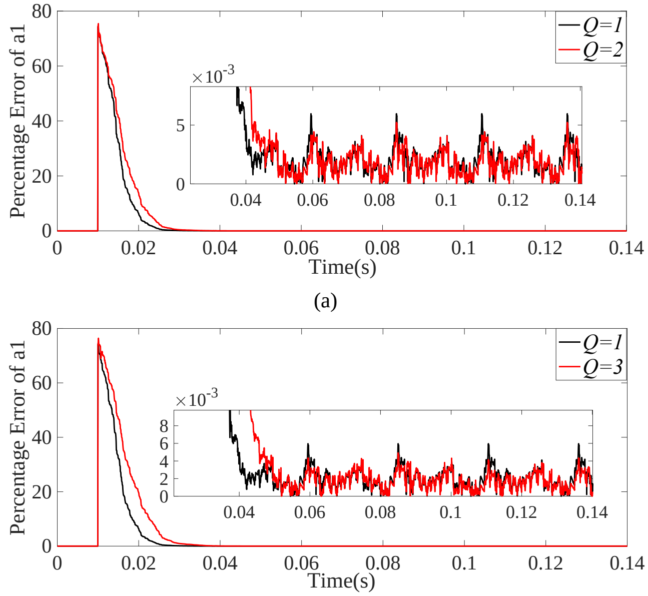
(b)

Fig.11. Estimation Error Curves in Different Iteration Frequencies

The number of iterations for the estimated parameters to converge depends on the performance of the applied algorithms and the complexity of the circuit model identified. It is found that the classical RLS iterates about 200 times to come out with reliable coefficient values of a second-order linear transfer function, (2). As shown in Fig.11, with the same forgetting factor (λ) value, 0.98, the speed of minimizing the error when Iteration Frequency (IF) equals Sampling Frequency (SF) is double faster than that when IF is half of SF, or three times faster than that when IF is 1/3 of SF. According to the proposed approach, after reducing λ from 0.98 to 0.94 only in Stage 1 of the low IF identification process, the convergence time is shortened correspondingly while the estimation error keeps the same level as that of $K = 1$ (see Fig.11). Therefore, the proposed approach of reducing IF and λ outperforms the conventional way in terms of half or even 2/3 less computational costs without compromises in Convergence Time and Estimation Accuracy.

### B. *Technique 2: Iteration with Re-using CMA*

This section presents the results of applying the second proposed approach to identify the parameters of the single buck converter; Fig.12 shows percentage error curves of reusing CMA of RLS calculated in the last iteration cycle into the current ($Q = 2$), or even the next ($Q = 3$) ones. λ is selected as 0.98 for $Q = 1$, 2 and 3. According to Fig.12, reusing CMA would not prolong the convergence time as the iteration frequency does not change, and both reusing CMA once and twice could achieve the same estimation accuracy with less computational costs.



(a)

(b)

Fig.12. Estimation Error Curves in Different Iteration Strategies

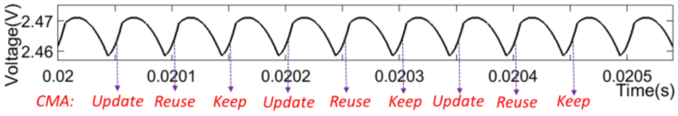


Fig.13. Arrangement for the Combination of Reducing Iteration Frequency and Reusing CMA

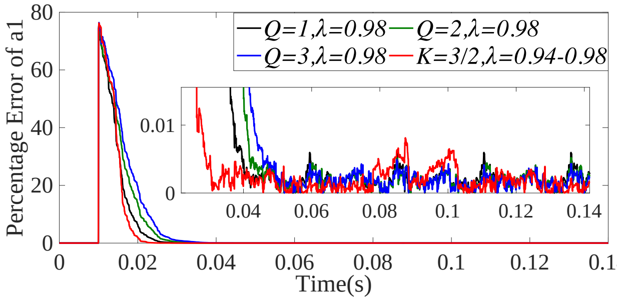


Fig.14. Performance Comparison between Q=3 and K=3/2

As shown in Table V, to identify three rails with RLS, although the computational complexities at $K = 3$ are lower than those at $Q = 3$, the scenario of $K = 3$ needs a significant decrease of λ in Stage 1 to shorten the prolonged convergence time. Therefore, the two proposed approaches might be combined as Fig.13 shows; CMA is reused once, after which there is a slot without iterations. As such, the convergence time will only be prolonged by 1.5 times instead of 3 times (as $K$ equaling 3 does), because the iteration frequency is 2/3 of the sampling frequency now ($K$ equals 3/2). As such, this combination could result in a less decrease in λ in Stage 1, and the estimation error is almost as same as $Q$ equaling 3 does.

### C. *Load Changes Rejection (Disturbance Disposal)*

This section demonstrates the impact of reducing iteration frequency to cope with abrupt disturbances (resistance load changes here); Because the load change occurs in Stage 2, λ is normally a large value such as 0.98. With a reduced iteration frequency, the proposed approach would spend a long time updating the estimation results. Reducing λ for shortening the convergence time, however, may not be suitable for disruption rejection, as it will assign more weights to recently updated regressors. Therefore, $K$ is temporarily increased to 10 (decreasing the iteration frequency) to dispose of disturbances in sampled signals; Fig.15 shows the iteration arrangement when an abrupt disturbance occurs in one rail of a three-rail architecture, where no iteration will be allocated when sampled signals are transient responses dealing with system variations.

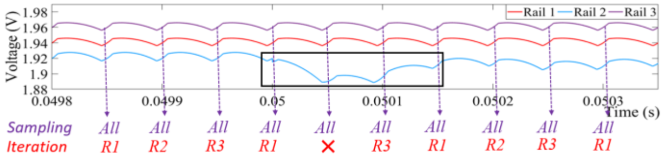


Fig.15. Disturbance Disposal within Sampled Signals

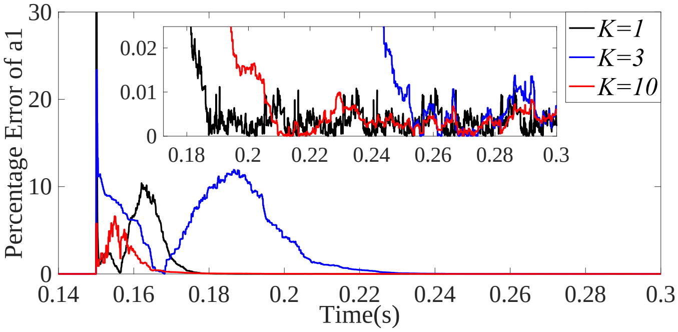


Fig.16. Estimation Error Comparison of $K = 10, K = 1$ and $K = 3$

A performance comparison is shown in Fig.16, the recursive curve of $K = 10$ enters the accuracy tolerance range, ±5% of real values, faster than other ways do and features the most moderate transient behavior.

Fig.17 presents the comparison between reusing CMA and locally decreasing the iteration frequency to deal with sudden load changes; CMAs respectively calculated from two consecutive iterations contain differences already, the transient responses of voltage and duty cycle signals to reject load changes would make the differences larger. Consequently, the CMA updated in the last iteration cycle will not be usable for the current one, proved by the severely fluctuated transient behavior in the recursive curve of $Q = 3$. Therefore, the iteration frequency is also locally reduced as $K = 10$, here for removal of sampled disturbances (see Fig.17), both the convergence time and transient behaviors are improved.

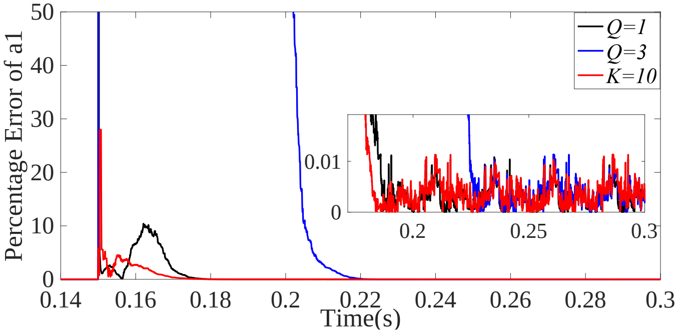


Fig.17. Error Comparison of $K = 10\ (Q = 2), Q = 3, Q = 1$

## V. Experimental Validation

To experimentally verify the proposed two approaches, parameter estimation is conducted, using RLS and KF, on a prototype multi-rail power conversion architecture comprising of three parallel buck converters (see Fig.18). Circuit components are similar to those used in the Simulink (see section V), apart from the output capacitors, resistance loads and the regulated output voltages as shown in Table VI. Table VII shows the power converter coefficients of each rail.

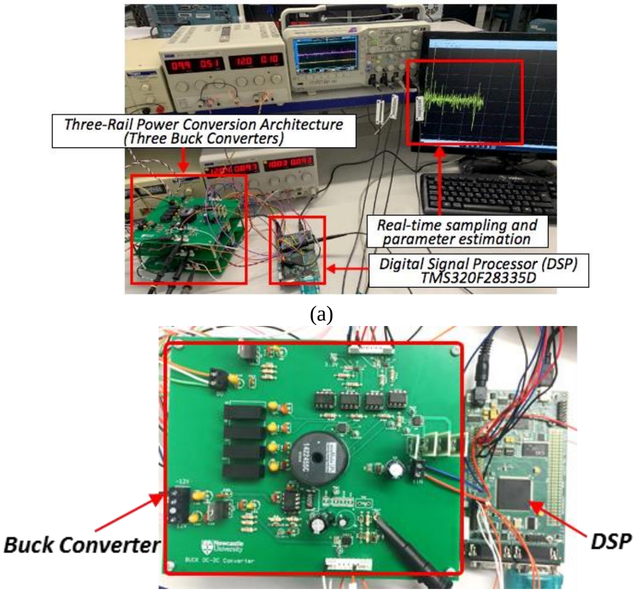
TABLE VI
The Circuit Component Values of Each Rail

| Parameters | Rail 1 | Rail 2 | Rail 3 |
|---|---|---|---|
| **C(μF)** | 470 | 330 | 220 |
| **R(Ω)** | 5 | 5 | 10 |
| $\mathbf{V_{out}(V)}$ | 1.8 | 3.3 | 5 |
| **Filter used** | 5 tap+4 tap | 5 tap+4 tap | 5 tap+3 tap |

TABLE VII
The Transfer Function Coefficients of Each Rail

| Parameters | Rail 1 | Rail 2 | Rail 3 |
|---|---|---|---|
| $\boldsymbol{a_1}$ | -1.9348 | -1.9163 | -1.9066 |
| $\boldsymbol{a_2}$ | 0.9586 | 0.95 | 0.9572 |
| $\boldsymbol{b_1}$ | 0.1759 | 0.2258 | 0.3099 |
| $\boldsymbol{b_2}$ | 0.0624 | 0.1118 | 0.1955 |

.


Three-Rail Power Conversion Architecture (Three Buck Converters)
Real-time sampling and parameter estimation
Digital Signal Processor (DSP) TMS320F28335D
Buck Converter
DSP

(a)

(b)

Fig. 18. (a) The Parameter Estimation Process. (b) The PCB Board of Buck Converter with DSP.

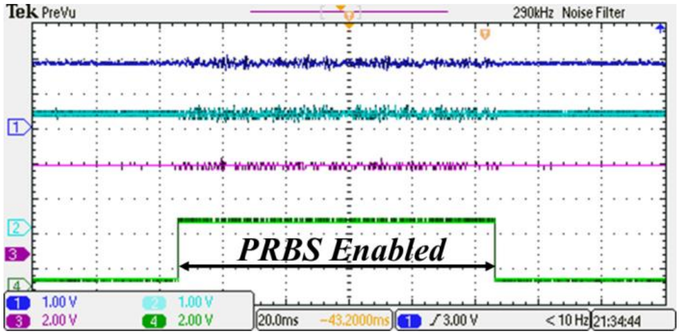


Fig. 19. The Three output voltages from the Multi-Rail Power Converter with PRBS.

As shown in Fig.2, each rail (Buck Converter) is controlled by independently tuned digital PI controllers. The transfer function of the controller is shown below:

$$G_c(z) = \frac{0.41 - 0.4z^{-1}}{1 - z^{-1}} \tag{5}$$

A Texas Instruments TMS320F28335 Digital Signal Processor (DSP) is used to implement the digital PID controllers, PRBS, and data captured using an Embedded Coder Support package [38]. For parameter estimation, a ±0.025 PRBS is injected for 100ms and the perturbation in output voltages of three rails are shown in Fig.19. At the perturbation period, 600 samples of the control signals and the output voltages for the three rails are collected at the sampling frequency equalling 20 kHz. Before going into algorithm blocks, sampled duty cycle and voltage signals should be filtered for noise removal; here Moving Average Filters (MAFs) are used for this purpose. Each filter is chosen to remove unwanted noises, but still ensure parameter estimation accuracy to within ±5% of expected results.

The experimental validation includes performance comparisons, in terms of estimation accuracy, computational costs and Convergence Time, of 1. Two iteration frequencies respectively equaling to sampling frequency and one-third of it. 2. The same CMA being once, twice and thrice used in consecutive iteration intervals. 3. Abrupt disturbance rejection with or without locally and temporarily disposing of transient responses caused by sudden load changes.

### A. Technique 1: Iterative Decimation Approach

Conventionally, iteration events occur in all three rails after every sampling action whereas in the proposed method after the update of regressors of three rails, only one rail will take the iteration cycle. The distribution of iteration events of each rail is presented in Fig.20, where the iteration frequency of each rail in the proposal ($K = 3$) is three times lower than the sampling frequency. Here, a Flag is allocated to indicate which rail is taking places to process with the iteration action.

To clearly analyze the estimation accuracy, estimation errors are considered in two ways described as 'Average Error (AE)' and 'Process Error (PE)'. AE means the offset between the true value of power converters' parameters and the average, averaging estimated results from the point that recursive curves start to enter and remain in the accuracy tolerance band (±5% of real values) to the end of the estimation process (here, at 0.03s). As this average is typically the results that controller retuning is based on, AE could express the performance of adaptive control. PE stands for fluctuations of recursive curves, which can indicate the stability/robustness of the estimation process, so does the Variance of recursive curves. Therefore, an estimation approach very capable of noise rejection would feature recursive curves with low PE and small variations.

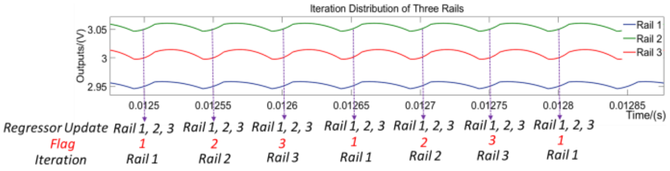
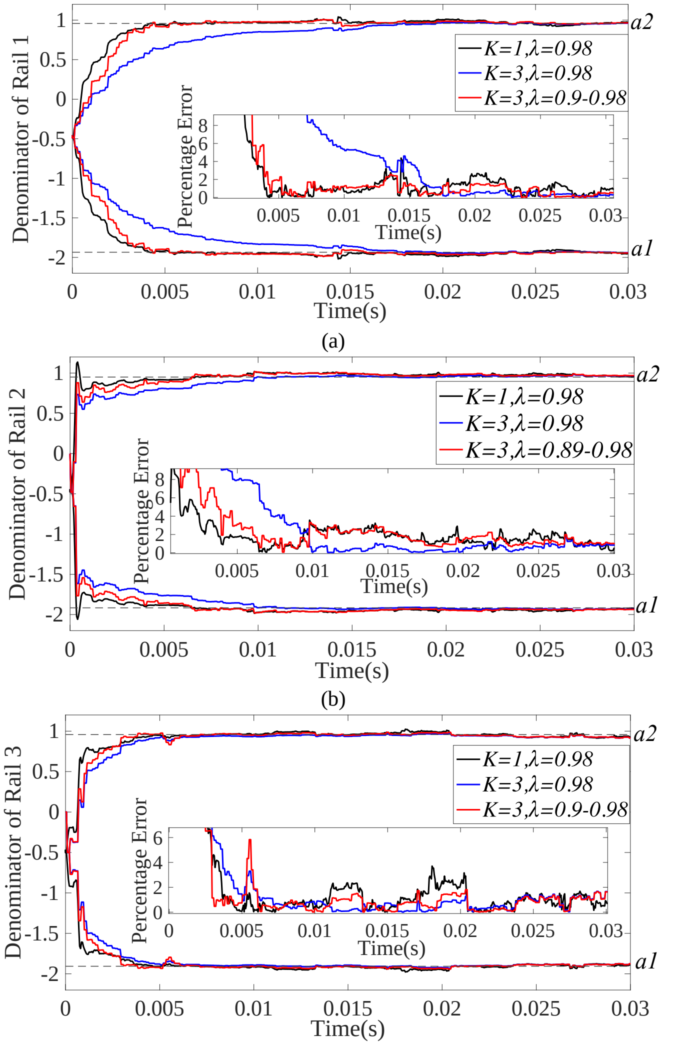


Fig. 20. The Iteration Distribution of Three-Rail Power Converter for Lowering Iteration Frequency.



(a)

(b)

(c)

Fig. 21. Recursive Curves of Three Rails with Different Iteration Frequencies

Fig.21 shows the convergence time and the estimation error of the denominator coefficients of the three converter rails with λ=0.98. Like simulation results, decreasing the iteration frequency 3 times leads to the convergence time increasing about 3 times in experimental results; In Rail 1 it is more than three times while in Rail 3 the two rates, acquired from the conventional way ($K = 1$) and the proposed approach ($K = 3$), are almost the same. As such, values of λ of each rail when $K = 3$ are respectively adjusted (see Table VIII), after which the prolonged convergence time is shortened (see Fig.21); In simulation λ in Stage 1 is only reduced by 0.05 from 0.98 to 0.94, however, it is decreased by 0.08 to 0.9 in experiments. As Fig.21 shows, PEs when $K = 1$ and $K = 3$ in three rails are all under ±5% of the real values (within the accuracy tolerance band), same with simulated results. Table IX lists estimation error (AE), the convergence time, and variance of recursive curves when $K = 1$ and $K = 3$ of each rail. The biggest difference of the convergence time between $K = 1$ and $K = 3$ is 1.55ms in Rail 2, whereas there is barely compromise of identification speed in other rails. Besides, apart from Rail 2, the approach of reducing iteration frequency and λ features the highest estimation accuracy (see AE) and the strongest stability (see Variance). Nevertheless, the most contribution of this work is in computational cost-saving (see Table X); In Stage 1, the computational cost of $K = 3$ is even less than half of it of $K = 1$. Therefore, the proposed method can be reliably applied on, and quite suitable for, online system identification of multi-rail architectures, as it achieves the computational costs on concurrently identifying multi-rails equals that on doing single rail without noticeable compromises on other performances.

TABLE VIII
ADJUSTMENT OF FORGETTING FACTOR FOR LOW ITERATION FREQUENCY

| Forgetting Factor | Rail 1 | Rail 2 | Rail 3 |
|---|---|---|---|
| Stage 1 (See Fig.3) | 0.9 | 0.89 | 0.9 |
| Stage 2 (See Fig.3) | 0.98 | 0.98 | 0.98 |

TABLE IX
PERFORMANCE COMPARISON OF PARAMETER ESTIMATION WITH DIFFERENT ITERATION FREQUENCIES

| | | Convergence time (ms) | Estimation Error (e-4) | Variance (e-4) |
|---|---|---|---|---|
| Rail 1 | $K = 1$ ($\lambda = 0.98$) | 3.05 | 119 | 5.7215 |
| | $K = 3$ ($\lambda = 0.98$) | 11.45 | 167 | 11 |
| | $K = 3$ ($\lambda = 0.9$) | 3.95 | 102 | 3.3989 |
| Rail 2 | $K = 1$ ($\lambda = 0.98$) | 2.3 | 155 | 9.976 |
| | $K = 3$ ($\lambda = 0.98$) | 6.85 | 0.577 | 6.6944 |
| | $K = 3$ ($\lambda = 0.89$) | 3.85 | 190 | 8.5467 |
| Rail 3 | $K = 1$ ($\lambda = 0.98$) | 3 | 46 | 5.9641 |
| | $K = 3$ ($\lambda = 0.98$) | 3.75 | 128 | 2.4643 |
| | $K = 3$ ($\lambda = 0.9$) | 3 | 14 | 3.6318 |

TABLE X
COMPUTATIONAL COSTS IN TOTAL USING RLS

| Iteration Frequency | Iteration Times | Computational Complexity | | |
|---|---|---|---|---|
| | | + | × | / |
| $K = 1$ | *167* | 10688 | 18203 | 167 |
| $K = 3$ ($\lambda = 0.9 - 0.98$) | *72* | 4608 | 7848 | 72 |

## *B. Technique 2: CMA Recycle Approach*

The iteration distribution arrangements of reusing CMA once ($Q = 2$) and twice ($Q = 3$) are separately presented in Fig.22 and Fig.10; The Flag value indicates which converter is to be identified after a sampling event. The experimental validation of reusing CMA once is a combination between 'Reducing Iteration Frequency' and 'Recycling CMA'; Shown in Fig.22, in every sampling interval, one rail accepts the 'Whole Iteration', whilst the other one takes 'Partial Iteration' and the left one is not allocated with iterations which therefore holds its results until Flag indicates its updates. Because the iteration frequency is 2/3 of the sampling frequency now, this scenario is described as $K = 3/2$ rather than $Q = 2$. As Table V shows, the computational costs of $K = 3/2$ are reduced from three times a 'Whole Iteration' costs to a 'Whole Iteration' and a 'Partial Iteration' cost (see Table V), which saves more than half of the computational effort of using the conventional estimation way ($Q = 1$). With lowering the iteration frequency, in simulation λ is lowered from 0.98 to 0.94 in Stage 1 to shorten the prolonged convergence time. However, as the estimation speed is not significantly lowered by increasing $K$ to 3/2 in practice, all results are still acquired with the same λ equaling 0.98.

As shown in Fig.10, re-using CMA twice ($Q = 3$) indicates that after every sampling event one rail conducts 'Whole Iteration' and the other two take 'Partial Iteration', the corresponding computational cost becomes a 'Whole Iteration' and two 'Partial Iteration' cost (see Table V).

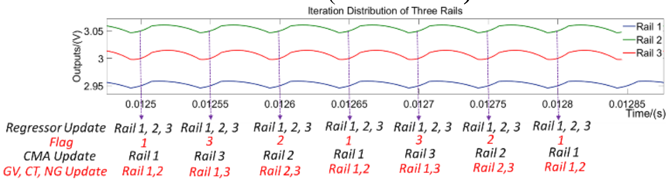


Fig. 22. The Iteration Distribution of Reusing CMA in a Three-Rail Converter

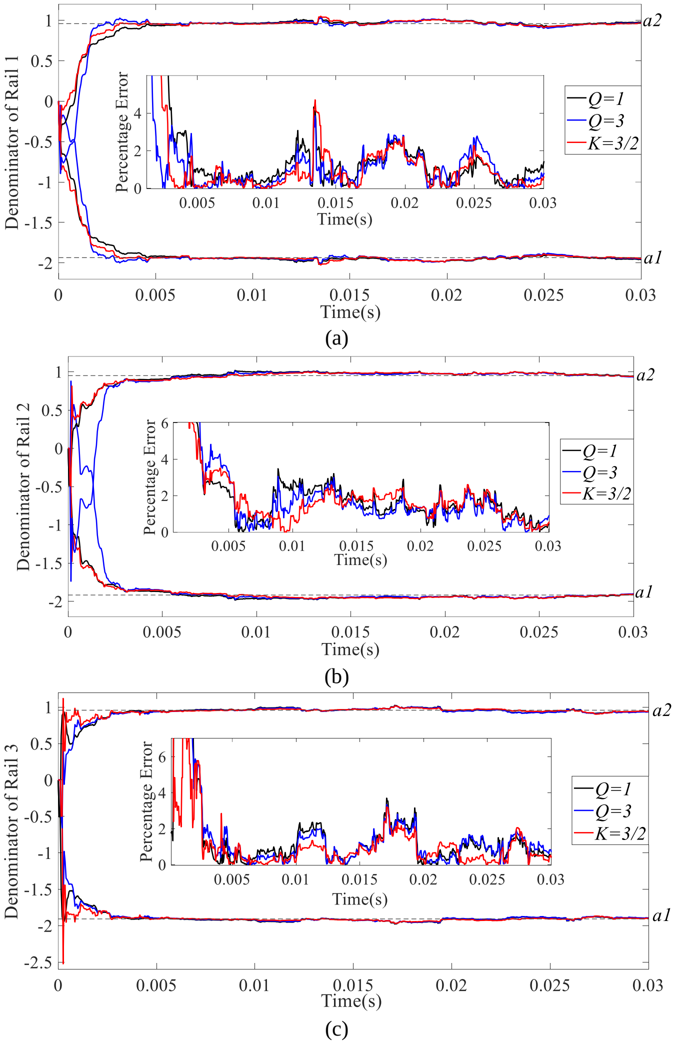


Fig. 23. Recursive Curves of Reusing CMA in Three-Rail Architecture (RLS)

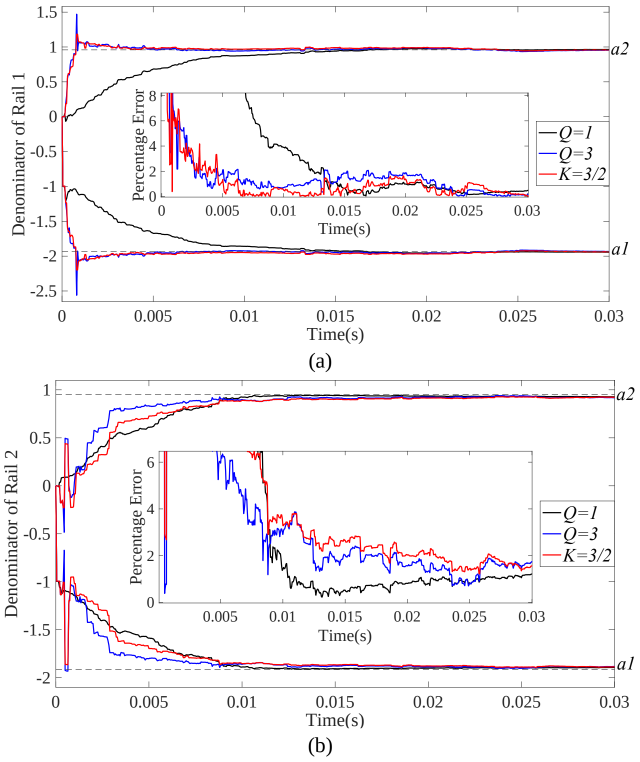
Fig.23 shows estimation curves of denominator coefficients of each rail under the afore-mentioned three scenarios, of which performances are compared in Table XI in terms of the convergence time and the estimation accuracy. As simulation results, PEs at $Q = 1$, $K = 3/2$ and $Q = 3$ of each rail are all in the same level (within the accuracy tolerance band). For all the three rails, there is no noticeable difference of the convergence time in $Q$ equaling 1 and 3 and of K equaling 3/2. Besides, AE or Variances, all in the same level, implies that in practical work, in the presence of noises, the differences of the CMAs in two, or even in three consecutive iterations would be small enough for reusing CMA. The total computational costs of each scenario are also compared in Table XII, which suggests that the proposed approaches may alleviate more than half of the computational burdens of the conventional way.

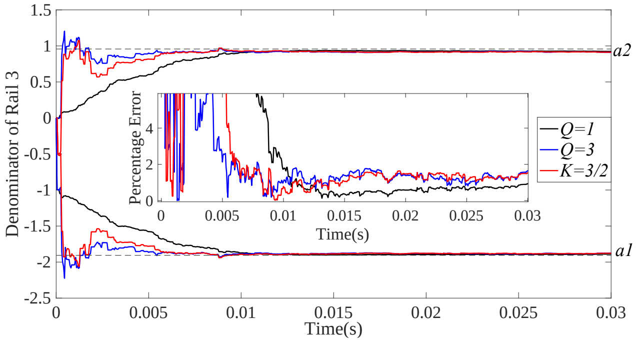
TABLE XI
ESTIMATION PERFORMANCE COMPARISON OF REUSING CMA OF RLS

| | | Convergence Time (ms) | Estimation Error (e-4) | Variance (e-4) |
|---|---|---|---|---|
| Rail 1 | $Q = 1$ | 2.95 | 97 | 6.1205 |
| | $Q = 3$ | 1.75 | 68 | 5.5957 |
| | $K = 3/2$ | 2.45 | 60 | 5.4385 |
| Rail 2 | $Q = 1$ | 2.7 | 180 | 7.7182 |
| | $Q = 3$ | 2.8 | 92 | 9.5067 |
| | $K = 3/2$ | 2.75 | 104 | 9.8121 |
| Rail 3 | $Q = 1$ | 2.2 | 46 | 5.818 |
| | $Q = 3$ | 2.5 | 4.8968 | 5.3967 |
| | $K = 3/2$ | 2.5 | 22 | 4.9184 |

TABLE XII
COMPUTATIONAL COST IN TOTAL USING RLS

| Iteration Frequency | Iteration Times in Total | Computational Complexity | | |
|---|---|---|---|---|
| | | + | × | / |
| $Q = 1$ | *157 'Whole'* | 10048 | 17113 | 157 |
| $Q = 3$ | *45 'Whole', 96 'Partial'* | 4800 | 7209 | 45 |
| $K = 3/2$ | *53 'Whole', 51 'Partial'* | 4412 | 7001 | 53 |





(c)

Fig. 24. Recursive Curves of Reusing CMA in Three-Rail Architecture (KF)

The realization of the CMA reusing approach on KF applies the same iteration distribution arrangements (see Fig.22 and Fig.10) used on RLS. Fig.24 shows the recursive curves of the estimated parameters of the three rails, which indicates the convergence time of reusing CMA once and twice in Rail 1 and 3 is even shorter than that of the conventional iteration way; The reused CMA, calculated from the last iteration cycle, may cause the CT overcorrecting the new guess in the current iteration cycle, then the next CMA, consequently, will keep being overcorrecting. As a result, the convergence time might be shortened but there are severer fluctuations in recursive curves. Nevertheless, all recursive curves still converged into the accuracy tolerance band (±5% of the true values). Table V shows that applying the CMA reusing approach on KF may save more than half of the computational cost of using the conventional way, as this approach being verified on RLS does. These qualified performances demonstrate that CMA reusing approach cannot only be employed on RLS but also on KF for computational burden alleviation.

### C. Abrupt Disturbance Rejection

When system variations, such as frequent and/or periodic load changes in SMPCs, occur during system identification processes, the ability to reject abrupt system changes to make recursive curves quickly updating new results is important [39]. To validate the disturbance rejection ability of locally reducing the iteration frequency, load changes are configured in Rail 1 (output current changes from 0.36 A to 1.8 A) and Rail 2 (from 0.66 A to 3.3 A) at 10ms, and in Rail 3 (from 0.5 A to 2 A) at 15ms. Fig.25 shows the transient response of the output voltage from Rail 2 coping with an abrupt load change during being estimated (PRBS is being injected), same with those in the other two rails. After the load variations, the four model coefficients of the three rails change to the values listed in Table XIII. 600 data including transient responses of output voltage coping with abrupt load change are sampled and estimated using the RLS algorithm with $\lambda = 0.98$. For the space limit, only one parameter curve, $a_1$, is presented to show the effects of load changes on estimation curves.

TABLE XIII
TRANSFER FUNCTION COEFFICIENTS AFTER LOAD CHANGE

| Rail | Loads | *a1* | *a2* | *b1* | *b2* |
|---|---|---|---|---|---|
| **Rail 1** | 1Ω | -1.8591 | 0.8827 | 0.1761 | 0.0603 |
| **Rail 2** | 1Ω | -1.8117 | 0.8447 | 0.2234 | 0.1058 |
| **Rail 3** | 2.5Ω | -1.8454 | 0.8949 | 0.3063 | 0.1887 |

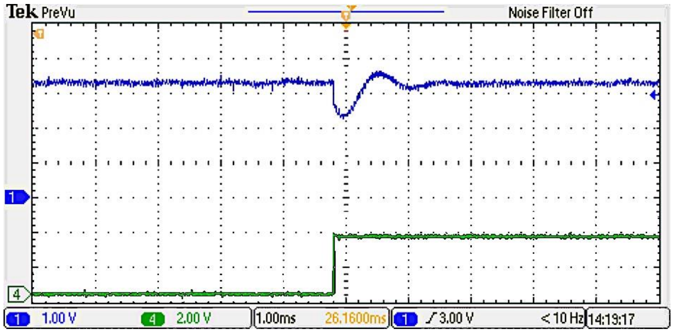
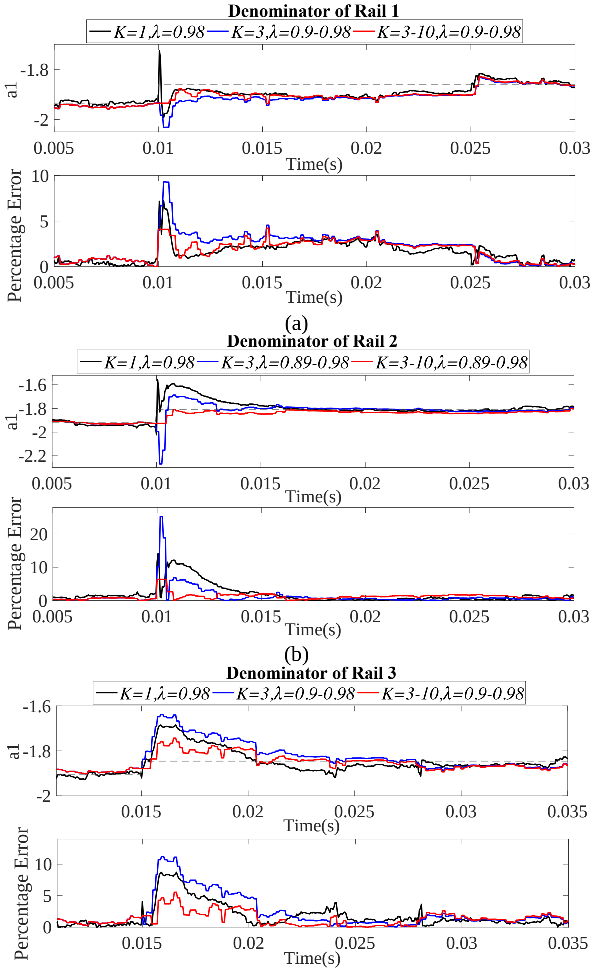


Fig. 25. The Transient Response of the Output Voltage from Rail 2



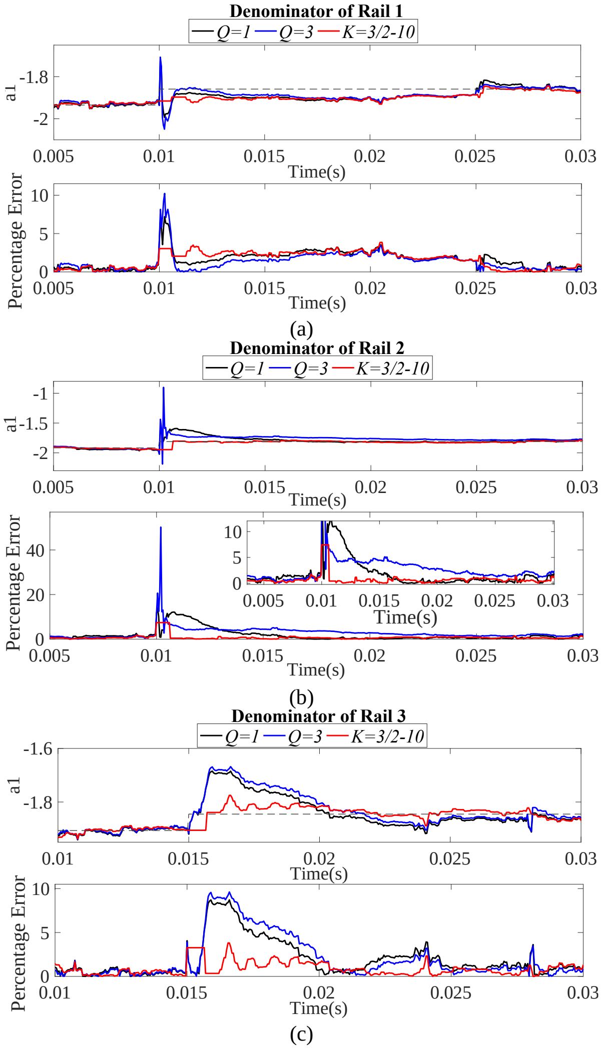
(a)

(b)

(c)

Fig. 26. Recursive Curve Comparisons between Locally Disposing Disturbance Signals and Fixed Iteration Frequency Approaches (RLS)

During oscillations, transient behaviors dealing with load changes, for performance comparison of disturbance rejection the iteration frequency is differently configured. Fig.26 shows the transient convergence behaviors for new result updates of the iteration frequency equaling the sampling frequency ($K = 1$), of lowering the iteration frequency throughout the entire estimation process ($K = 3$) and of locally and temporally lowering the iteration frequency ($K = 10$) only for the 0.5ms in which oscillations exist. Accordingly, the recursive curves of $K = 10$ features the fastest update speed and barely contain fluctuations, same as simulated results. At $K = 1$ and $K = 3$, however, oscillations of the sampled data, taken into algorithm recursion account, result in severer fluctuations of recursive curves. Meanwhile, as the convergence time also depends on the performance of the employed algorithm, the reduced iteration frequency, (or the lowered iteration density), of $K = 3$ prolongs the convergence time consequently. However, at $K = 10$, removing sampled oscillations out from iterations avoids noticeable fluctuations in recursive curves, and consequently, there is no need to consume time on coping with oscillations.



(a)

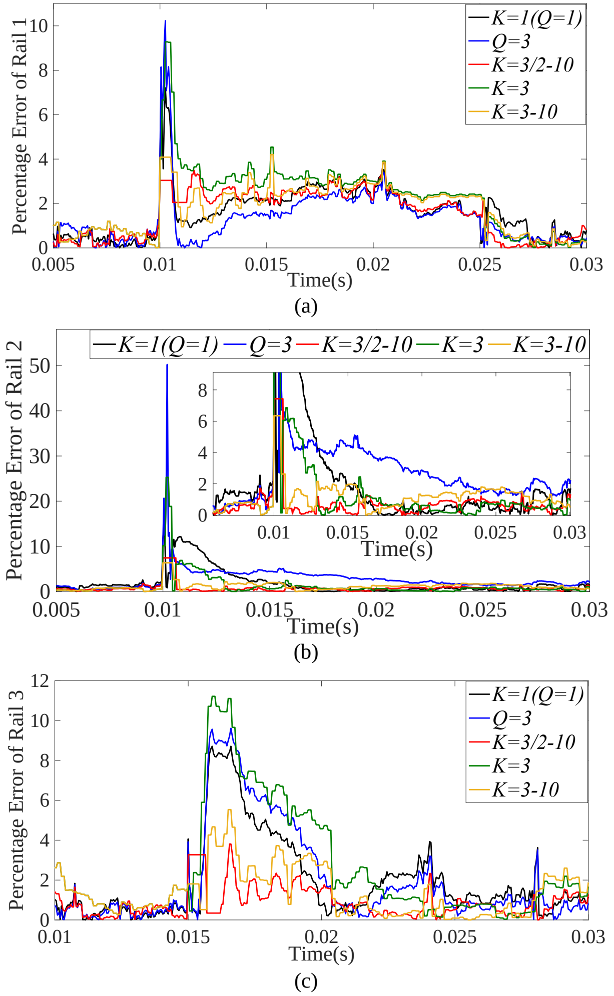
(b)

(c)

Fig. 27. Recursive Curve Comparisons between Locally Disposing Disturbance Signals and Fixed Iteration Frequency Approaches (RLS)

Fig.27 shows the convergence behaviors of respectively using the conventional way ($Q = 1$), the CMA reusing approach ($Q = 3$) and locally lowering iteration frequency approach ($K = 3/2 - 10$). Here, the iteration frequency is significantly lowered only for the 0.5ms where oscillations exist. It turns out that the experimental results are almost the same as the simulation ones. In $Q = 3$, when the sampled transient behavior of rejecting abrupt system changes is considered in recursive algorithms, two CMAs respectively calculated in the last iteration and the next, next one, are hugely different, which means CMA gotten from the last might not be a suitable replacement for the current or the next iteration events. Even worse, these inappropriate substitutions could bring more fluctuations in recursive curves and then prolong the estimation duration, demonstrated in Rail 2 (see Fig.27(b)).

As Fig.27 shows, the proposed approach of $K = 10$ performs best for disturbance rejection. However, $K$ equaling 10 may be only suitable in this case as the value selection of K for transient response removing depends on the disturbance severity, sampling frequency; A severer disturbance may cause a transient response with a larger overshoot and a longer settling time, $K$, therefore, should be adjusted larger. A high sampling frequency may lead to more transient response signals being sampled, which will be removed, $K$, therefore, should be adjusted larger. For a broad discussion of the selection of $K$, the magnitude of control error signals can be tracked to investigate the severity of variations and disturbances suffered by systems. Then a larger magnitude of the error signal may decide a larger $K$ to guarantee the sampled transient responses being removed. Better than simulation, in practical work, only the recursive curves of Rail 2 (in Fig.26 and Fig.27) are slightly excess the accuracy tolerance band (±5% of real values) in rejecting disruptions, which suggests the proposed approach may provide acceptably accurate results throughout estimation process even under disruption occurrences.



(a)

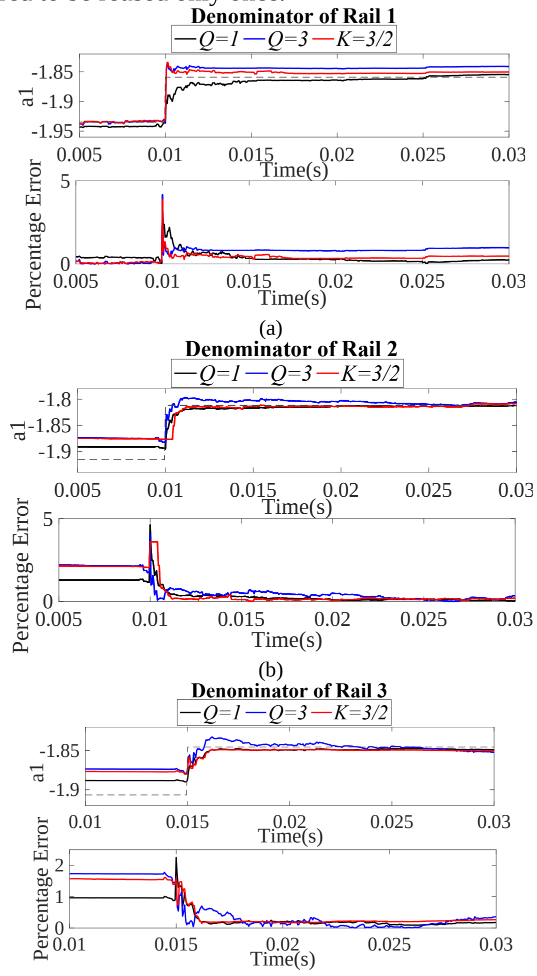
(b)

(c)

Fig. 28. Recursive Curve Comparisons among Iteration Reduction Approach, CMA Recycle Approach and Their Combination (RLS)

If iteration frequency reduction approach ($K = 3$), CMA reusing approach ($Q = 3$) and their combination ($K = 3/2$) are compared for abrupt load change rejection, the corresponding recursive curves collected in Fig.28 indicate there are no significant performance differences between the three methods. The disturbance rejection time and transient response magnitude of the estimation curves in the three rails are almost the same. However, by removing transient responses of the sampled signals ($K = 10$), estimation curves are improved.

The estimation curves of KF rejecting abrupt load changes are collected in Fig.29. The load change setups in the three rails for KF validation are the same as those for RLS validation. Here, the CMA of KF is reused. Different from the proposed approach being applied in RLS, there is no need to remove transient responses of the sampled signals for KF. Although all of the recursive curves after load changes are still in the accuracy tolerance band, that of reusing CMA twice (Q = 3) after load changes suffer severer fluctuations, which might suggest that CMA calculated in the last iteration cycle may be an inappropriate substitution for the next one. To reduce computational costs with high estimation accuracy, CMA is preferred to be reused only once.



(a)

(b)

(c)

Fig. 29. Recursive Curve of the CMA Reuse Approach in Dealing with Load Changes (KF)

## VI. Conclusion

Unlike the conventional recursive parameter estimation approaches in which, typically, the computational effort is directly proportional to the increase in the number of power converter rails, it is shown that the proposed iteration decimation approach and CMA reusing approach, can significantly reduce the computational burden of parameter estimation in complex multi-rail converter circuits. The first method has shown that the iteration frequency can be reduced. By staggering iteration events among multiple rails, the computational burden during each sampling interval could be alleviated from '$A \cdot B$', caused by using the conventional way, to '$B$' ($A$ is the number of rails and $B$ is the computational complexity of achieving one iteration cycle of a single rail). However, this approach may need a careful balance between the forgetting factor of RLS and the decimation factor, $K$, to guarantee the convergence time would not be prolonged. In the second method, it is noted that updating CMA accounts for more than half of the computational complexity of applying the recursive algorithm. Therefore, re-using the CMA over more than one iteration can reduce the computational efforts either. This work clearly shown how computational burden alleviation can be managed, whilst taking into consideration stability issues and response to abrupt system variation into consideration. Overall, by applying either of the two methods, or their combinations, the total computational effort on multi-rail architecture estimation can effectively be reduced to that on a single power converter estimation, whilst preserving the overall transient behavior of all converters.

## References

[1] R. Miftakhutdinov, L. Sheng, J. Liang, J. Wiggenhorn, and H. Huang, "Advanced control circuit for intermediate bus converter," in *2008 Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, 2008: IEEE, pp. 1515-1521.

[2] R. Miftakhutdinov, "Power distribution architecture for tele-and data communication system based on new generation intermediate bus converter," in *Telecommunications Energy Conference, 2008. INTELEC 2008. IEEE 30th International*, 2008: IEEE, pp. 1-8.

[3] J. Brown, "Addressing the topologies, converters, and switching devices for intermediate bus architectures," in *2005 European Conference on Power Electronics and Applications*, 2005: IEEE, pp. 9 pp.-P. 9.

[4] R. V. White, "Emerging on-board power architectures," in *Eighteenth Annual IEEE Applied Power Electronics Conference and Exposition, 2003. APEC'03.*, 2003, vol. 2: IEEE, pp. 799-804.

[5] E. Pritchard, L. Mackey, D. Zhu, D. Gregory, and G. Norris, "Modular electric generator rapid deployment DC microgrid," in *2017 IEEE Second International Conference on DC Microgrids (ICDCM)*, 2017: IEEE, pp. 106-110.

[6] B. Aluisio, S. Bruno, L. De Bellis, M. Dicorato, G. Forte, and M. Trovato, "DC-microgrid operation planning for an electric vehicle supply infrastructure," *Applied Sciences,* vol. 9, no. 13, p. 2687, 2019.

[7] B. Haug. "72V Hybrid DC-to-DC Converter Reduces Intermediate Bus Converter Size by up to 50%." Power Electronics. https://www.powerelectronics.com/technologies/power-management/article/21864090/72v-hybrid-dcdc-reduces-intermediate-bus-converter-size-by-up-to-50 (accessed Jan. 29, 2018).

[8] "TPS653850-Q1 Multirail Power Supply for Microcontrollers in Safety-Relevant Applications." TEXAS INSTRUMENTS. https://www.ti.com/lit/ds/symlink/tps65381-q1.pdf?ts=1590492873017 (accessed June, 2017).

[9] S. Davis. "42V Quad Monolithic Synchronous Step-Down Regulator." Power Electronics. https://www.powerelectronics.com/technologies/regulators/article/21862796/42v-quad-monolithic-synchronous-stepdown-regulator (accessed Aug. 10, 2015 ).

[10] R.-J. Wai, Y.-F. Lin, and Y.-K. Liu, "Design of adaptive fuzzy-neural-network control for a single-stage boost inverter," *IEEE*

*Transactions on Power Electronics,* vol. 30, no. 12, pp. 7282-7298, 2015.

[11] G. E. Pitel and P. T. Krein, "Real-time system identification for load monitoring and transient handling of Dc-Dc supplies," in *2008 IEEE Power Electronics Specialists Conference*, 2008: IEEE, pp. 3807-3813.

[12] Y. Yin *et al.*, "Advanced Control Strategies for DC-DC Buck Converters With Parametric Uncertainties via Experimental Evaluation," *IEEE Transactions on Circuits and Systems I: Regular Papers,* 2020.

[13] L. Liu, W. X. Zheng, and S. Ding, "An adaptive SOSM controller design by using a Sliding-Mode-Based filter and its application to buck converter," *IEEE Transactions on Circuits and Systems I: Regular Papers,* 2020.

[14] M. Jovanovic, "Power conversion technologies for computer, networking, and telecom power systems-past, present and future," in *International Power Conversion & Drive Conference (IPCDC) St. Petersburg, Russia*, 2011, pp. 156-161.

[15] G. A. Dumont and M. Huzmezan, "Concepts, methods and techniques in adaptive control," in *American Control Conference, 2002. Proceedings of the 2002*, 2002, vol. 2: IEEE, pp. 1137-1150.

[16] M. Al-Greer, M. Armstrong, M. Ahmeid, and D. Giaouris, "Advances on System Identification Techniques for DC–DC Switch Mode Power Converter Applications," *IEEE Transactions on Power Electronics,* vol. 34, no. 7, pp. 6973-6990, 2018.

[17] N. Beohar, V. N. Malladi, D. Mandal, S. Ozev, and B. Bakkaloglu, "Online built-in self-test of high switching frequency dc–dc converters using model reference based system identification techniques," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 65, no. 2, pp. 818-831, 2017.

[18] R.-G. Li and H.-N. Wu, "Iterative Approach With Optimization-Based Execution Scheme for Parameter Identification of Distributed Parameter Systems and its Application in Secure Communication," *IEEE Transactions on Circuits and Systems I: Regular Papers,* 2020.

[19] F. Hafiz, A. Swain, E. M. Mendes, and L. A. Aguirre, "MultiObjective Evolutionary Approach to Grey-Box Identification of Buck Converter," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 67, no. 6, pp. 2016-2028, 2020.

[20] A. Pedross-Engel, H. Schumacher, and K. Witrisal, "Modeling and identification of ultra-wideband analog multipliers," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 65, no. 1, pp. 283-292, 2017.

[21] Y. Du, F. Liu, J. Qiu, and M. Buss, "A Semi-Supervised Learning Approach for Identification of Piecewise Affine Systems," *IEEE Transactions on Circuits and Systems I: Regular Papers,* 2020.

[22] M. Shirazi, R. Zane, and D. Maksimovic, "An autotuning digital controller for DC–DC power converters based on online frequency-response measurement," *IEEE Transactions on Power Electronics,* vol. 24, no. 11, pp. 2578-2588, 2009.

[23] B. Miao, R. Zane, and D. Maksimovic, "System identification of power converters with digital control through cross-correlation methods," *IEEE Transactions on Power Electronics,* vol. 20, no. 5, pp. 1093-1099, 2005.

[24] T. Roinila, T. Helin, M. Vilkko, T. Suntio, and H. Koivisto, "Circular correlation based identification of switching power converter with uncertainty analysis using fuzzy density approach," *Simulation Modelling Practice and Theory,* vol. 17, no. 6, pp. 1043-1058, 2009.

[25] C. Wang, M. Armstrong, S. Gadoue, and P. Missailidis, "System identification of a DC-DC converter system using a Fast Affine Projection algorithm," in *7th IET International Conference on Power Electronics, Machines and Drives (PEMD 2014)*.

[26] B. Li and K. Low, "Low sampling rate online parameters monitoring of DC–DC converters for predictive-maintenance using biogeography-based optimization," *IEEE Transactions on Power Electronics,* vol. 31, no. 4, pp. 2870-2879, 2016.

[27] M. Kanzian, H. Gietler, C. Unterrieder, M. Agostinelli, M. Lunglmayr, and M. Huemer, "Low-Complexity State-Space-Based System Identification and Controller Auto-Tuning Method for Multi-Phase DC–DC Converters," *IEEE Transactions on Industry Applications,* vol. 55, no. 2, pp. 2076-2087, 2018.

[28] H. J. Tan, S. C. Chan, J. Q. Lin, and X. Sun, "A New Variable Forgetting Factor-Based Bias-Compensated RLS Algorithm for Identification of FIR Systems With Input Noise and Its Hardware Implementation," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 67, no. 1, pp. 198-211, 2019.

[29] M. Algreer, M. Armstrong, and D. Giaouris, "Active online system identification of switch mode DC–DC power converter based on efficient recursive DCD-IIR adaptive filter," *IEEE transactions on power electronics,* vol. 27, no. 11, pp. 4425-4435, 2012.

[30] Y. Zhang and Q. Xu, "Adaptive Sliding Mode Control With Parameter Estimation and Kalman Filter for Precision Motion Control of a Piezo-Driven Microgripper," *IEEE Transactions on Control Systems Technology,* vol. 25, no. 2, pp. 728-735, 2017.

[31] J. Liu, Y. V. Zakharov, and B. Weaver, "Architecture and FPGA design of dichotomous coordinate descent algorithms," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 56, no. 11, pp. 2425-2438, 2009.

[32] M. Ahmeid, M. Armstrong, M. Al-Greer, and S. Gadoue, "Computationally efficient self-tuning controller for dc–dc switch mode power converters based on partial update Kalman filter," *IEEE Transactions on Power Electronics,* vol. 33, no. 9, pp. 8081-8090, 2017.

[33] D. Maksimovic and R. Zane, "Small-signal discrete-time modeling of digitally controlled PWM converters," *IEEE transactions on power electronics,* vol. 22, no. 6, pp. 2552-2556, 2007.

[34] L. Corradini, D. Maksimovic, P. Mattavelli, and R. Zane, "Continuous-time averaged modeling of DC-DC converters," 2015.

[35] P. Navrátil and V. Bobál, "Recursive identification algorithms library," in *Proceedings17th International Conference on Process Control*, 2009, pp. 516-523.

[36] M. Ahmeid, M. Armstrong, S. Gadoue, M. Al-Greer, and P. Missailidis, "Real-Time Parameter Estimation of DC–DC Converters Using a Self-Tuned Kalman Filter," *IEEE Transactions on Power Electronics,* vol. 32, no. 7, pp. 5666-5674, 2017.

[37] J. Xu, M. Armstrong, and M. Al-Greer, "Computational Burden Reduction in Real-Time System Identification of Multi-Rail Power Converter by Re-using Covariance Matrix Approximation," in *2020 IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2020: IEEE, pp. 2150-2157.

[38] J. Castello and J. M. Espi, "DSP implementation for measuring the loop gain frequency response of digitally controlled power converters," *IEEE Transactions on Power Electronics,* vol. 27, no. 9, pp. 4113-4121, 2012.

[39] J. Morroni, L. Corradini, R. Zane, and D. Maksimovic, "Adaptive tuning of switched-mode power supplies operating in discontinuous and continuous conduction modes," *IEEE Transactions on Power Electronics,* vol. 24, no. 11, pp. 2603-2611, 2009.

## APPENDIX OF ABBREVIATION

| | |
|---|---|
| AP | Affine Projection |
| AE | Average Error |
| CT | Correction Term |
| CMA | Covariance Matrix Approximation |
| DCD-RLS | Dichotomous Coordinate Descent – Recursive Least Square |
| DSP | Digital Signal Processor |
| ESR | Equivalent Series Resistance |
| FAP | Fast Affine Projection |
| FPGA | Field-Programmable Gate Array |
| λ | Forgetting Factor |
| GV | Gradient Vector |
| IF | Iteration Frequency |
| KF | Kalman Filter |
| KG | Kalman Gain |
| LMS | Least-mean Square |
| MAFs | Moving Average Filters |
| NG | New Guess |
| POL | Point of Load |
| PE | Process Error |
| PRBS | Pseudo Random Binary Sequence |
| RLS | Recursive Least Square |
| SF | Sampling Frequency |
| SALS | Step-adaptive Approximation Least Squares |
| ε | Step Size |
| SMPC | Switch Mode Power Converter |