On Real-Time Simulation for Harmonic and Flicker Assessment of an Industrial System With Bulk Nonlinear Loads

Gary W. Chang, *Fellow, IEEE*, Yu-Jen Liu, *Member, IEEE*,
Venkata Dinavahi, *Senior Member, IEEE*, and Huai-Jhe Su

Abstract—This paper presents application experiences of real-time simulation (RTS) techniques for harmonic and flicker studies of an industrial power system, where the system and nonlinear loads are properly modeled. A PC-cluster-based real-time parallel simulator is implemented under MATLAB/SIMULINK, where the studied system consists of an ac electric arc furnace, dc and ac motor-drive loads, and the static var compensator. Guidelines for model partition of the studied system and the solver settings under an RTS environment are reported. In addition, the most commonly used offline simulation with variable-step solver and the actual field measurements are included for comparison. Results indicate that the RTS achieves satisfactory solution accuracy within much less execution time and can be applied for more complicated studies such as installing new nonlinear loads with different levels of model complexities or designing/tuning mitigation devices of power-quality disturbances, where the repeated time-consuming analysis is required.

Index Terms—Electric arc furnace (EAF), flicker, harmonics, motor drives, power quality (PQ), real-time simulation (RTS).

I. INTRODUCTION

THE INCREASING use of power electronic and arcing equipment, such as adjustable-speed motor drives (ASDs) and electric arc furnaces (EAFs), in the industrial environment has led to a growing concern for power-quality (PQ) disturbance and causing considerable impacts on the system and its equipment [1]. Literature survey shows that PQ studies associated with ASDs and EAFs mostly focus on harmonic and flicker assessments and the modeling evaluation [2]–[8], where many of the proposed methods are in the time domain due to the advantages of modeling interactions between the supply system and the nonlinear loads. The most familiar time-domain method is to perform the offline simulation (OLS) by using software packages, such as MATLAB/SIMULINK, PSCAD/EMTDC, SPICE, SABER, and EMTP-RV, so that the

transient or steady-state results can be acquired. In this case, any electric system developed by these simulation tools are further converted to differential equations or state-space matrices, and a fixed-step or variable-step numerical-integration method is used for solving them. Meanwhile, the choice of a suitable time-step may influence the accuracy of the time-domain simulation. Although many applications of traditional OLS for PQ studies have been reported, it is also well known that the OLS takes a long time to simulate a large nonlinear system with a small submicrosecond time step. The time-consuming process of OLS is attributed to factors including the underlying solution algorithm, the sequential execution, and the simulator hardware configuration. For example, it takes a long time to achieve satisfactory results when a complex and stiff system is simulated with variable step size because of a large number of differential equations or state-space matrices of the system to be computed for each circuit topology being considered during simulations. To alleviate the inaccuracy due to interstep switching events in EMTP-type offline tools, over the years, several compensation methods have been developed using techniques such as interpolation [9]. In offline electronic circuit simulation tools such as SABER, dynamic switches in conductions typically have small time constants, and thus, discretization with a small fixed time step is often required so that the switching phenomena can be observed in detail. Moreover, PQ analyses for systems with tens to hundreds of harmonics-producing loads often require detailed simulation to assess their impact on an individual or collective basis. There are many instances, such as the design, implementation, and tuning of devices for mitigation of PQ disturbances, and new installation of large nonlinear loads or compensation devices such as ASDs, EAFs, and static var compensators (SVCs) that require multiple repeated simulations for collecting data on various types of test scenarios. Furthermore, the capability of interfacing with real-world devices for hardware-in-the-loop (HIL) test is nonexistent for traditional OLS techniques.

With the advances of computer and communication technologies, real-time simulation (RTS) techniques have matured. The significant requirement of RTS is to ensure that the calculation for a time step is accomplished within the chosen step size. To achieve the goal of computational efficiency, RTS includes some advanced simulation techniques, such as the power of parallel computing, partition of complex model, time-step synchronization and effective solution algorithm, and

the switching-event correction for power electronic modeling. Since all the computational resources are effectively used and the simulation process becomes more optimized, RTS results in more efficient simulation compared with OLS with the same time step. Thus, the RTS can simulate a larger nonlinear system under the limitations of real-time hardware. It must be noted however that most of the aforementioned advanced simulation techniques are increasingly being employed in OLS tools as well to gain higher accuracy and speed.

In recent years, RTS has been applied for many system studies [10]–[25]; however, applications in the PQ area, particularly for harmonic and flicker analyses of a high-voltage system, are rarely found and are limited to small test systems [8], [15], [16]. The purpose of this paper is to present a detailed experience of adopting RTS techniques for harmonic and flicker studies of a sizable realistic industrial power system, where the system network and bulk nonlinear loads are properly modeled in the simulation. With a combination of computer hardware and efficient software for parallel simulation, the impact of PQ disturbances caused by multiple nonlinear loads in the studied system can be efficiently observed through RTS, and effective countermeasures can be designed and implemented in a timely manner.

The organization of this paper is as follows. In Section II, a description of a PC-cluster-based real-time parallel simulator is presented. The parallel computing techniques, synchronization modes, model-partition guidelines, and solver settings are addressed in Section III. Representative test results are illustrated in Section IV to show the usefulness of RTS for harmonic and flicker studies followed by the conclusion in Section V.

II. IMPLEMENTATION OF A PC-CLUSTER-BASED REAL-TIME PARALLEL SIMULATOR

The model-based OLS performed by the widely used simulation packages generally provides a host of functions for PQ studies. Many useful electric modules (or blocks) are offered by such packages so that users can easily build the problem model. In the following, a real-time simulator setup, including the requirements for hardware, software, and communication that can work with MATLAB/SIMULINK, is described. The discussion of using proper time-step and the simulation modes for various PQ studies are also presented.

A. Hardware and Software Architecture for the Simulator

Fig. 1 shows the configuration of the cluster-based real-time simulator. The simulator is constructed under MATLAB/SIMULINK environment, which includes two groups of computers, *target cluster* and *host*, with a multicore PC with Intel Core 2 Quad 2.4-GHz CPU and 1.0-GB random access memory (RAM) being the *target* cluster, and another multicore PC with an Intel Core 2 Duo 1.86-GHz CPU and 1.0 GB RAM being the *host*. The *target cluster* works as four computation engines inside the simulator responsible for real-time execution of the models, communication between cluster nodes, data acquisition and delivery between *target cluster* and *host*, and computational tasks, as well as interfacing with external hardware. A real-time operating system (RTOS) is required for the *target cluster* to manage the assignment of the utilized resources
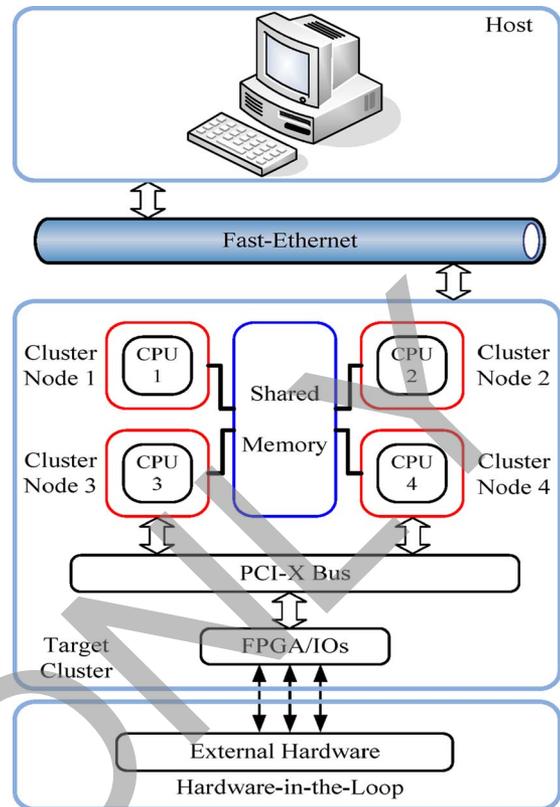

Fig. 1. Configuration of cluster-based real-time simulator.

of the simulator. An open-source RTOS is chosen to run the *target cluster*. The *host* PC runs on Windows XP operating system and is in charge of control and observation of the simulation processes, where MATLAB/SIMULINK, SimPower-System blockset (SPS), and other blocksets are installed for model development, compilation, code loading, and monitoring outputs. A real-time interfacing software RT-LAB that runs on the *host* also plays a vital role, which is a scalable simulation and control package that works with SIMULINK to provide an effective capability for the execution of parallel computing. In addition, a toolbox add-on for SPS called Advanced Real-Time Electro-Mechanical Simulator (ARTEMIS, [23]) is introduced that offers an alternative to the solver of SPS and provides enhanced nonlinear solution algorithms for supporting real-time implementation of the system simulations and increases numerical stability.

B. Communications

An effective communication between the *target cluster* and the *host* is required. The following describes several types of communication techniques that are adopted in the simulator.

1) *Fast Ethernet Link*: An Ethernet protocol with 100-Mb/s data rate called *Fast Ethernet* has been established to carry out the communication and data conversion between the *target cluster* and the *host*.

2) *Shared Memory*: The communication between each cluster node of the simulator is made by *Shared Memory* which provides a large space of RAM that can be accessed by several CPUs in a multiprocessor architecture.
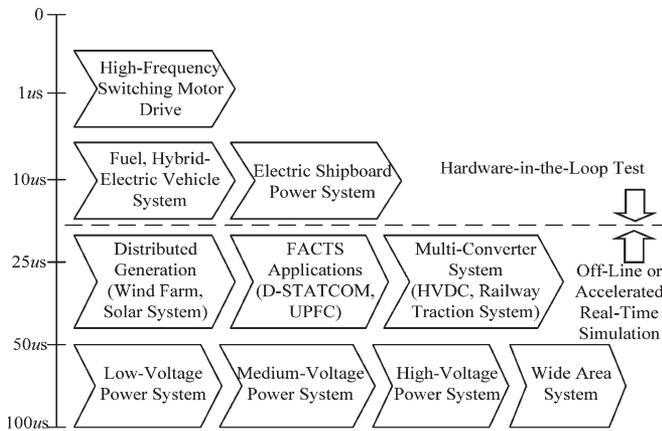
Fig. 2.   Selection of time-step sizes for various PQ simulations.

3) *OpComm*: A complex electric system model that is developed by MATLAB/SIMULINK needs to be modified so that the model can be appropriated separately and be converted to the RTS environment. The built-in communication module, named *OpComm*, in the RT-LAB is thus necessary to be inserted into each distributed subsystem model for effective connection and data acquisition in real-time status.

C. Selection of Proper Time-Step Size

The choice of applicable time-step is crucial for most time-domain simulations. Using an acceptable time-step size that can be achieved by available simulation techniques may ensure the desired solution accuracy. Typically, a time-step of 50 to 100 μs is able to obtain an accurate PQ simulation if low-frequency electric responses are considered. However, for the simulation (or HIL test) of the switching power electronics and motor drives, such time-step size may not be sufficiently low to account for the switching procedure that occurs between two simulation time steps, and it may lead to considerable errors in results. These high-speed switching elements need very small time steps to reach acceptable accuracy. The selection of the time step also depends on the switching frequency and the number of switch components.

Literature survey indicates that the use of time-step sizes depends on various PQ applications, as shown in Fig. 2. For instance, a time step of 50 to 100 μs is commonly required for typical power system simulations. The applications, including distributed generations, FACTS devices, and multiconverter systems, often consider a time-step within 20 to 50 μs. Most of the aforementioned studies are present in OLS or accelerated RTS. For electric systems involving high-speed switching power electronics or motor drives, a time step of about 10 μs or less is widely used in an HIL test.

D. Available Simulation Modes

The presented simulator offers three simulation modes to deal with various application needs. These simulation modes are shown in Fig. 3.

1) *General OLS*: The *target cluster* in this case is not active, and the whole simulation is performed only on the *host*

PC. Such implementation is like simulating on a general stand-alone PC with a typical simulation procedure (i.e., model editing, simulation parameter setting, simulation running, and results collecting).

2) *Accelerated RTS*: It is an application of soft real time which performs software synchronization with the timer offered by the RTOS. The *Target cluster* in this mode is active, thus, the simulation may be performed on the whole simulator. Meanwhile, the computational tasks of the simulated system are assigned to different computing nodes, and utilized resources can be effectively managed by the RTOS, which in turn provides parallel computing for the simulations.

3) *HIL Simulation*: A simulated plant and control devices are always considered in the previous two modes. However, these modes have difficulty in showing the interaction between the plant and the controller that is close to reality. HIL simulation is an added benefit for interfacing external hardware. HIL can be recognized as a hard real-time application which needs hardware synchronization, and the completion of a computation with maximum accuracy can satisfy its timing constraint. A commonly seen setup of HIL simulation in motor drives is the power circuit, which includes ac/dc converter, dc/ac inverter, and the motor. Moreover, a real controller includes a control module, and a pulse-width modulation (PWM) generator board is connected with I/O to simulate the plant. During operation, the gate signals are generated from the external controller under test.

III. PARALLEL COMPUTING AND SYNCHRONIZATION MODES

Parallel computing is one of the significant features that speeds up the execution in real time and is the simultaneous use of multiple computing resources to finish a computational task. The performance of the parallel-in-space technique is introduced in the simulator which separates the complete system model into several subsystem models so that the computational burden can be distributed over each cluster node to achieve accelerated simulation. There are two critical functions to perform parallel computing: model partition and synchronization. For the model partition, after the complete system model is developed, this model must be appropriately broken into several subsystem models based on the users' own knowledge so that the computational task of each subsystem model can be assigned to each computation engine on the *target cluster*. Meanwhile, one subsystem model must be named subsystem master (SM) and the others are recognized as subsystem slaves (SS). A subsystem console (SC) is required whose operation is not performed on the *target cluster*; it resides on the *host* to receive and display the output data to the scope modules or an external oscilloscope, and to send commands to the RTS being executed on the *target cluster*.

Two frameworks, "natural" parallelism and "forced" parallelism, for model partition can be achieved in the simulator. A model with natural parallelism is one where a model can be distributed into several subsystem models that do not interact with
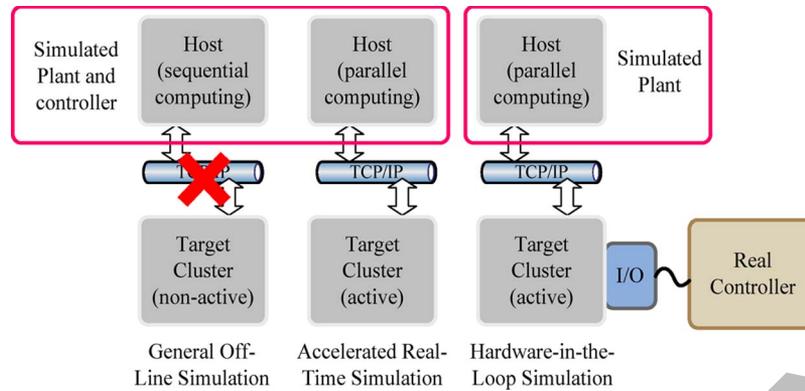
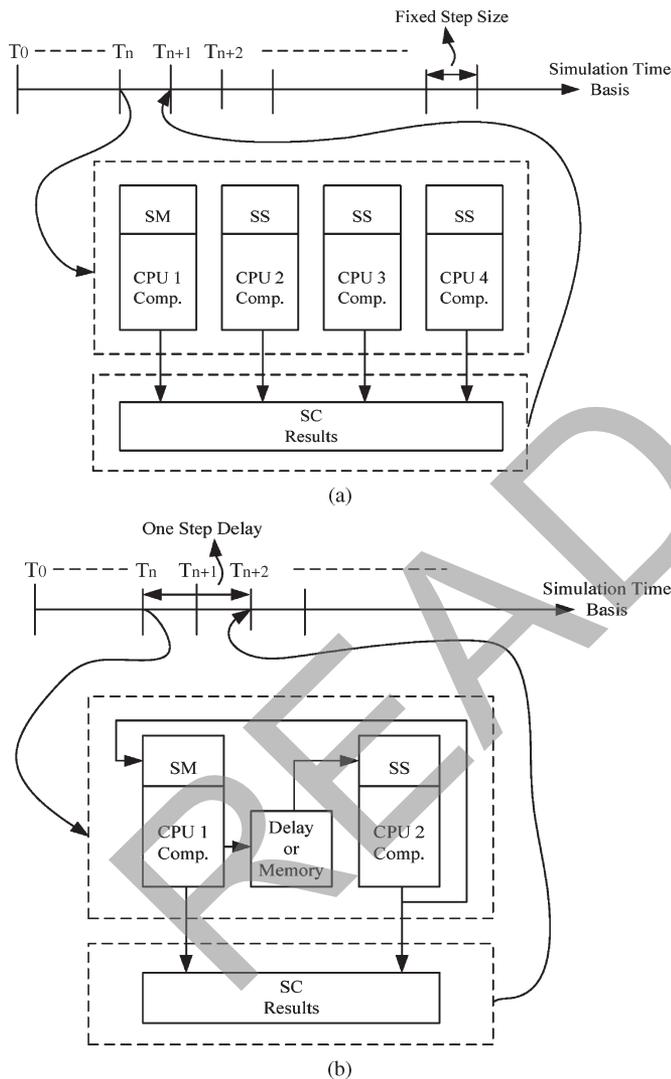Fig. 3. Available simulation modes of cluster-based real-time simulator.

Fig. 4. Parallel methodologies. (a) Natural parallelism. (b) Forced parallelism.

each other except at the time when collecting their respective outputs. Fig. 4(a) shows a sketch of the natural parallelism used in this study, where one model is distributed into four subsystem models; meanwhile, the computation of each model will run on different cluster nodes (i.e., computation engines or CPUs) without any interactions, and each computation may take one

or less than one time step. For some applications, the forced parallelism may be used to replace natural parallelism for RTS. For example, if the SM in Fig. 4(b) is an electric system or a plant and the SS is a feedback controller, each subsystem model may interact with each other when performing some parallel tasks. Assume that the delay or memory block is currently ignored in Fig. 4(b) (the computation between SM and SS without any continuity); the SS will stop computing after the output of the SS has been sent to the SM until the next input comes from SM. Similarly, once the output of SM has been sent to SS, it will stop computing until the next input comes from SS. Under this condition, the capability of parallel computing may be lost. Thus, a delay or memory block is inserted to overcome this problem, which ensures that the computations in SM and SS are continued without any interruptions. Moreover, a one-step delay may be shown in the results, which is inevitable for implementing parallel computing using the forced-parallelism method, and the acceptable step delay which does not influence the performance of RTS can be determined by the users.

The technique of model partition still presents a challenge when applying the studied real-time simulator which currently does not support any automatically or half-automatically partitioning methods for decomposing the system model. The users must separate the model by their own knowledge. The following remarks are suggested to be the general guideline to obtain a balanced model partition for parallel computing on the real-time simulator.

1) Partition of power electronic switching circuit: A switch component, such as a diode or a thyristor, is expressed as a two-state (ON/OFF) device. Since the circuit topology depends on the switch state, a circuit comprising n switches may entail 2^n topologies in theory. RTS can precompute the possible states of these switch circuits without repeated considerations of each circuit topology during simulation. However, these precomputed switch states may be stored in a big number of state-space matrices on a PC memory; while running the simulation, the program may suffer considerable computational overhead on accessing these switch states. It is thus recommended to partition the circuits that consist of power electronic switches, such as power converter/inverter or other thyristor-controlled devices, as an individual subsystem model. It must be noted that some parallel
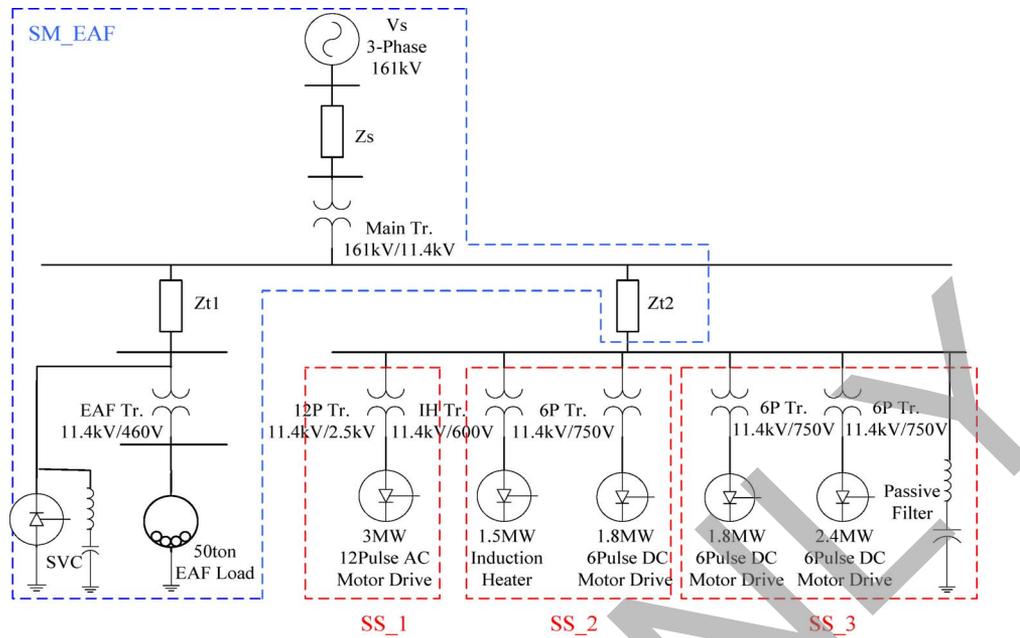
Fig. 5. One-line diagram of the steel plant electric system under study.

simulation tools suggest that model partitioning should be based on the nature of signals exchanged between partitions. In such a case, partitioning should be chosen such that the added time delay between partitions has a minimal impact on the simulation accuracy. Yet, another way of choosing partition points (in models that do not involve power electronic switching components) is by tracking slow-moving signals that vary little over the simulation time step.

2) Distribution of mathematical units and logical operators: For many model-based control-system simulations, a significant number of elements, including mathematical units and logical operators, are always used to implement the control algorithm. For example, the PWM pulse generator needs some logical operators to compare modulating and carrier signals so that the firing pulse can be produced; certain proposed control strategies must be carried out by multipliers, differentiators, and integrators, as well as other mathematical operations. While running the simulation, a heavy computational overhead may present in these elements. Hence, it is suggested to distribute the circuit that includes many of such elements as an individual subsystem model.

3) CPU workload check: The RTOS provides an indication diagram of the CPU workload which can be used to estimate the computing capability of each cluster node, and an efficient computation will be achieved when the workload of each CPU is balanced.

For synchronization, it is often necessary to provide a synchronization point within an RTS application where a task may not proceed until another task meets the same or logically equivalent point. There are three synchronization modes provided in the simulator: free-run simulation, hardware synchronization, and software synchronization. There is no synchronization being done while the simulation is performed in

the free-run mode. In the software-synchronization mode, the models are synchronized with an internal RTOS timer that contains a limited resolution and is suitable for soft RTS applications. In the hardware-synchronization mode, the models are synchronized with an external hardware timer on an FPGA card with a 10-ns resolution and, thus, can provide a very small timing for hardware synchronization while implementing hard RTS for HIL testing. Since the study presented in this paper does not involve any hardware testing, the software-synchronization mode was adopted for soft RTS. Meanwhile, only one computation node (SM) is synchronized when executing a distributed simulation; the other nodes are synchronized by a designated real-time link between the SM and SSs, and TCP/IP is used for communication between the SM and SC.

IV. CASE STUDY: RTS FOR HARMONIC AND FLICKER ASSESSMENT OF AN INDUSTRIAL SYSTEM

Large steel plants are often sources of harmonics and flickers. In this study, the system of a realistic steel plant is under test to assess the harmonics and flicker with the described RTS techniques. Fig. 5 shows the one-line diagram of the test system, where the system consists of two electric loops: EAF/SVC and rolling mill (RM) loops. The EAF/SVC loop includes a 50-ton EAF load for steel making and an SVC for reactive-power compensation. The RM loop includes a bulk of ac and dc motor drives which are responsible for controlling the width and the thickness of the refined steel product. Results obtained from the traditional OLS with variable-step solver and actual field measurements are included to validate the performance of RTS on harmonics and flicker assessment.

A. System Modeling

In the test system, an ideal three-phase ac voltage source model is used to feed electric power to the system, and several
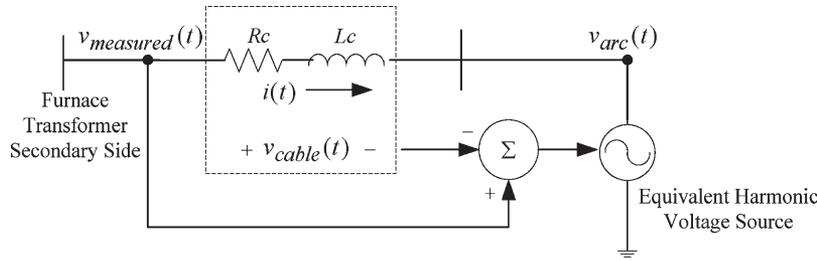
Fig. 6.   Equivalent harmonic voltage-source model for the ac EAF.

linear transformer models are connected to manage the voltage level at each transmission bus. For a system with EAF loads, it is difficult to describe the arcing behavior of the EAF since the arcing is a nonstationary stochastic process that provides a highly nonlinear v–i characteristic of the EAF. In the literature, it is common to use a steady-state model to describe the operation of the EAF during the steel-refining period [6], [7]. Although the EAF can be modeled by complicated models which represent its v–i characteristic more precisely, our goal is to show the application of RTS for PQ assessment of the studied system, and, thus, only the harmonic voltage-source model is adopted. As shown in Fig. 6, the EAF load is modeled as a steady-state harmonic voltage source, which is considered up to the ninth harmonics, behind lumped impedance (i.e., R_c and L_c) which represents the cable impedance from the EAF transformer secondary side to the electrode, and the effect of frequency dependence is ignored. However, it is found that the measured arc voltage $v_{\text{measured}}(t)$ in the real plant can only be obtained at the secondary side of the furnace transformer due to the difficulty of measuring the arc voltage at the furnace body. Moreover, because the measured arc voltage obtained at the furnace transformer secondary side already includes the voltage of the cable impedance, the voltage across the cable $v_{\text{cable}}(t)$ must be deducted in the model so that the EAF arc voltage can be properly modeled in the simulation, as given in

$$v_{\text{measured}}(t) = \sum_{h=1}^{H} V_h \sin(h\omega_0 t + \theta_h) \tag{1}$$

$$v_{\text{cable}}(t) = L_c \frac{\mathrm{d}i(t)}{\mathrm{d}t} + R_c i(t) \tag{2}$$

$$v_{\text{arc}}(t) = v_{\text{measured}}(t) - v_{\text{cable}}(t) \tag{3}$$

where H is the considered highest order harmonics.

For compensating the reactive-power consumption of the EAF, the SVC, composed of delta-connected thyristor-controlled reactors (TCR) in parallel with fixed-capacitor banks, which often set up as LC filters to absorb harmonic currents produced by the TCR, is installed in the studied system. Once the EAF operates, the SVC acts like a shunt-connected variable reactance, which either generates or absorbs reactive power to regulate the voltage at the connection point to the inner network of the plant. Fig. 7 shows the single-phase equivalent of the SVC model with parallel passive LC filters. In the simulation, a fixed firing angle is adopted to control the TCR thyristors and to supply a constant reactive power for compensation.
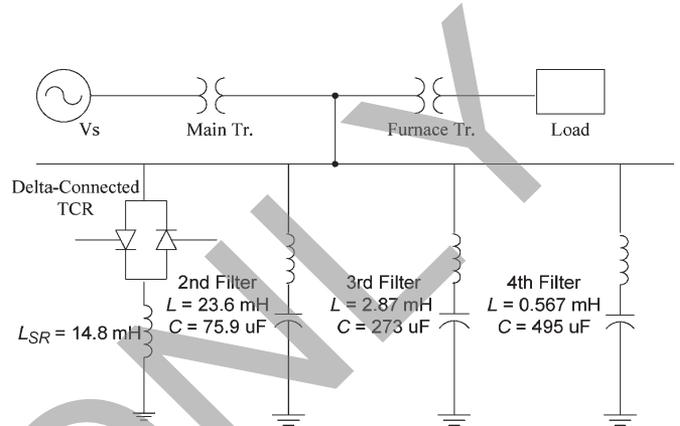

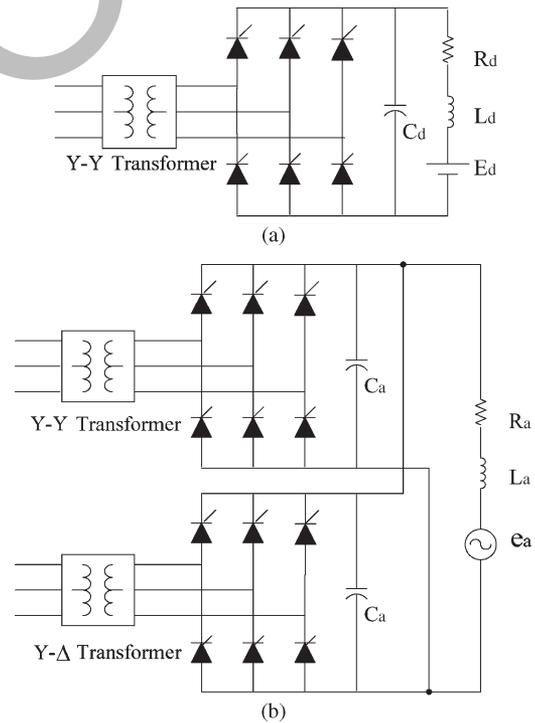Fig. 7.   Equivalent single-phase representation of the SVC in Fig. 5.

Fig. 8.   Equivalent motor-drive circuits of Fig. 5. (a) Six-pulse dc motor drive. (b) Twelve-pulse ac motor drive.

Fig. 8(a) shows the circuit of the six-pulse ac/dc thyristor bridge rectifier for dc motor drives, where the motor load is modeled as equivalent impedance in series with a back EMF. The dc-link capacitor is also connected to form a low-pass filter for reducing the ripple. Moreover, two 6-pulse rectifiers are connected in parallel, as shown in Fig. 8(b), as a front-end converter of the 12-pulse ac motor drive. The source side of

TABLE I
PARAMETERS OF DC AND AC MOTOR DRIVES

	R_d (Ω)	L_d(mH)	C_d(uF)	E_d(V)	α
DC_2.7MW	0.126	0.16		455	25.5°
DC_1.8MW	0.132	0.155	1500	405	25.7°
DC_2.4MW	0.148	0.25		420	25.3°
	R_a (Ω)	L_a(mH)	C_a(uF)	e_a(V)	α
AC_3MW	2	0.235	3000	225∠ -12.4°	20°

α: firing angle

TABLE II
SOLVER SETTINGS FOR OLS AND RTS

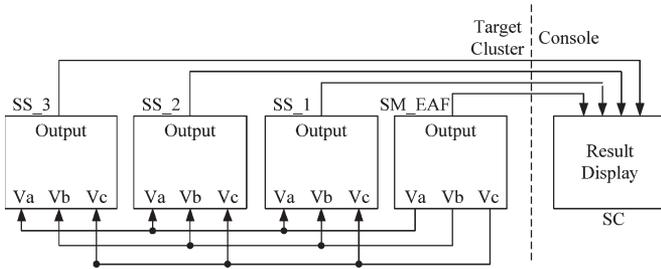| Type \ Item	Off-Line	Real-Time
Solver	ode23tb (stiff/TR-BDF2)	ARTEMIS art3hd algorithm
Step Type	Variable-Step	Fixed-Step
Time-step	Auto	100 µs
Simulation Time Length	0.6 sec	Infinite-run but assigned to stop at 0.6 sec

Fig. 9. Schematic diagram of model partition for the system in Fig. 5.

the two rectifiers is provided with two 3-phase two-winding transformers, while the secondary sides are with a 30° phase shift. The back-end inverter in the 12-pulse motor drive is modeled as equivalent impedance in series with an ac source. Table I lists the parameters associated with the motor drives. For the six-pulse and 12-pulse motor drives, a synchronized firing-angle generator is modeled to trigger the thyristors with an equidistant firing scheme. However, if more complicated switching techniques (e.g., PWM) are required for controlling the firing angles [5], an additional add-on time-stamped (TS) block of SIMULINK, which is called RT-EVENTS, must be adopted to replace SPS for real-time compensation of the switch states of the electric system when switching events occur in the middle of the time steps [13]. The inclusion of the TS block can capture and label the incoming gating signals providing a superior interpolation mechanism to reduce numerical errors.

B. Model Partition

The complete system model shown in Fig. 5 is distributed into four subsystem models, one SM and three SSs. Then, the computation of each subsystem model can be performed in parallel on different cluster nodes to accelerate the simulation. By following the model-partition guideline described in Section III, a separated model of the complete system is shown in Fig. 9. In this model, SM_EAF includes a three-phase source, components of power transmission, and components of the EAF/SVC loop; for the RM loop, a 12-pulse ac motor drive is modeled in SS_1; SS_2 consists of two 6-pulse dc motor drives, and SS_3 contains a six-pulse dc motor drive and an induction heater. In addition to SM and SS, an SC is also required for observing the outputs of the simulation results. The operation of SC is not performed on the *target cluster*; it only receives and displays the output data to scope blocks or an external oscilloscope by the communication link that connects the *host*

with the *target cluster*. The parameters used in each model are collected from actual system data.

C. Solver Setting

To estimate the performance of the RTS, the traditional OLS is included in the study as well. The solver settings for OLS and RTS are listed in Table II. Because the simulated system includes many dynamic components, such as power electronic components and time-varying models, a great variation of system topologies may exist on the associated differential equations that are required for describing the system model. The solver ode23tb is thus used for solving such stiff nonlinear differential equations in OLS, which is an implementation of TR-BDF2, an implicit Runge–Kutta formula with a first stage that is a trapezoidal-rule step and a second stage that is a backward differentiation formula of the order two. On the other hand, a fixed-step solver is necessary for RTS, but the solvers supported by the SPS cannot meet the need of the simulator. Therefore, the ARTEMIS algorithm art3hd is adopted for solving the studied power-system model [9], [23], [24]. Other than the OLS solver that requires very tiny time-step sizes to obtain the same solution accuracy, the art3hd is a fixed time-step-size integration method which provides a discretization and decoupling technique; it is based on a suitable order of Padé approximation of matrix exponentials and provides an L-stable method that is free of numerical oscillation for a wide range of step sizes [25]. Moreover, with the capability of the interpolation technique supporting the algorithm, an accurate detection of the switching events occurring at each time-step can be achieved.

Since the study does not involve the HIL test, the detail switching technique (e.g., PWM control) is thus not considered to control the TCR, the six-pulse, and the 12-pulse motor drives, which in turn is not necessary to use a very small time-step for simulation. The time-step size is sometimes constrained by factors such as jitter and latency of the CPU, the clock speeds of the peripheral devices, the communication overhead, and the limited bandwidth of the sensors and actuators. According to the applications described in Fig. 2, a 100-µs step size is chosen and is sufficient for the study. If the concern is to obtain satisfactory accuracy with a much smaller time-step for HIL test, some reported efforts can be found in [9] and [23]. The simulation time duration of 0.6 s is set to perform both OLS and RTS. The general OLS mode is performed only on the *host* PC without using any computing resources in the target PC, while the latter is performed with a whole real-time simulator.
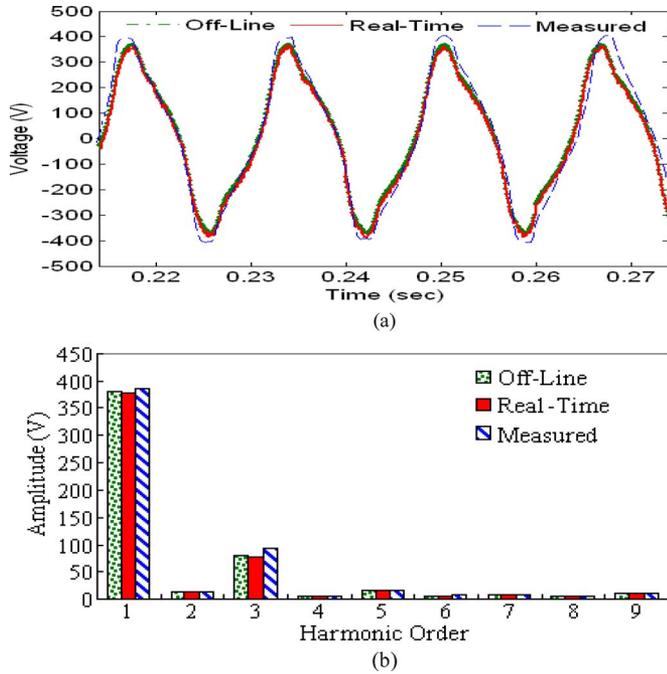
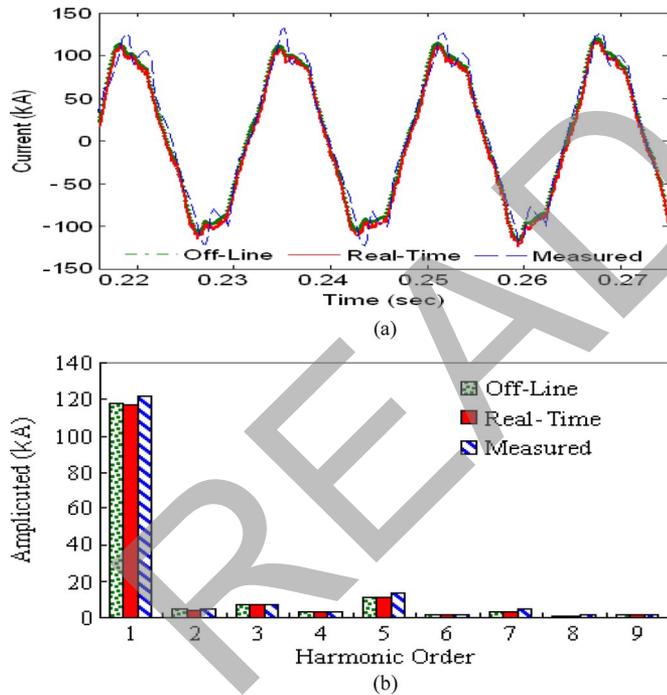Fig. 10. EAF arc voltage. (a) Waveforms. (b) Harmonic spectra.

Fig. 11. EAF arc current. (a) Waveforms. (b) Harmonic spectra.

D. Results

For the harmonic assessment, harmonic currents produced by the EAF and ASDs are investigated first. During the simulation, the harmonic voltage-source model representing the EAF is connected to the system; the distorted arc-voltage and arc-current waveforms at the EAF transformer secondary side are observed. Figs. 10 and 11 show the time-domain waveforms and harmonic spectra that are obtained by the OLS and by the RTS. At the same time, results obtained by actual mea-

surements are also included for validation. To describe the nonlinear behavior of the EAF load, Fig. 12 shows the v–i characteristics during the refining stage. Results of simulations and measurements of a 1.8-MW six-pulse dc motor drive and the 12-pulse ac motor drive are shown in Figs. 13 and 14, where the current waveforms are rich in $6p \pm 1$ and $12p \pm 1$ ($p = 1, 2, 3, \ldots$) characteristic harmonics plus the third harmonic which is mainly due to the slightly imbalanced operation that occurred in the test system.

The chosen time-step size is 100 μs in the RTS. The frequency bandwidth that can be observed is thus up to 5 kHz, although this bandwidth shrinks with the inclusion of nonlinear elements in the simulation. However, the sampling frequency of the PQ meter used for field measurements is set to be 3840 Hz to provide an effective frequency bandwidth when the harmonic components to be observed are up to the 32nd harmonic order. Furthermore, some measured high-frequency harmonic components of the EAF arc and the dc motor-drive currents are relatively small in comparison with their fundamental components, and, thus, they are not shown in the harmonic spectra. Since the PQ study focuses on the harmonics and flicker, the frequency range of up to the 25th order for analysis is generally sufficient. Therefore, the harmonic spectra of the input currents to be observed for the EAF, the six-pulse dc motor drive, and the 12-pulse ac motor drive, are up to the 9th, 15th, and 25th orders, respectively.

Table III indicates the average relative error of the harmonic spectral magnitudes in Figs. 10, 11, 13, and 14, where the actual measurement is adopted as the reference. It is noted that there is less than 1% average error between the two simulation modes. However, the results also show that the average relative errors compared with the actual measurement are in the range of 4% to 6% for both simulation modes. Overall, by observing Table III, it is seen that a fair agreement has been reached between the results of OLS and RTS, in comparison with actual measurements.

For the voltage-flicker assessment, the previously adopted harmonic voltage-source model based on field measurements is insufficient to describe the dynamic voltage fluctuations caused by the EAF load. To model the time-varying behavior of the EAF arc voltages, the sinusoidal and band-limited white-noise laws are thus considered to provide simulated waveforms for evaluation of the stationary and nonstationary flickers, respectively. The arc-voltage variations of sinusoidal and white noise are presented as

$$v_{\text{arc\_sin}}(t) = v_{\text{arc}}(t) \cdot (1 + m \sin \omega_F t) \tag{4}$$

$$v_{\text{arc\_whitenoise}}(t) = v_{\text{arc}}(t) \cdot (1 + m\text{BLW}(t)) \tag{5}$$

where $v_{\text{arc}}(t)$ is derived from the harmonic voltage-source model used in the harmonic assessment, m is the modulation index indicating the flicker intensity, ω_F is the flicker frequency, and BLW is the band-limited zero-mean white noise which is in the range of 4–14 Hz.

The simulated arc voltages with dynamic variations are shown in Fig. 15. To observe the voltage-flicker severity of the offline and real-time simulated waveforms, the recommended approach for flicker assessment according to
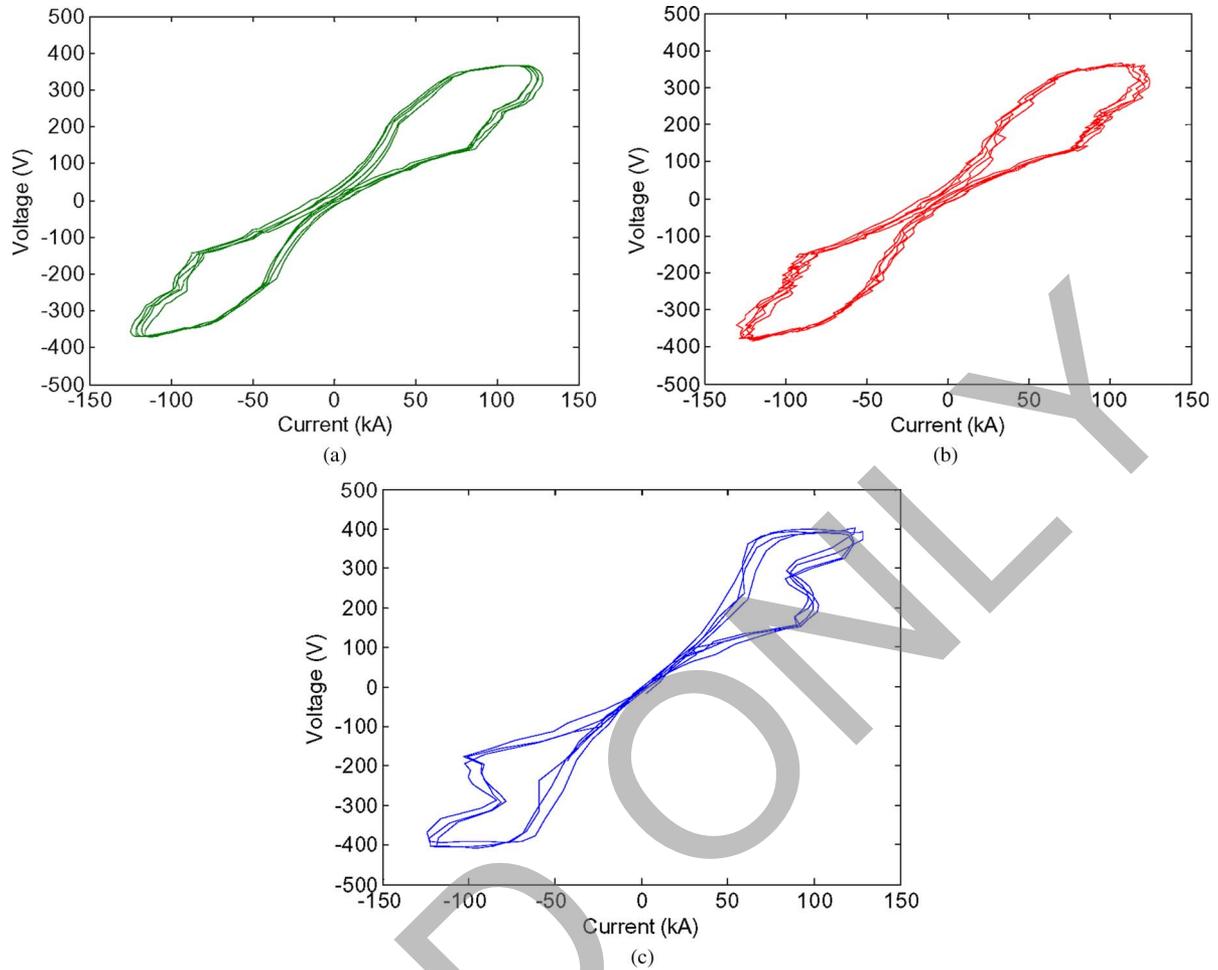
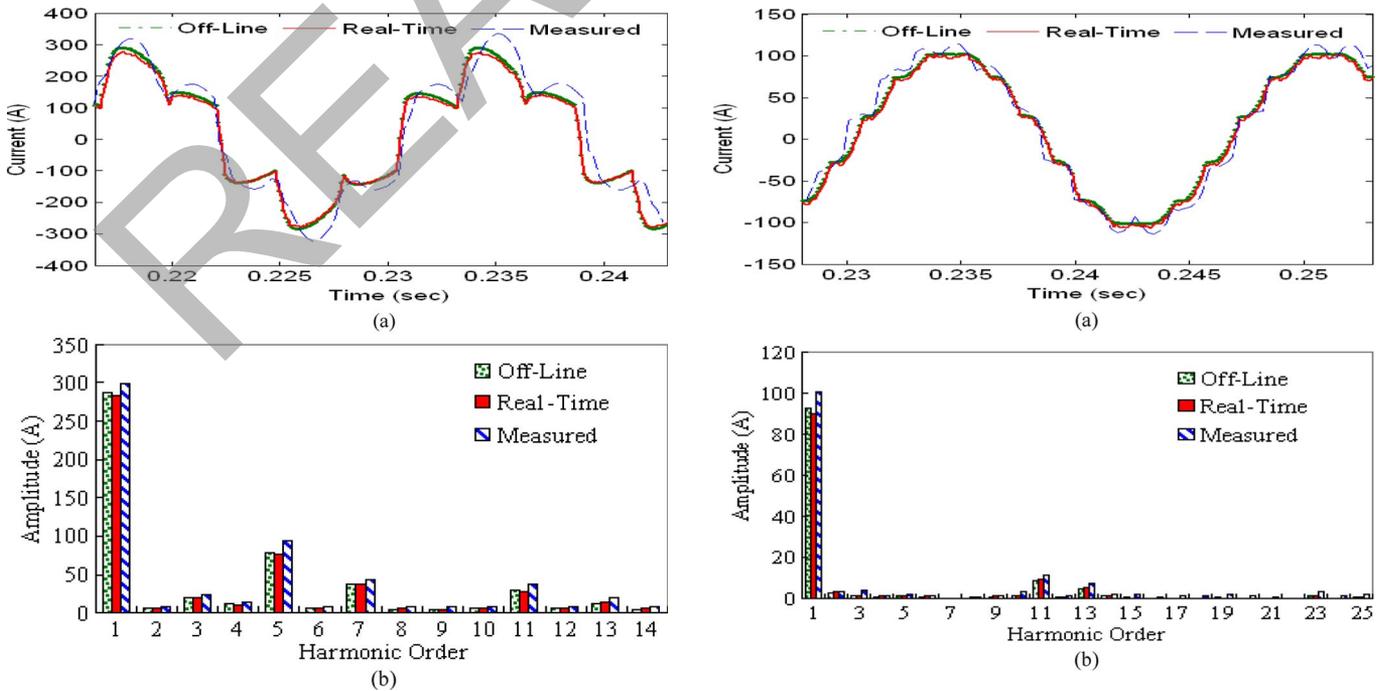Fig. 12. EAF $v–i$ characteristics. (a) OLS, (b) RTS. (c) Measured.

Fig. 13. Current waveforms and harmonic spectra of the 1.8-MW dc ASD.

Fig. 14. Current waveforms and harmonic spectra of the 12-pulse ac ASD.

TABLE III
COMPARISON OF AVERAGE RELATIVE ERRORS BETWEEN OLS, RTS, AND ACTUAL MEASUREMENTS

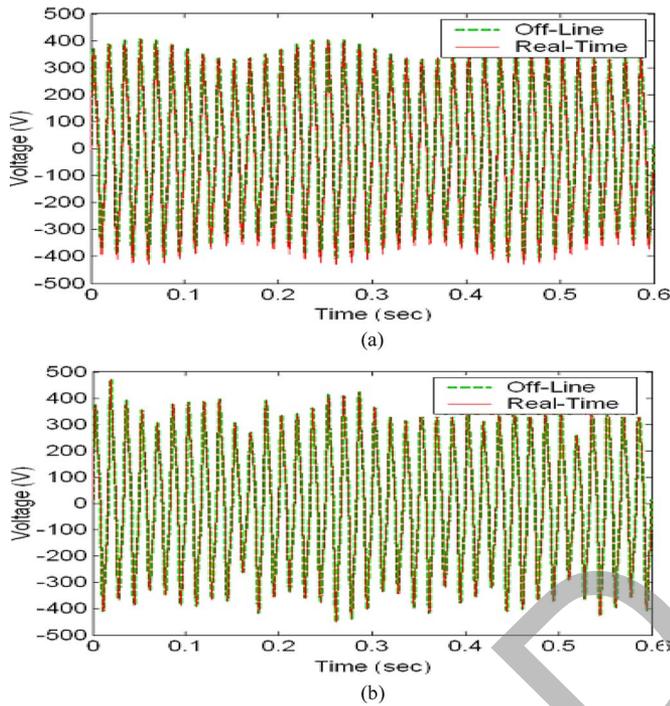| Type — Average Error (%)	Off-Line	Real-Time
EAF Arc Voltage	4.24	5.15
EAF Arc Current	4.78	5.59
6-Pulse Motor Drive Current	4.92	5.75
12-Pulse Motor Drive Current	5.41	5.66

Fig. 15. EAF arc-voltage flicker. (a) Stationary. (b) Nonstationary.

TABLE IV
SHORT-TERM FLICKER SEVERITY OF EAF VOLTAGE OBTAINED BY OLS AND RTS

Simulation	$P_{st\ (stationary)}$	$P_{st\ (non\text{-}stationary)}$
Off-Line	18.79	22.67
Real-Time	18.62	22.44

IEC 61000-4-15 [26] is adopted to obtain the short-term flicker severity P_{st} which is an index commonly used to assess the perception level of flicker. The calculated P_{st} values are given in Table IV. In addition, the dynamic $v–i$ behavior of the EAF may lead to the significant variation of its reactive power consumptions. Fig. 16 shows the simulation results of the reactive power collected at the EAF transformer secondary side. It is found that the variation of reactive power is well compensated while SVC is operating.

Table V lists a comparison of computational performance to achieve satisfactory results for the studied system. It is observed that the execution time of fixed-step OLS using ARTEMIS art3hd is 156.21 s and the variable-step OLS using ode23tb takes more than 90 min, while RTS can take as low as 6.32 s to accomplish the work. It is evident that RTS provides a better benefit on simulation-time manner. By using the same simulator, solver settings, and simulated-time length, the simulated
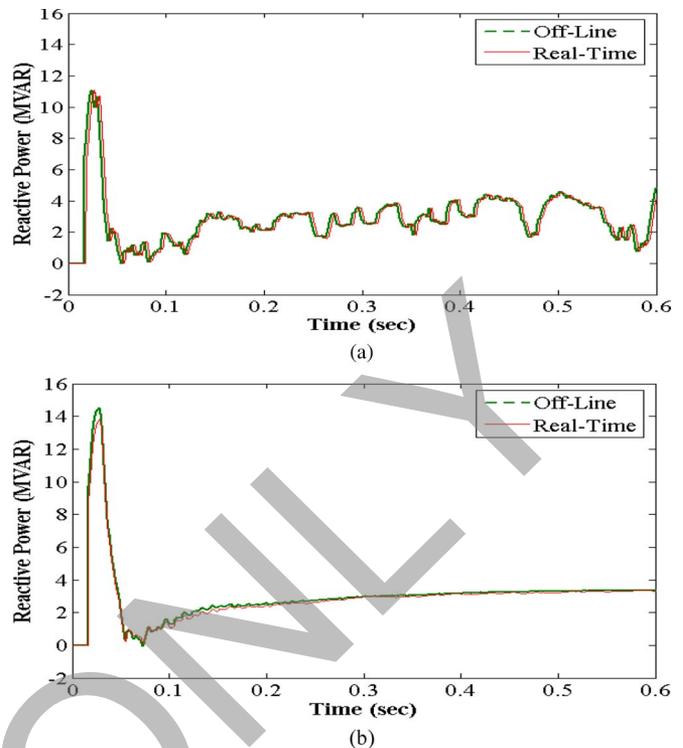

Fig. 16. Variation of reactive power consumption at the 11.4-kV side of the main transformer. (a) Without SVC. (b) With SVC in operation.

TABLE V
EXECUTION TIME FOR DIFFERENT SOLVERS AND MODEL PARTITION ON OLS AND RTS

Off-Line		
Types	Simulated Time	Execution time (sec)
Fixed-Step (ode4)	0.6 sec	Diverge
Fixed-Step (art3hd)		156.21
Variable-Step (ode23tb)		5659.35
Real-Time		
Types	Simulated Time	Execution time (sec)
1SM+3SS+1SC	Infinite-run but assigned to stop at 0.6 sec	6.32
1SM+2SS+1SC		11.15
1SM+1SS+1SC		17.58
1SM+1SC		94.32

time required for different numbers of computing engines for RTS are also included in Table V, where the suggested model-partition type is superior to the other options. It is also seen that, for OLS, the simulation diverges when using the fixed-step ode4 algorithm. Although the use of the fixed-step art3hd algorithm can effectively reduce the execution time while performing stand-alone OLS, it still takes more computational time than that simulated by the RTS due to sequential computing.

As shown in aforementioned results, despite the fact that the solutions of RTS are close to those obtained by OLS, some differences are observed in the results. The differences between actual measurements and simulations are due to the simplified models being adopted and the characteristics of each nonlinear load not being described in detail. For example, the harmonic voltage-source model for the EAF only considers up to the ninth order, and hence, differences are introduced because of neglecting higher order harmonic components. Furthermore, some differences exist since the equidistant firing scheme is
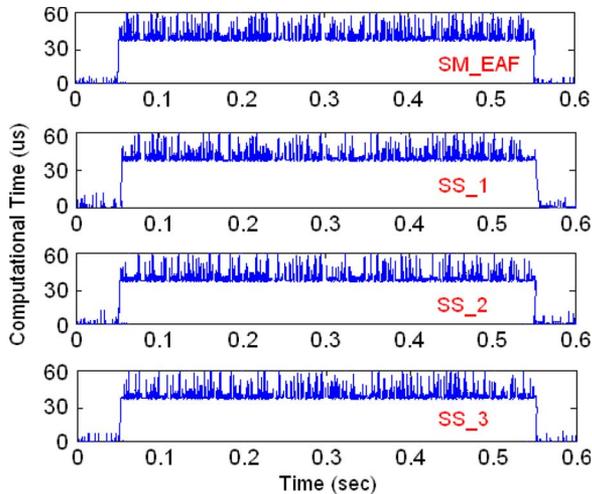
Fig. 17. Computational time of each target node versus simulation time.

adopted for each motor drive, and the input impedance of the motor drives is assumed to be balanced. In addition, the use of different numerical-solution techniques and computing mechanism (sequential or parallel) may influence the solution results and execution performance for RTS and OLS as well.

E. Real-Time Execution

The computational time of four target nodes per integration step is shown in Fig. 17. For the simulation time length of 0.6 s, the average computational time of SM_EAF is about 42.3 μs, and for SS_1, SS_2, and SS_3, they are approximately 39.2, 37.5, and 33.3 μs, respectively, which are accomplished within the assigned step size of 100 μs. The jitters in Fig. 16 are due to the OS in the host PC being required to handle the CPU interruptions of the other processes which create unexpected jitters that may cause a part of the actual simulation execution time exceeds the desired time-step size. In addition, the simulator reserves certain amount of idle time; for SM_EAF, it is 10.48 μs, and for SS_1, SS_2, and SS_3, they are 9.36, 8.17, and 12.53 μs, respectively. The idle times imply that the chosen time-step size may be further reduced. Numerical experiences show that a time-step size of 70 μs still can give satisfactory solution accuracy.

V. CONCLUSION

In this paper, we have demonstrated how a practical PQ study for a realistic industrial system with bulk nonlinear loads can be performed by using a parallel cluster-based real-time simulator. The industrial system includes several variable-speed drives, an ac EAF, an induction heater, passive filters, and an SVC. Such systems are quite time consuming to simulate for PQ studies since they require multiple repeated simulations for collecting data on various types of test scenarios. The RTS system combines improved numerical methods, a parallel algorithm, a parallel compiler, and a parallel hardware which is managed by an RTOS and achieves a satisfactory computational performance by efficiently utilizing all resources.

In this paper, guidelines for the model partition and solver settings based on the authors' application experiences are also

reported. The results obtained from the RTS are in good agreement with the offline and measured field results providing ample confidence in the viability of the adopted simulation approaches. The real-time simulator setup that can work with any widely used mathematical-modeling package possesses a promising potential for different PQ studies. It is also concluded that the PC-cluster-based parallel RTS is an effective enhancement for the traditional OLS to expedite more complicated PQ studies, such as installing new nonlinear loads, validating different nonlinear load models, or designing and tuning PQ mitigation devices in a large system, where the repeated time-consuming analysis is required.

REFERENCES

[1] R. C. Dugan, M. F. McGranaghan, S. Santoso, and H. W. Beaty, *Electrical Power Systems Quality*. New York: McGraw-Hill, 2002.
[2] V. A. Katic, J. M. Knezevic, and D. Graovac, "Application-oriented comparison of the methods for AC/DC converter harmonics analysis," *IEEE Trans. Ind. Electron.*, vol. 50, no. 6, pp. 1100–1108, Dec. 2003.
[3] M. F. McGranaghan and D. R. Mueller, "Designing harmonic filters for adjustable-speed drives to comply with IEEE-519 harmonic limits," *IEEE Trans. Ind. Appl.*, vol. 35, no. 2, pp. 312–318, Mar./Apr. 1999.
[4] M. H. Bierhoff and F. W. Fuchs, "DC-link harmonics of three-phase voltage-source converters influenced by the pulsewidth-modulation strategy—An analysis," *IEEE Trans. Ind. Electron.*, vol. 55, no. 5, pp. 2085–2092, May 2008.
[5] K. Lee, G. Venkataramanan, and T. M. Jahns, "Source current harmonic analysis of adjustable speed drives under input voltage unbalance and sag conditions," *IEEE Trans. Power Del.*, vol. 21, no. 2, pp. 567–576, Apr. 2006.
[6] S. Varadan, E. B. Makram, and A. A. Girgis, "A new time domain voltage source model for an arc furnace using EMTP," *IEEE Trans. Power Del.*, vol. 11, no. 3, pp. 1685–1691, Jul. 1996.
[7] C. J. Wu and T. H. Fu, "Effective voltage flicker calculation algorithm using indirect demodulation method," *Proc. Inst. Elect. Eng.—Gener. Transm. Distrib.*, vol. 150, no. 4, pp. 493–500, Jul. 2003.
[8] A. Hernandez, J. G. Mayordomo, R. Asensi, and L. F. Beites, "A method based on interharmonics for flicker propagation applied to arc furnaces," *IEEE Trans. Power Del.*, vol. 20, no. 3, pp. 2334–2342, Jul. 2005.
[9] M. O. Faruque, V. Dinavahi, and W. Xu, "Algorithms for the accounting of multiple switching events in digital simulation of power-electronic systems," *IEEE Trans. Power Del.*, vol. 20, no. 1, pp. 1157–1167, Apr. 2005.
[10] IEEE Task Force on Harmonic Modeling and Simulation, "Real-time digital time-varying harmonic modeling and simulation techniques," *IEEE Trans. Power Del.*, vol. 22, no. 2, pp. 1218–1227, Apr. 2007.
[11] J. A. Hollman and J. R. Martí, "Real time network simulation with PC-cluster," *IEEE Trans. Power Syst.*, vol. 18, no. 2, pp. 563–569, May 2003.
[12] M. Park and I. K. Yu, "A novel real-time simulation technique of photovoltaic generation systems using RTDS," *IEEE Trans. Energy Convers.*, vol. 19, no. 1, pp. 164–169, Mar. 2004.
[13] C. Dufour and J. Belanger, "Real-time PC-based simulator of electric systems and drives," in *Proc. Int. Conf. Parallel Comput. Elect. Eng.*, Sep. 2004, vol. 1, pp. 105–113.
[14] V. Dinavahi, R. Iravani, and R. Bonert, "Design of a real-time simulator for a D-STATCOM system," *IEEE Trans. Ind. Electron.*, vol. 51, no. 5, pp. 1001–1008, Oct. 2004.
[15] B. De Kelper, H. F. Blanchette, and L.-A. Dessaint, "Switching time model updating for the real-time simulation of power-electronic circuits and motor drives," *IEEE Trans. Energy Convers.*, vol. 20, no. 1, pp. 181–186, Mar. 2005.
[16] Y. Liu, M. Steurer, and P. Ribeiro, "A novel approach to power quality assessment: Real time hardware-in-the-loop test bed," *IEEE Trans. Power Del.*, vol. 20, no. 2, pp. 1200–1201, Apr. 2005.
[17] L.-F. Pak, M. O. Faruque, X. Nie, and V. Dinavahi, "A versatile cluster-based real-time digital simulator for power engineering research," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 455–465, May 2006.
[18] H. Li, M. Steurer, K. L. Shi, S. Woodruff, and D. Zhang, "Development of a unified design, test, and research platform for wind energy systems based on hardware-in-the-loop real-time simulation," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1141–1151, Jun. 2006.
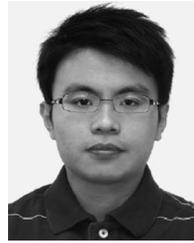
[19] S. Jung and S. S. Kim, "Hardware implementation of a real-time neural network controller with a DSP and an FPGA for nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 265–271, Feb. 2007.

[20] B. Lu, X. Wu, H. Figueroa, and A. Monti, "A low-cost real-time hardware-in-the-loop testing approach of power electronics controls," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 919–930, Apr. 2007.

[21] G. G. Parma and V. Dinavahi, "Real-time digital hardware simulation of power electronics and drives," *IEEE Trans. Power Del.*, vol. 22, no. 2, pp. 1235–1246, Apr. 2007.

[22] X. Tu, L.-A. Dessaint, N. Fallati, and B. De Kelper, "Modeling and real-time simulation of internal faults in synchronous generators with parallel-connected windings," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1400–1409, Jun. 2007.

[23] C. Dufour, J. Belange, and S. Abourida, "Accurate real-time simulation of a 6-pulse inverter with real-time event compensation in ARTEMIS," *Math. Comput. Simul.*, vol. 63, no. 3–5, pp. 161–172, Nov. 2003.

[24] N. Léchevin, C. A. Rabbath, and C. Dufour, "A digital control perspective for real-time modeling of electrical switching systems—A case study of linear circuits with diodes," *IEEE Control Syst. Mag.*, vol. 25, no. 6, pp. 69–85, Dec. 2005.

[25] N. Léchevin, C. A. Rabbath, and P. Baracos, "Distributed real-time simulation of power systems using off-the-shelf software," *IEEE Can. Rev.*, pp. 4–8, Aug. 2001.

[26] *Electromagnetic Compatibility (EMC). Part 4: Testing and Measurement Techniques. Section 15: Flickermeter Functional and Design Specifications*, IEC Std. 61000-4-15, 2003.

Yu-Jen Liu (S'07–M'09), received the Ph.D. degree from National Chung Cheng University, Chiayi, Taiwan, in 2009.

His areas of research interest include power quality, power-system harmonic modeling, and real-time simulation.

Venkata Dinavahi (M'00–SM'08) received the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2000.

He is currently an Associate Professor in the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His areas of research interest include electromagnetic transients, power electronics, and real-time simulation and parallel processing.

Dr. Dinavahi is a Professional Engineer in the Province of Alberta.

Gary W. Chang (M'94–SM'01–F'10) received the Ph.D. degree from the University of Texas at Austin, in 1994.
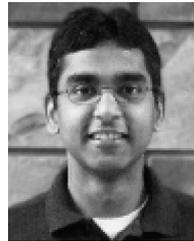
He is currently a Professor in the Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan. His areas of research interest include harmonics and power quality.

Dr. Chang is a Registered Professional Engineer in the State of Minnesota. He chairs the IEEE Task Force on Harmonics Modeling and Simulation.

Huai-Jhe Su received the B.S.E.E. and M.S.E.E. degrees from National Chung Cheng University, Chiayi, Taiwan, in 2007 and in 2009, respectively, where he is currently working toward the Ph.D. degree.

His areas of research interest include power quality and real-time simulation.