CrossMark

ORIGINAL PAPER

Interactive simulation-based-training tools for manufacturing systems operators: an industrial case study

Alberto Vergnano[1] · **Giovanni Berselli**[2] · **Marcello Pellicciari**[1]

Abstract Industrial process plants are increasingly becoming complex structures with high level of automation. Nonetheless, the final plant productivity and the overall equipment efficiency does not solely depend on an optimized engineering design/installation practice, but also on human operators supervision. In parallel, along with the classic demand to minimize costs and time-to-market during the design phases, issues concerning human safety and failure prevention play a crucial role, one of the highest target being the avoidance of dangerous process states. Within this context, Simulation-Based-Training (SBT) allows plant operators to learn how to command complex automated machineries within a secure virtual environment. Similar to its usage in medical, aerospace, naval and military fields, SBT for manufacturing systems can be employed in order to involve the user within a realistic scenario, thus providing an effective, lifelike, interactive training experience under the supervision of experienced personnel. In addition, also according to previous literature, industry-driven SBT may be effectively envisaged as a natural extension of the plant life-cycle simulation practice, comprising Design Simulation & Optimization, Virtual Commissioning, Operator Training, up to Plant Maintenance. In this context, since the overall system behavior depends both on manufacturing process dynamics and Control Logics, the main challenge for an effective SBT is related with the development of a real-time environment where control system responsiveness is fully reproduced. Owing to this consideration, this paper reports a successful industrial case study, concerning a novel SBT workbench used for steel plants operator training, discussing both the virtual prototyping phase and the development of a real-time simulation architecture. In particular, a hybrid process simulation is employed, where a virtual process model is coupled with physical PLC and Human–Machine Interface, thus achieving an accurate reproduction of the real plant/operator interaction.

✉ Marcello Pellicciari
  marcello.pellicciari@unimore.it

  Alberto Vergnano
  alberto.vergnano@unimore.it

  Giovanni Berselli
  giovanni.berselli@unige.it

1 Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, Via Pietro Vivarelli 10, 41125 Modena, Italy

2 Department of Mechanics, Energetics, Management and Transportation, University of Genova, Via All'Opera Pia, 15, 16146 Genova, Italy

1 Introduction

Automatic machinery design must strive to high production rates by optimizing the mutual interaction between mechanics, electronics, control logics and operation sequences. Along with the increasing complexity of automated plants, also the engineering challenges for plant design, installation, maintenance and compliance with safety requirements consequently increase [1,2]. Nonetheless, an important aspect which is usually underestimated during the design process, is that the actual plant performances can be quantified as the rated output only in the best case, since the overall equipment efficiency does not solely result from the

🙂 Springer

reliability/optimization of the machinery itself but also as a consequence of the interaction with the human operators [3,4]. For instance, concerning the particular case of batch processes automation, the German norm DIN8743 [5] introduces the distinction between *theoretical* and *actual* production. On one hand, the *theoretical* production capacity is set in relation to an ideal condition, in which the machine works continuously over the complete time interval without any stoppages while producing only compliant product units, (i.e. no waste). On the other hand, the *actual* output, which is of course smaller than the theoretical production capacity, results from the fact that some product units may not comply with the required features and shall be rejected. Moreover, within the set time interval the machine operation normally undergoes interruptions, due to both functional and accidental (*operator-related*) causes.

As for human related inefficiencies, the interactions spans from supervisory control through Human–Machine Interfaces (HMI) [6,7] to man-machine collaboration [8,9]. This research focuses on the first case, as a typical scenario for heavy and hazardous machineries. In practice, the correct states interpretation enables to fire the proper HMI inputs for the transitions to the desired controlled states. On the contrary, a wrong input sequence results in performance degradations or bad failures [10,11]. Both literature and practitioners report that the operators tasks should be envisaged as critical and variable (in function of the specific work skills, physical and mental workloads and personal motivations [12,13]). Therefore, in order to improve the operator performances and, as consequence, the system ones, a key issue is to provide effective and interactive operator training [14,15].

It should be underlined, at first, that the design of any training material starts from the consideration that operator training is different from teaching. Teaching is focused on knowledge, to be learned from e.g. written instructions, movies and tutorials. Training should focus on work skills through guided experiences (not just information), in order to acquire higher confidence on specific results in a short timeframe. The practical operator training on the job is fundamental, although critical in terms of effort, time, costs, hazard to trainees and to the equipment itself, especially in case of safety procedures. These drawbacks can be bypassed in training sessions on digital materials to some extents [16].

In this context, Simulation-Based-Training (also named Virtual Training) has gradually become a useful technology available in many fields [17,18], as e.g. surgery [19,20], aerospace [21], automotive [22]. The virtual experience involves hazards similar to a videogame and literature reports even performances improvements for training on simulated equipment, compared to the actual one. For instance, the most advanced flight simulators can provide a degree of reliability/realism that are even certified to add credit hours for achieving/maintaining the pilot license. In general, however, each field is characterized by its own requirements, the main research areas being (1) physics and/or control simulators, (2) human-computer interaction and visual systems and (3) training session aids. Reducing the investigation to industrial training on heavy automatic machineries, these drawbacks arise:

- *Resource usage efficiency* Many virtual training applications are too sophisticated and costly to be adapted to each single machinery that can be customized or even engineered to order;
- *Predictive modelling* The reliability of the behaviour of a virtual prototype is a priority with respect to its graphical representation. In fact, the deep interaction between mechanics and control software logics through sensor and actuation systems [23] involves also real time phenomena with cycle times like few ms, that are much faster than the graphics updates perceivable by humans;
- *Integration with engineering processes* The control software, fundamental part of the machinery behaviour, is not practically portable from control to simulation technologies, due to its weight of hundreds or even thousands of inputs/outputs (IOs), along with the software dependency to the specific vendor firmware. Then, to avoid doing the work twice, the reuse of already existing software tools is a demand for introducing SBT as part of the machinery development process.

These considerations provide the foundation for the present work in investigating the engineering issues in conceiving/providing virtual training material, focusing on heavy machinery industry. The paper recalls the concept of industry-driven SBT as an integral part of the various engineering phases to be faced during the plant life-cycle [24,25]. The methods and the commercial software tools employed in this paper are presented, along with the particularities of an industrial case study, where SBT has been successfully employed for training the operators of an Electric Arc Furnace (EAF).

The manuscript is organized as follows: Sect. 2 provides an insight of SBT as a natural extension of the Virtual Commissioning approach, also introducing hybrid virtual/physical simulations; Sect. 3 generally describes the specifications to build a virtual prototype for training purposes. A framework for reusing an existent 3D modelling package although enabling interactivity through Man-in-the-Loop (MIL) and Hardware-in-the-Loop (HIL) is presented in Sects. 4 and 5. Section 6 finally reports the discussions on the user experience, whereas Sect. 7 draws the concluding remarks.
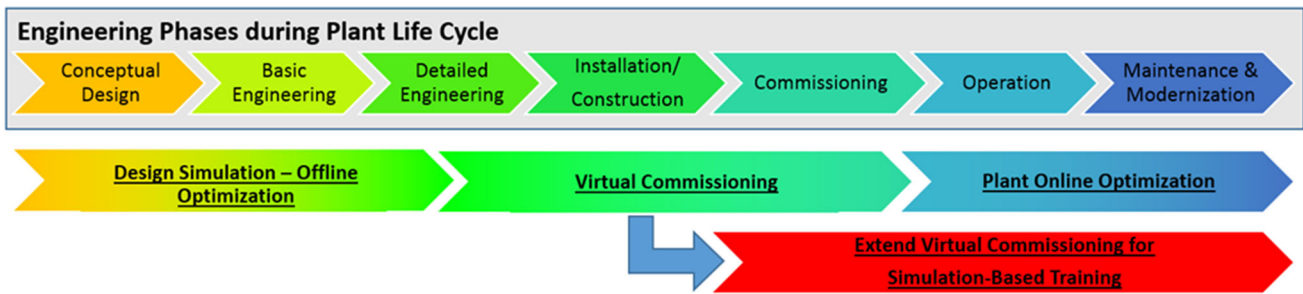
Fig. 1 Simulation processes in the various phase of a plant life-cycle (adapted from [22]). Extension of the virtual commissioning approach to simulation-based-training of industrial automated plants

2 Simulation-based-training as part of the plant design process

In the specific case of industrial plants, as previously proposed by Oppelt et al. [26], SBT may actually be envisaged as a natural extension of the overall plant simulation process. This concept is schematized in Fig. 1, which underlines the use of simulation on the various engineering phases of the plant life-cycle. In particular, Oppelt identifies four use cases, namely:

1. *Design Simulation & Offline Optimization* [27,28], which may be classified into static (steady-state) and dynamic simulation of the production process. The main outcome of the former is a locally optimized process and machinery layout, whereas the latter investigates start-ups, shutdowns and transient plant behaviour, thus providing information about the design of the product unit.

2. *Simulation-Supported Engineering & Virtual Commissioning* [29,30], which builds up on the previous phase, also embedding the control system design. According to the terminology employed in [26], the controller (such as a PLC) can be either physically included, i.e. Hardware-in-the-Loop (HIL) setup, or emulated,

i.e. software-in-the-loop setup. The purpose is to tentatively mimic the actual controlled behavior of the later established physical plant. The four various cases this hybrid virtual/physical design & testing procedure for automated manufacturing system design, as recently discussed in [31], are schematized in Fig. 2. This picture highlights that either (or neither) the Process and the PLC can be included as HIL or emulated mock-ups. The situation in which a virtual prototype of plant and process are coupled to a physical HIL controller is termed Virtual Commissioning (VC), a procedure allowing to validate the controller behavior before actual plant installation.

3. *Simulation supported (Online) Plant Optimization,* the plant models developed in the previous stages can also be used to provide suggestion of improvements (e.g. model-based predictive control) and/or maintenance of already established plants (e.g. virtual machines for condition based maintenance [32]).

4. *Simulation-Based Training*, whose purpose, as already said, is to train the personnel for normal production procedures, emergency/abnormal situations, startup and shutdown behavior and abnormal process conditions. According to this design framework, simulation methods and related software tools employed during the plant design (at least, during the first two above mentioned
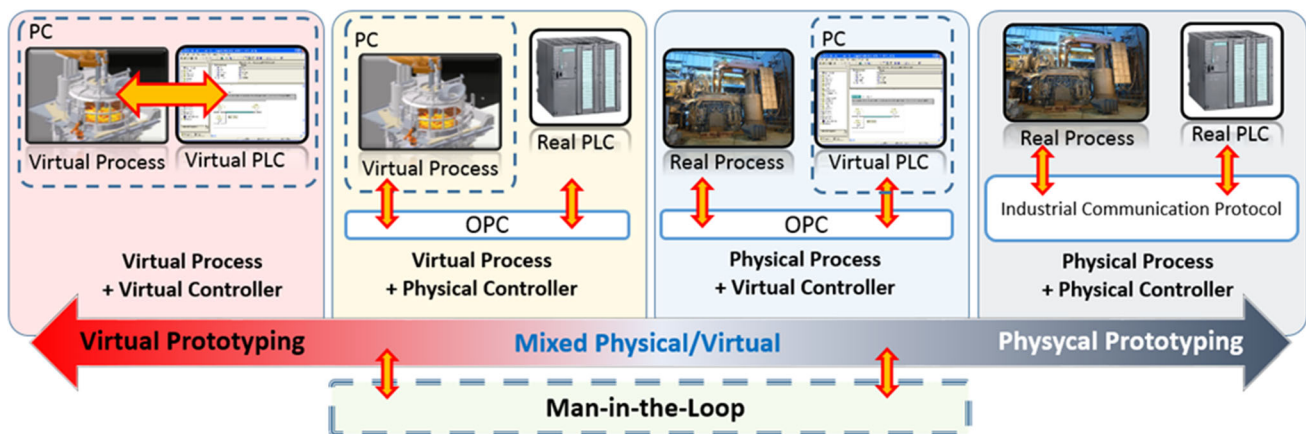

Fig. 2 Hybrid physical/virtual simulation (control Hardware-in-the-loop + Man-in-the-loop)
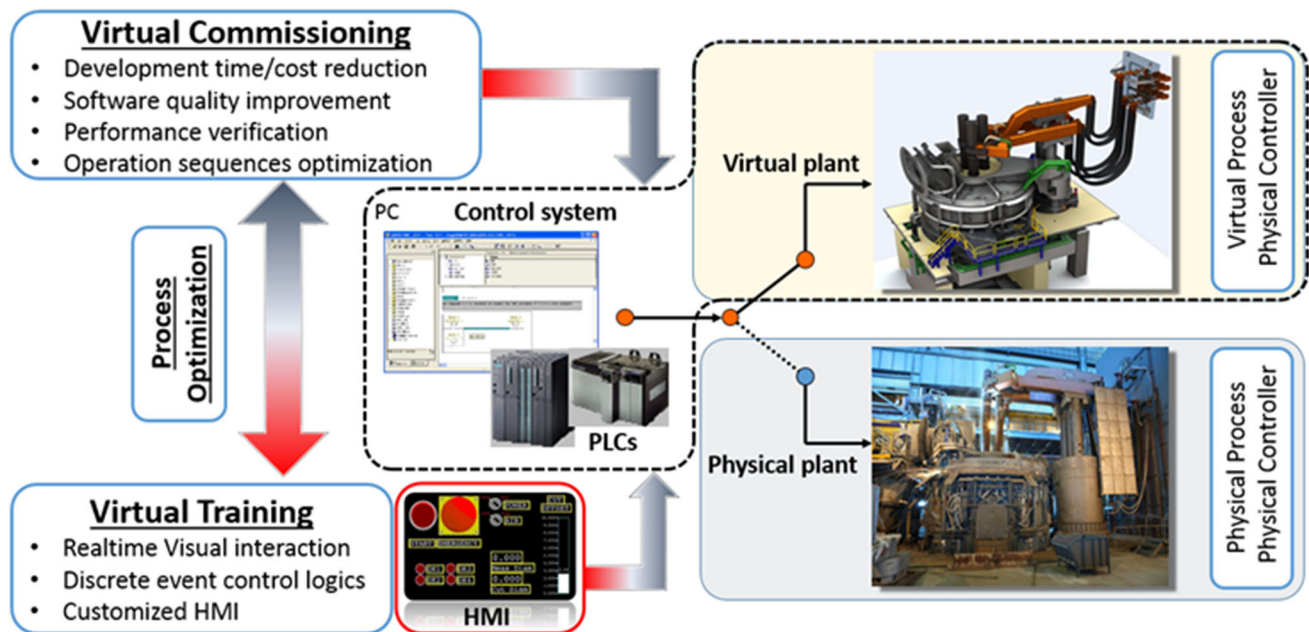
Fig. 3 The concept of virtual commissioning and its extension to simulation-based training

phases) can be effectively re-employed for SBT, which may be envisaged as a further design step following the VC stage.

In particular, focusing on the VC approach and on the subsequent virtual training phase, whose implementation framework and main features are schematized in Fig. 3, several potential benefits can be highlighted. For instance, VC tools are capable to provide an interactive and dynamic 3D visualization, while modelling complex machineries with a system perspective (including mechanics, control logic, operational sequences). In parallel, when connected to a VC model, the controller can be extensively tested in all different working scenarios (such as serious emergencies and extreme performance), although not leading to irreparable damages of the real plant. In practice, also according to prior projects and surveys presented in the literature [33], VC tools allows achieving non-negligible reduction of development time/cost, efficient performance verification prior to physical testing, optimization of operation sequences on a virtual benchmark. Building on these basic considerations, a successful introduction of SBT in an industrial environment requires the virtual training material to be seamlessly embedded into existing engineering processes and design phases, thus reducing further efforts/costs as much possible [34]. A key feature for the VC to be extended for SBT purposes, besides the HIL controller, is the introduction of a Human–Machine Interface (HMI) to include human actions in the simulated world, namely a Man-in-the-Loop (MIL) architecture, see Figs. 2 and 3. On one hand, MIL allows to provide stimuli to the trainee during the training ses-

sions while keeping track of his/her actions, whereas HIL is needed in order to interface simulated IOs to a physical controller/component. Since the physical controller would not be in any case affected by potential damages during SBT, its inclusion with a HIL approach is fundamental for an efficient reuse of the work already done in the previous design phases and for achieving a predictive virtual model that replicates the real plant. Owing to these considerations, also recalling previous definitions from the literature (see e.g. [35]), the proposed simulation framework can be envisaged as: a) *multidisciplinary* and *highly integrated* [36], since knowledge from mechanical, electronic and process engineering are simultaneously leveraged; b) *interactive* [37], since it entails the close collaboration of engineers, expert users (trainers) and unexperienced users (trainees) on a hybrid virtual/physical platform simulating the behaviour of a rather complex machine; c) *mostly based* on a *virtual prototyping* (VP) approach. Specific features of a VC model to be applied in SBT are furtherly discussed in the following section.

3 Requirements and challenges for effective SBT in industrial scenarios

A SBT platform for automated manufacturing system operators is characterized by a number of features, which mimic similar simulators in other fields of technology (e.g. medical, aerospace, naval industries). On the other hand, the effective implementation of SBT in an industrial scenario, must surely deal with some peculiar requirements and challenges, which may be outlined as follows:

- *Cost effectiveness VS Model Size* On one hand, a cost reduction in setting up an SBT platform can be achieved by exploiting existing engineering work, such as CAD models and/or simulations available from the initial plant design stages. In case a physical controller is employed, the size of each virtual prototype (3D virtual representation + behavioral models) and the depth of details are practically determined by the HIL interface. In practice, the models must comply with the concept of *transparency*, which means that the virtual prototype must account for the whole Input-Output (IO) map of the physical plant, so that the controller does not ideally encounter any difference from the actual equipment [38]. Therefore, the model size depends on the considered control level, increasing from process, machine, cell up to plant [33]. The opposite trend is followed for the details, where a process model can even reproduce the instantaneous motions of the servo drives, whereas a plant model mainly considers a discrete event simulation. Therefore, for what concerns the 3D graphic representation, it should include only those phenomena that cause a stimulus perceivable by the operator (unperceivable quantities, like e.g. high-speed motions/vibrations, may add useless computational weight). If 3D models exported in neutral formats from common mechanical CAD packages are re-used for saving costs/time, then these 3D models should be simplified to save computational weight by deleting useless information or parts. The simulation of the manufacturing process may be an exception and, if necessary, additional CAD parts must be set up for their visualization and interaction with the machinery (as a clarification example, the mechanical CAD of an automated machine does not usually include the graphical representation of the product units and their transformation during the process). At last, specific features must be included, which explicitly record the signals detected by the MIL interfaces, which are clearly not inherently available in other engineering simulations.
- *Interactive Experience* Interactivity features are fundamental to virtually reproduce the actual experiences, which are necessary to train the operators skills. Actually, the operators act on the machinery from the HMI through the interposed controller, while keeping inputs and feedbacks from the same HMI or from the plant views through cameras and control pulpit windows. As for the state interpretation, a realistic representation on a 2D monitor reproduces the operating environment. Although a conventional screen is deemed sufficient for training purposes, that does not mean that a kind of interactive movie would be enough. In fact, the computation of the system behaviors and the effects of operation actions needs complicate physics models with parameters taken from IOs in a shared memory of the controller.

Real Time Computing Two features are deemed as fundamental when connecting the models with HIL and MIL:

- *Synchronization, real-time flow, and continuous cycle execution* As for synchronization, all the models in a simulation must be synchronized for the results to be predictive of the real behaviors. Due to the heavy computational weight, the engineering simulations generally run in *virtual/slowed-down* time, meaning that all the models must wait for the slowest one to be synchronized. Therefore, if the models are connected with a HIL, also the external hardware clock must be slowed down. This is possible only in case of advanced controllers, and not for the most common PLCs, as far as the authors experienced. Finally, a MIL can be somehow delayed by agreement, but the trainee brain does not, so that the training results would be misrepresented. Therefore, the models execution synchronization is mandatory but not sufficient. In fact, the simulation must run in a time flow close to reality, in order to be effectively interfaced with Hardware and Man-in-the Loop. According to standard nomenclature, *real-time* refers to computing systems subjected to time constraints, such as diagnostics deadlines. It can be generally declined in hard, firm or soft real-time, depending on the consequences involved in a missed deadline. In particular, soft real-time basically means that the system performances just degrade when overtaking the deadlines (without catastrophic failures). In the present case of operator training, the system performances are measured as the human perception of fluent or jerky phenomena. The historical speeds used in film making, about 42–62 ms of cycle time [39], are quite far from the control real time specifications, so this feature can be heuristically relaxed to soft real time. At last, SBT requires a continuous cycle execution. In fact, a traditional simulation model is run from initial conditions with a predefined set of computation parameters, like a finite test time, and the results are available for analysis only at the end. On the contrary, HIL and MIL need a virtual prototype interacting with external agents with unpredictable behavior rather than just a numerical computation, namely the SBT requires a simulation without predefined temporal limits or, at most, dependent on operator actions. The virtual prototype is then run through start, all nominal, auxiliary and degraded behaviors, transitions between operating scenarios, stop.

4 Modelling framework for HIL/MIL systems simulation

A possible framework for implementing an SBT platform for automatic system is depicted in Fig. 4, which highlights
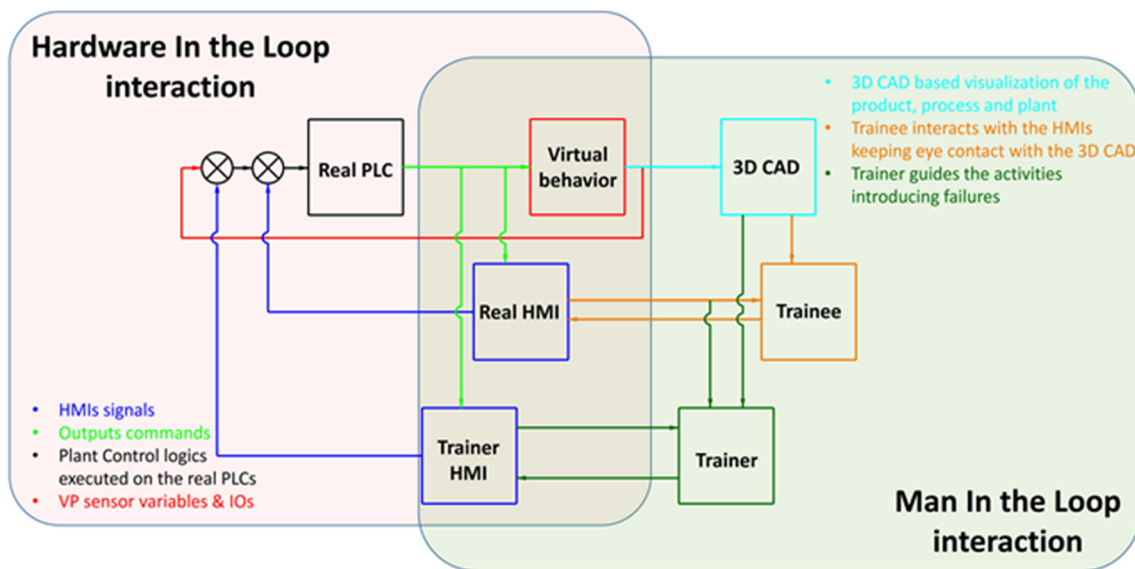
Fig. 4 Schematic of the modeling framework for the HIL and MIL system simulation

both virtual and physical components enabling HIL and MIL interactions.The physical controller (real PLC) executes the machine software as if the actual plant was running. The PLC is slightly reconfigured for inclusion of additional features (such as, inputs from the trainer). PLC outputs are fed to the plant virtual prototype that comprises both plant behavioral models (virtual behavior box) and 3D graphical representation (3D CAD). The machinery VP processes its variables according to its internal state and the controller outputs read from the virtual image updated through the HIL interface. The model then commands the 3D CAD for visualization. Since common 3D CAD can natively implement certain models of some physical features (e.g. collision detection and rigid-body dynamics), the CAD environment may take part to the VP behavior or, alternatively, stands just for an animated graphical interface. The choice of either of these solutions is actually a topic of discussion [40]. Nonetheless, in case the CAD is used also for physical simulation, its update computation in hard real time would fail the real time control deadlines. Therefore, it is easier to adopt the second solution, that is having the 3D CAD representation separated from the machinery physics model and updated in soft real time with quite longer and independent cycle times. In parallel to the CAD update, the trainee keeps track of both HMI information and the 3D CAD representation. Consequently, the operator can decide upon suitable strategies and provide inputs to the controller through the physical HMI. These inputs are complementary to the behavior results in the IOs map just as in the actual machinery. Furtherly, a second HIL interaction is embedded, since a skilled operator (the trainer) can guide the trainee activities with additional information from another HMI. This second HMI can be either physical (although it

would add one un-necessary component) or even set up via a virtual panel in the same VC software. For training purposes, the second HMI should be simply hidden to the operator (e.g. using a second screen). The trainer actions can also bypass the virtual behavior results to force transitions between machinery states (for instance, simulate a catastrophic event).

5 Case study—SBT for electric arc furnace operators

As an industrial case study, a virtual prototype of an Electric Arc Furnace (EAF) has been employed to empirically test SBT features and efficacy. As known, an EAF is a powerful machinery that heats and melts charged material (e.g. steel scraps) via an electric arc. The main nominal data concerning the EAF considered in this paper are reported in Table 1. Obviously, this kind of machinery involves non trivial hazards and high fixed and variable costs. At the state of the art, EAF operator training requires long times/efforts, and cannot fully reproduce many incipient critical scenarios for obvious

Table 1 Electric arc furnace—some nominal data

Weights	∼1.000 tons
Production	100 tons/h
Operation cycle duration	1 h
Molten steel temperature	>1600°C
Power	90 MW
Alternating current	60000 A
Electrodes size (3 × graphite electrodes)	$\phi\,400 \times L6500$ mm

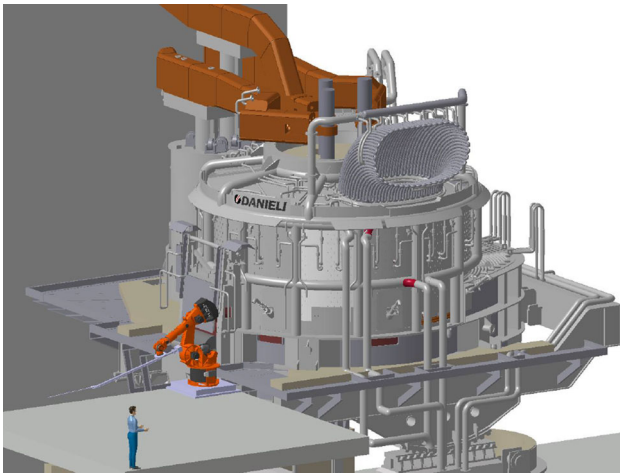**Fig. 5** Operator control pulpit (courtesy of Danieli Automation)

Fig. 6 EAF Virtual Prototype (courtesy of Danieli Automation)

reason (i.e. real accident may result in huge money losses). The EAF is controlled by two SIMATIC S7-400 PLCs [41]. The first PLC supervises the process variables, such as the current waves, and adjusts the working parameters accordingly. The second PLC manages the machinery functions controlled through the HMIs, like scraps charging, slagging, liquid steel tapping, robot temperature, and chemical sampling. The operator works through the HMIs from a control pulpit, shown in Fig. 5, whereas the graphical 3D plant representation for SBT is depicted in Fig. 6.

5.1 Tools for building the virtual model

The overall SBT platform has been developed using the VC tools from Dassault Systemes, DELMIA^{TM} and DELMIA Automation^{TM}. As widely commented in the literature [31, 42–44], these set of PLM tools can be used, among several other tasks, to design, model and simulate automated manufacturing systems. Initially, a realistic virtual representation is achieved by importing neutral *.step files generated from Inventor CAD 3D software. As previously highlighted, the CAD assemblies must be optimized in order to reduce the

computational weight of the simulations. Usually the standard parts, such as screws and bearings, are the less useful for the virtual experience and can be deleted. Then, starting from a simplified 3D CAD, the nominal behavior of the essential system modules is set up, emulating the necessary IOs to/from the controller. A schematic of the Delmia Simulation environment is reported in Fig. 7. The overall system is conceptually divided into four types of simulation modules (also named "smart devices"): (1) mechanical devices, (2) actuators, (3) electrical devices, (4) processes. Mechanical devices are linkages, dynamic behaviors if necessary, logic interactions and collisions. The actuators are electric motors, hydraulic cylinders and a robot. Electrical devices are the EAF electrodes. The considered processes are steel scraps handling, melting, liquid roll and flows. In theory, a fifth conceptual module could be included, embedding the systems sensors. Nonetheless, the sensors are not mentioned as simulation modules since they are simply modelled as variables from other modules, that are then defined as global variables (IO ports). The modules behaviors are modeled with IEC 61131-3 programming languages [24,31]. Finite state behaviors use Sequential Function Charts (SFC) for sequential and parallel operations. Dynamic, continuous and logic behaviors are set up with Function Block Diagram (FBD). The models use internal variables to keep their state and external ports to communicate to other entities. The ports are just global variables, but named differently as M_*, E_* and L_* to describe, respectively, mechanical, electrical or logic parameters. Special functions link the model variables to the CAD models and to the simulation timeline. When the smart devices have been set-up, the IOs can be connected to a virtual controller (also including a virtual HMI) for debugging and then to the physical PLC/HMI. Finally, the graphic representation is completed with additional CAD models to visualize the process transition states and any other graphic effects that can act as stimuli for the operator. A schematic of the first HIL simulation prototype is depicted in Fig. 8.

5.2 Hardware in the loop interfaces

As previously said, in order to command the 3D virtual prototype by means of the plant software logics, a virtual "controller emulator" may be employed or, otherwise, the plant software can be run onto the physical PLCs and HMIs. The case study presented in this paper follows the latter approach, in order to: (1) provide a realistic experience while reusing the real HMIs; (2) spare the effort in porting\adapting the source code within the controller emulator. Therefore, the physical PLCs are interfaced with the simulated machinery through OPC-Scout, an OLE for Process Control server [31]. The software is completed with specific data blocks to read/write signals from the VP, whereas the operator HMIs
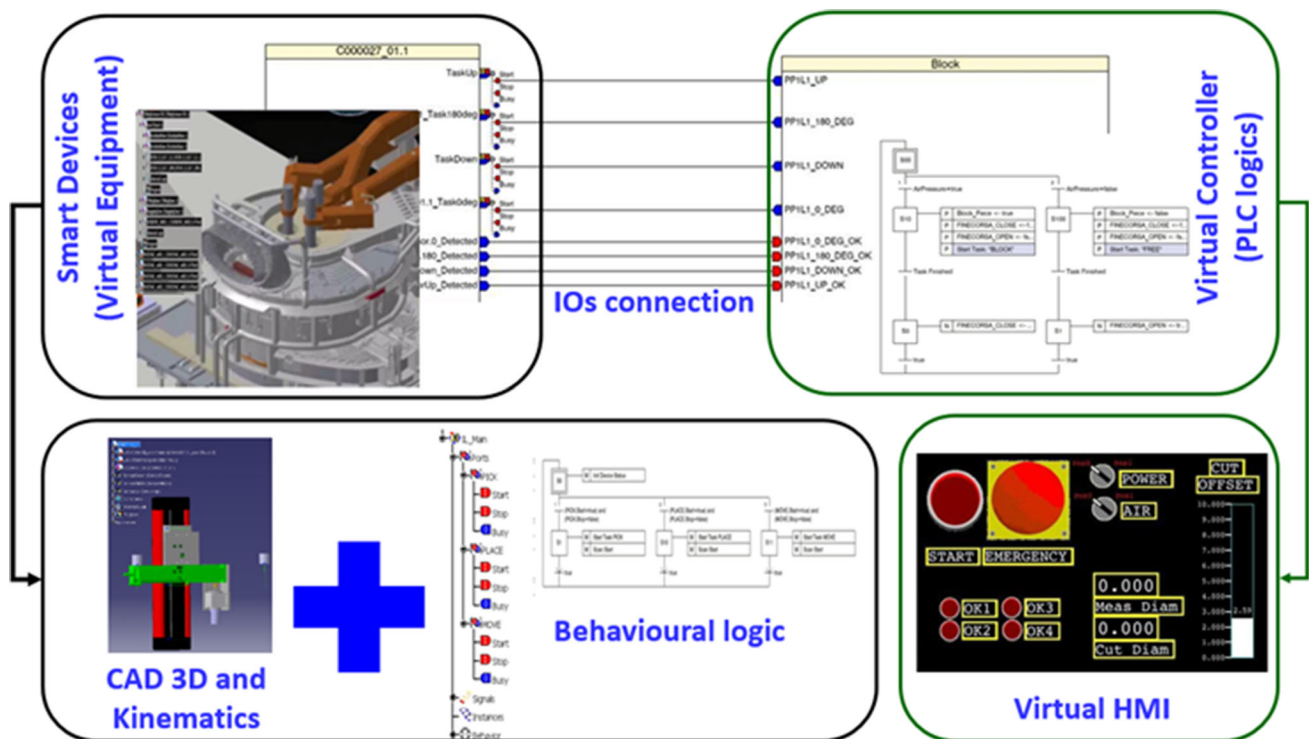
Fig. 7 Schematic showing the connection of smart devices and virtual controller (comprising HMI) in Delmia Automation
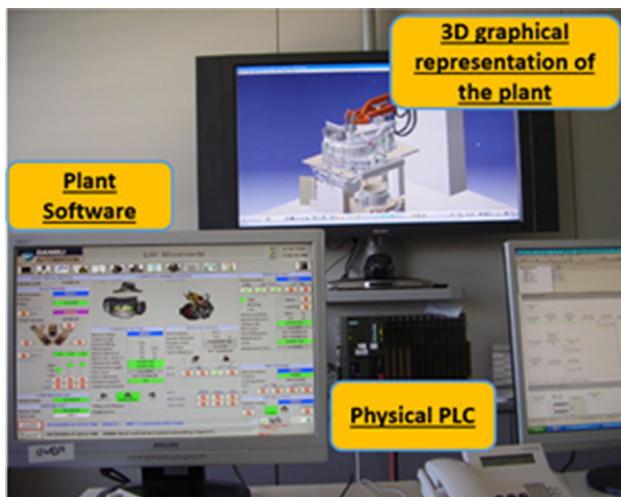

Fig. 8 Laboratory Prototype including graphical representation, virtual HMI and physical PLC

are directly connected to the PLCs, as in the real system. In practice, the proposed SBT architecture comprises:

1. PLC units (2 CPU Siemens S7-400 run in parallel). The PLCs operate in hard real time (2 ms), with no possibility to be forced to operations employing a virtual clock;
2. The 3D virtual prototype, which operates in "virtual" time, due to the computational weight for the synchro-

nization of behavioral models and CAD representation. This "virtual" time is variable (e.g. in the range of 20–500 ms), since different computations are involved in different simulation states. For example, the computations required are different if (at a certain time instance) one single actuator is moving or several motions must be contemporarily computed;
3. The OPC communication, which does not operate in real time, being slow, variable (about 100–200 ms), and non-homogeneous (i.e. there is no certainty to which PLC time instance the OPC communication is actually dealing with).

The simpler solution is to synchronize virtual process, PLC and OPC is to run a clock into the PLC, as a watchdog, and then to parameterize the virtual process integrators with the sampled PLC time intervals, instead of the simulation time. In practice, the process model advances of one single step any time it receives an input from the PLC. This solution is easily implemented but the outcome is a slow update of the 3D model (at about 8–10 Hz), which is perceived as jerky motions.

The final solution is the algorithm reported in Fig. 9, which monitors both the PLC and VP simulation times, the variables definitions being reported as comment lines in the same figure. The script synchronizes the virtual prototype with the PLC controller in case the virtual prototype computa-

```
Macro Main synchronizer

Input:   PLC.time  % PLC clock;
         SIM.time  % Synchronized simulation time flow;
         HiL Inputs % Outputs from PLC and HMI;
Output:  SIM.time.step % Time for the numerical integration steps;
         HiL Outputs % Simulation outputs;

1    Init Variables

2    Model run // also run the PLC

3    loop

4        if PLC.time > pre.PLC.time

5            PLC.time.step = PLC.time − pre.PLC.time

6            next.SIM.time = PLC.time + PLC.time.step

7            SIM.time.step = ( next.SIM.time − SIM.time ) / SIM.steps

8            pre.PLC.time = PLC.time

9            SIM.steps = 1

10       else

11           SIM.steps = SIM.steps + 1

12       if SIM.time >= next.SIM.time

13           SIM.time.step = 0

14       SIM.Outputs = f ( Inputs , State , SIM.time.step )
```

Fig. 9 The macro that synchronizes simulation and PLC

tional weight varies during time, the simulation being either sometimes slower and sometimes faster than the PLC communication. On one hand, even if the synchronization with the OPC is guaranteed by the macro, the virtual process will comprise motions that will be perceived as "jerky" whenever the simulation is excessively heavy. On the other hand, if the virtual process computational weight is low (i.e. computations can be handled in that machine state with a time step lower than the OPC communication one), the simulation will simply proceed of more steps as compared to the PLC, and the machine operator will perceive a fluid 3D movie without latches. In practice, an efficient (in terms of computational weight) simulation is a demand for these kind of applications. With reference to Fig. 9, the macro mainly comprises three routines:

- A first *"if"* routine, which relates to a case where the OPC sends a new communication (including several PLC steps) and a new PLC time step is computed (i.e. the variable *PLC.time.step*). Then, the main macro advances to the next (future) time instant (*next.SIM.time*), computed as the actual *PLC.time* increased by the *PLC.time.step*. In practice, the algorithm computes what should be the simulation time step during the next communication instant, in order to be synchronized with the PLC. The simulation time step (*SIM.time.step*) for the simulation integrators is computed as the time step the simulation

should cover in the next "blind communication time" divided by the number of steps the simulation ran during the yet concluded "blind communication time". Finally, the variable *pre.PLC.time* is re-set to *PLC.time* and the variable *SIM.steps* is re-set to 1.

- An *"else"* routine, which relates to a case where the OPC did not send a new communication during the previous cycle run, meaning that the simulation is faster than the OPC. The number of *SIM.steps* for each OPC communication is simply increased by 1. In this case, *SIM.steps* will be used for the computation of the next *SIM.time.step*.

- A second *"if"* routine, which relates to a case where the simulation has not yet reached its *next.SIM.time*, as computed in the previous cycle, meaning that it made a transition to a computationally lighter step. The *SIM.time.step* used in the integrators is then set to 0, so that the simulation time is stopped. The simulation still runs in order to update the previous *SIM.steps*. However the events, e.g. motions, are all stopped since the integration time is null.

Finally, before the next cycle, the simulation outputs are computed, as function of the simulation Inputs (namely, the PLC outputs), State (state of the machine in the simulation), and *SIM.time.step*. Note that all the simulation routines should be parameterized in function of the *SIM.time.step*, such as integrators or delays for any event which is function of time. The final SBT platform installation is depicted in Fig. 10, which shows an EAF operator managing the virtual prototype from a physical control pulpit.

6 Additional features to effectively enable human interaction and SBT in industry

6.1 Interactions for trainees and trainers

As compared to a model employed for VC only, the SBT platform comprises additional functions to enhance interactivity with both trainees and trainer. For what concerns the trainees, the following functions have been implemented:

1. The virtual model capabilities are extended with "more than real" functions, namely color/transparency changes in the 3D graphical representation. In particular, part color changes in case of parts collisions or in case the joint limits of some mechanical devices are reached. Similarly, temperature and/or pressure variations in air, oil pipes, vessels are visually highlighted by further color changes. In addition, some parts can be made translucent during the EAF operation phases, like the furnace walls
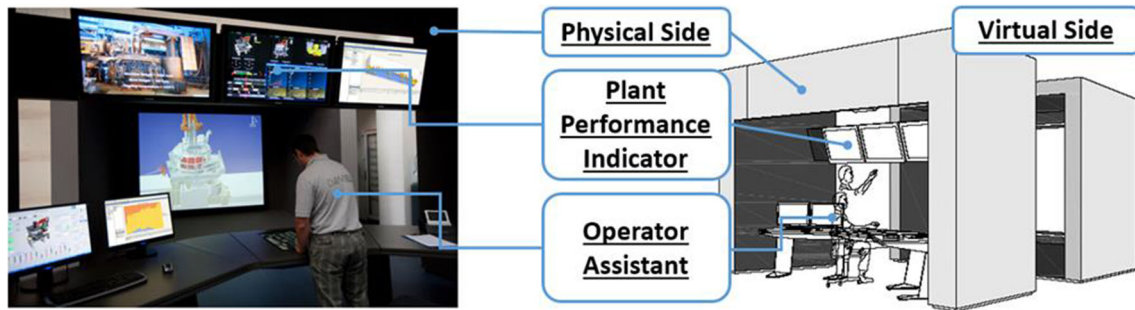
Fig. 10 Operator experiencing the EAF virtual prototype from the control pulpit, (courtesy of Danieli Automation)
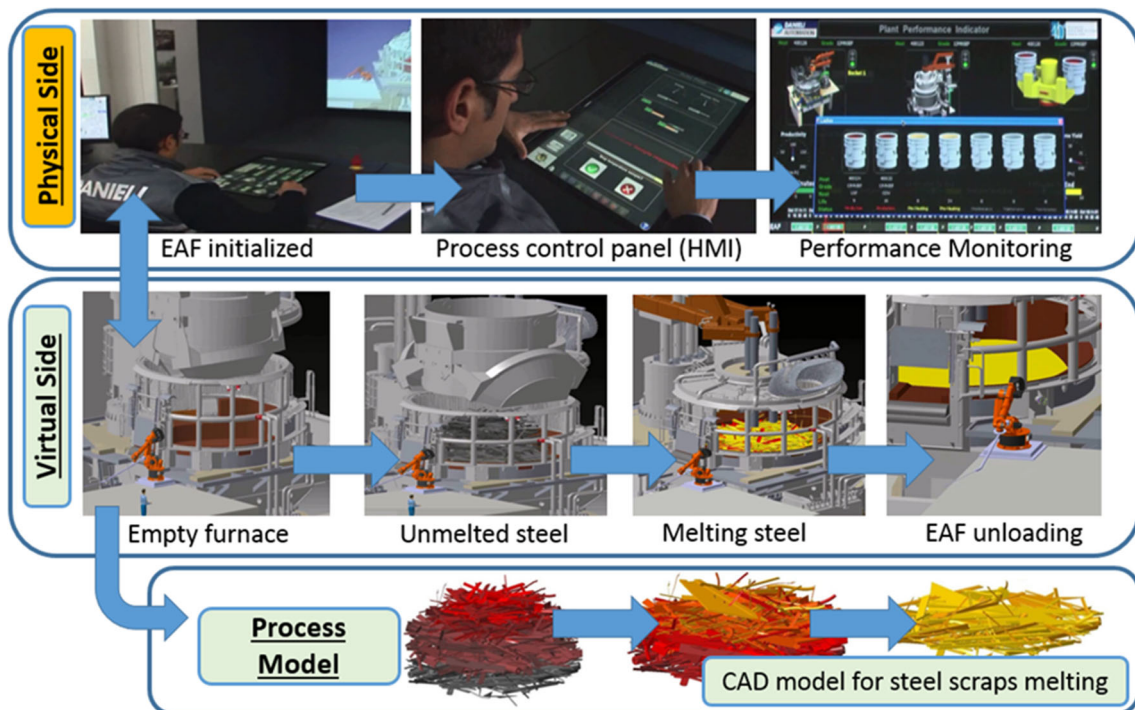

Fig. 11 Diffeent phases of the SBT experience (courtesy of Danieli Automation)

that can be either shown (as in the real system) or hidden (to highlight the process advancements—see Fig. 11).

2. Inclusion of the process visualization, so that the materials to be melted are visualized while changing in shape, dimensions and colors, as shown in Fig. 11. For instance, the steel scraps melt depending on the electrodes positions and powers (as controlled by the operator).

Similarly, the tapping flow changes in function of molten steel quantity and furnace inclination.

For what concerns the trainers, the following functions have been implemented:

1. The SBT platform includes an additional virtual HMI for the trainer that can be used to manage IOs and transitions through different machinery states. As the physical HMIs let the trainees to interact with the machinery through the PLCs IOs, the virtual HMI lets the trainer to interact with the virtual environment by changing at will the operating scenarios and operator stimuli.

2. Possibility to enforce view changes and variation within the CAD models. In particular, different views/orientation of the graphical model can be automatically enforced from the trainer HMI to rapidly present different locations. This is especially important for this machinery dimensions. In addition, color/transparency changes (see previous features) may be enforced through the virtual HMI on the trainer side.

6.2 Operator training

The operator training is guided in the workstation of Fig. 10 (physical side), including the presented 3D CAD representation, HIL, PLCs and HMIs and additional equipment to reproduce the actual pulpit. Several HMIs variants have been introduced and tested, with different command sequences and nested menus. For what concerns simulated training-on-the-job and knowledge delivery, best practices and patterns have been firstly recorded from actions delivered by experienced operators. Then, the recorded material has been used to train new operators, while monitoring the responsiveness of the user. The interactive simulation uses all but only the PLCs IOs, without any bias behavior from past work habits. Therefore, the model reliably responds to correct commands as well as inputs that may lead to possible failures. In practice, the SBT platform can even provide experiences that go beyond conventional training. That is the case of critical scenarios, which are not reproducible in reality, but can be easily simulated without any real damages to humans or equipment. For instance, plant failures have been modeled with a bi-stable behavior, namely a catastrophic emergency is actually reached or it depicts an incipient but still remediable dangerous situation. In addition, even if self-evident, it should be highlighted that the knowledge is delivered through training sessions in scenarios that can be quickly reached without having to wait for the whole manufacturing sequence to start from the beginning. This means that the difficult operations can be iterated several times until its the goals are reached. Also, the training may follow specific sequences, different from the actual machinery ones, that are simulated just depending on the training purposes. At present, the evaluation of the virtual experiences are being collected from interviews to trainers, trainees and experienced operators, with the aim of providing feedbacks on the features for operator training.

6.3 Training material costs

As highlighted in the introduction section, a crucial point for an actual SBT implementation in industry is cost/time effectiveness, meaning that the re-use of existing engineering tools and related data is mandatory. For what concerns cost effectiveness, VC software (such as Delmia Automation) can be interfaced with other commercial tools for control software development and for 3D CAD modeling. The first is achieved by including the real controller through HIL interfaces, whereas the latter is achieved by importing a neutral CAD format from mechanical design. This approach bypasses the vendors limitations for protecting their intellectual property and results in a general purpose tool. Even more important, the reuse of other engineering tools takes advantage of the industrial knowledge gathered in tens of years

at least. For what concerns time effectiveness and resource savings, a VC model is used to test a complex system prior to actual equipment delivery, meaning a safer forecasting of heavy investment cashflows. At present, the system developed for the presented case study is not portable, even if its portability could be enabled with few modifications. That been said, it is not fundamental for the training material and for the future actual machinery to be located in the same place, meaning that operators coming from different locations can be trained in a single advanced training center.

7 Summary and discussion

In the present paper, specifications, requirements and challenges for adapting existing simulation tools for the industrial application of operators virtual training have been presented. It has been highlighted that fundamental features are user interaction (for both trainees and trainers), real-time simulation interfaced with physical hardware (controller and human–machine interface), adequate level of detail (limited to the operator skills to be trained) and, last but not least, cost effectiveness. In fact, owing to the time/costs restriction of any industrial application, re-usage of existing hardware, software, and expertise becomes an actual requirements for SBT to be used in practice.

In light of these considerations, this paper envisages SBT as an integral part of the plant design process, here including the plant Virtual Commissioning, where a CAD model of the system is interfaced with a physical controller prior to the actual plant establishment. This approach allows to reuse existing engineering tools and to set up hybrid process simulations with control HIL in a system perspective. For validation purposes, a large and complex case study on an industrial Electric Arc Furnace has been evaluated. According to the industrial partner, the resources required to set up the training material have been sustainable. In addition, SBT tools enabled new training strategies for different operators and employees, well beyond conventional training mainly based on information sharing via written instructions. Other advantages are lower costs, easier logistics and safety for the courses.

By means of a Virtual Commissioning tool (such as Delmia Automation), the operator training can be carried concurrently with the system final development stages, up to the actual commissioning, thus shortening the plant rump up time. Enablers of the Hardware and Human in the Loop simulations are the interfaces with mechanical and control engineering software tools. Re-usage of existing knowledge from the mechanical designers is simply obtained with common neutral CAD file formats, which allows data transfer from any different CAD vendor. Re-usage of existing knowledge from the control architecture is more critical. The OPC

communication is the ready solution but usable only for slow processes, as far as realistic time flow and synchronization are satisfied.

Envisaged industrial benefits of extending a Virtual Commissioning approach to SBT are: (1) operator training focused on developing skills through guided experiences and not just on acquiring information; (2) training-on-the-job, also in case of safety procedure which include hazards hardly replicable on the physical plant; (3) potential optimization of those process parameters which are heavily dependent on the operator skills; (4) acquisition/recording of best operator practice for recognition of weak points and possible directions of improvement, along with strategies for further plant design optimization.

References

1. Garza-Reyes, A.J., Eldridge, S., Barber, K.D., Soriano-Meier, H.: Overall equipment effectiveness (OEE) and process capability (PC) measures. Int. J. Qual. Reliab. Manag. 27(1), 48–62 (2010)
2. Chand, G., Shirvani, B.: Implementation of TPM in cellular manufacture. J. Mater. Process. Technol. 103(1), 149–154 (2000)
3. O'Hara, J.M., Hall, R.E.: Advanced control rooms and crew performance issues: implications for human reliability. IEEE Trans. Nucl. Sci. 39(4), 919–923 (1992)
4. Parasuraman, R., Sheridan, T.B., Wickens, C.D.: A model for types and levels of human interaction with automation. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. 30(3) (2000)
5. VDMA–Packaging Machinery Association: DIN 8743: Packaging machines and packaging installations, time related definitions, reference factors and calculation fundamentals, available online (2005)
6. Lim, T.G., Reeu, C.W.: Development of human–machine interface for automatic control of electro-slag remelting process. In: Proceedings of 37th IAS Annual Meeting Industry Applications Conference
7. Chao, G.: Human–machine interface: design principles of visual information in human-machine interface design. In: Proceedings of International Conference on Intelligent Human–Machine Systems and Cybernetics, vol. 2, pp. 262–265, Hangzhou, Zhejiang, China (2009)
8. Bradshaw, J.M., Dignum, V., Jonker, C., Sierhuis, M.: Human–agent–robot teamwork. IEEE Intell. Syst. 27(2), 8–13 (2012)
9. Stork, S., Stößel, C., Schubö, A.: Optimizing human–machine interaction in manual assembly. In: Proceedings of The 17th IEEE International Symposium on Robot and Human Interactive Communication, pp. 113–118, Munich, Germany (2008)
10. Bolton, M.L., Bass, E.J., Siminiceanu, R.I.: Using formal verification to evaluate human-automation interaction: a review. IEEE Trans. Syst. Man Cybern. Syst. 43(3), 488–503 (2013)
11. O'Hara, J.M., Higgins, J.C., Brown, W.S., Fink, R.: Human factors considerations with respect to emerging technology in nuclear power plants, U.S. Nuclear Regulatory Commission, Washington, DC, BNL Tech. Report 79947–2008, 20555–0001 (2008)
12. Nachreiner, F., Nickel, P., Meyer, I.: Human factors in process control systems: the design of human-machine interfaces. Saf. Sci. 44(1), 5–26 (2006)
13. Flach, J.M.: The human as a critical component in an adaptive meaning processing system. Proc. IEEE Int. Conf. Syst. Man Cybern. 3, 2586–2591 (2004)
14. Elzer, P., Weisang, C., Zinser, K.: Knowledge-based system support for operator tasks in S&C environments. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, vol. 3, pp. 1078–1083, Cambridge, MA, USA (1989)
15. Dauphinee, J.L.: Development and implementation of a sustainable operator training and certification process. In: Proceedings of IEEE International Semiconductor Manufacturing Symposium, pp. 509–512, San Jose, CA, USA (2001)
16. Stone, R.: Virtual reality for interactive training: an industrial practitioner's viewpoint. Int. J. Hum. Comput. Stud. 55, 699–711 (2001)
17. Salas, E., Rosen, M.A., Held, J.D., Weissmuller, J.J.: Performance measurement in simulation-based training: a review and best practices. Simul. Gaming 40, 328–376 (2009)
18. Bell, B.S., Kanar, A.M., Kozlowski, S.W.J.: Current issues and future directions in simulation-based training in North America. Int. J. Hum. Resour. Manag. 19(8), 1416–1434 (2008)
19. Lian-Yi, C., Fujimoto, H., Miwa, K., Abe, T., Sumi, A., Ito, Y.: A dental training system using virtual reality. Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom. 1, 430–434 (2003)
20. Violante, M.G., Vezzetti, E.: Design and implementation of 3D Web-based interactive medical devices for educational purposes. Int. J. Interact. Design Manuf. 1–14 (2015). doi:10.1007/s12008-015-0277-0
21. Loftin, R.B.: Virtual environments for aerospace training. In: Proceedings of IEEE Technical Applications Conference and Workshops Northcon, Portland, OR, USA (1995)
22. De Filippo, F., Stork, A., Schmedt, H., Bruno, F.: A modular architecture for a driving simulator based on the FDMU approach. Int. J. Interact. Design Manuf. 8, 139–150 (2014)
23. Isermann, R.: Mechatronic systems–a challenge for control engineering. In: Proceedings of the American Control Conference, vol. 5, pp. 2617–2632, Albuquerque, New Mexico, USA
24. Oppelt, M., Wolf, G., Urbas, L.: Towards an integrated use of simulation within the life-cycle of a process plant–a prototypical implementation. In: IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), pp. 1–8 (2015)
25. Oppelt, M., Barth, M., Graube, M., Urbas, L.: Enabling the integrated use of simulation within the life cycle of a process plant: an initial roadmap: results of an in-depth online study. Proceeding of IEEE International Conference on Industrial Informatics, INDIN, art. 7281709, 49–55 (2015)
26. Oppelt, M., Urbas, L.: Integrated virtual commissioning an essential activity in the automation engineering process: from virtual commissioning to simulation supported engineering. In: Proceedings of the 40th Annual Conference of the IEEE Industrial Electronics Society, art. no. 7048867, pp. 2564–2570 (2014)
27. Bausa, J., Dünnebier, G.: Life Cycle Modelling in the chemical industries: is there any reuse of models in automation and control?. In: Proceedings of the 16th European Symposium on Computer Aided Process Engineering and 9th Int. Symposium on Process Systems Engineering, pp. 3–8 (2006)
28. Engl, G., Kröner, A., Pottmann, M.: Practical aspects of dynamic simulation in plant engineering. In: Proceedings of the 20th European Symposium on Computer Aided Process Engineering, ESCAPE20 (2010)
29. Gu, F., Harrison, W.S., Tilbury, D.M., Yuan, C.: Hardware-in-the-loop for manufacturing automation control: current state and identified needs. In: Proceedings of the 3rd Annual IEEE Conference on Automation Science and Engineering, pp. 1105–1110 (2007)
30. Hoffmann, P., Schumann, R., Maksoud, T.M.A., Premier, G.C.: Virtual commissioning of manufacturing systems a review and new approaches for simplification. In: Proceedings 24th European Conference on Modelling and Simulation (2010)
31. Hincapié, M., de Jesús Ramírez, M., Valenzuela, A., Valdez, J.A.: Mixing real and virtual components in automated manufacturing

systems using PLM tools. Int. J. Interact. Design Manuf. **8**(3), 209–230 (2014)

32. Bohlmann, S., Becker, M., Balci, S., Szczerbicka, H.: Online simulation based decision support system for resource failure management in multi-site production environments. In: Proceedings of the 18th IEEE International Conference on Emerging Technologies and Factory Automation (2013)

33. Reinhart, G., Wünsch, G.: Economic application of virtual commissioning to mechatronic production systems. Prod. Eng. **1**, 371–379 (2007)

34. Drath, R., Weber, P., Mauser, N.: An evolutionary approach for the industrial introduction of virtual commissioning. In: Proceedings of IEEE International Conference on Emerging Technologies and Factory Automation, pp. 5–8, Hamburg, Germany (2008)

35. Briand, R., Fischer, X., Arrijuria, O., Terrasson, G.: Multidisciplinary design process based on virtual prototyping for microsystem design. Virtual Phys. Prototyp. **5**(3), 153–162 (2010)

36. Legardeur, J., Merlo, C., Fischer, X.: An integrated information system for product design assistance based on artificial intelligence and collaborative tools. Int. J. Product Lifecycle Manag. **1**(3), 211–229 (2006)

37. Fischer, X., Nadeau, J.: 2006 Research in Interactive Design—Virtual, Interactive and Integrated Product Design and Manufacturing for Industrial Innovation, vol. 3, Springer, France (2011)

38. MacDiarmid, M., Bacic, M.: Quantifying the accuracy of hardware-in-the-loop simulations. In: Proceedings of American Control Conference, pp. 5147–5152, New York, NY, USA (2007)

39. Read, P., Meyer, M.P.: Restoration of motion picture film, In: Butterworth-Heinemann Series in Conservation and Museology, pp. 24–26 (2000)

40. Harrison, W.S., Tilbury, D.M., Yuan, C.: From hardware-in-the-loop to hybrid process simulation: an ontology for the implementation phase of a manufacturing system. IEEE Trans. Autom. Sci. Eng. **9**(1), 96–109 (2012)

41. http://w3.siemens.com/mcms/programmable-logic-controller/en/advanced-controller/s7-400/pages/default.aspx. Accessed 28 Oct 2016

42. Ahmad, M., Ahmad, B., Harrison, R., Alkan, B., Vera, D., Meredith, J.O., Bindel, A.: A framework for automatically realizing assembly sequence changes in a virtual manufacturing environment. Procedia CIRP **50**, 129–134 (2016)

43. Lee, C.G., Park, S.C.: Survey on the virtual commissioning of manufacturing systems. J. Comput. Design Eng. **1**(3), 213–222 (2014)

44. Delmia 3D digital manufacturing solution, Dassault Systems. http://www.3ds.com/products-services/delmia/