



HAL
open science

Rapid Model Predictive Control Prototyping with LabVIEW/CDSim and CompactRIO Target

Mohamed Derouiche, Soufiene Bouallègue, Joseph Haggège, Guillaume Sandou

► **To cite this version:**

Mohamed Derouiche, Soufiene Bouallègue, Joseph Haggège, Guillaume Sandou. Rapid Model Predictive Control Prototyping with LabVIEW/CDSim and CompactRIO Target. 4th International Conference on Control Engineering and Information Technology (CEIT-2016) , Dec 2016, Hammamet, Tunisia. 10.1109/ceit.2016.7929015 . hal-01415057

HAL Id: hal-01415057

<https://centralesupelec.hal.science/hal-01415057v1>

Submitted on 12 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Rapid Model Predictive Control Prototyping with LabVIEW/CDSim and CompactRIO Target

Mohamed L. DEROUICHE^{1,2}, Soufiene BOUALLÈGUE¹, Joseph HAGGÈGE¹ and Guillaume SANDOU²

¹: Laboratoire de Recherche en Automatique (LARA), Ecole Nationale d'Ingénieurs de Tunis (ENIT),
Université Tunis EL MANAR, BP 37, le Belvédère, 1002 Tunis, Tunisia.

²: L2S, CentraleSupélec, CNRS, Université Paris Sud, Université Paris-Saclay,
3 rue Joliot Curie, 91192 Gif-sur-Yvette Cedex, France.

E-mails: mohamedlotfi.derouiche@enit.rnu.tn, soufiene.bouallegue@issig.rnu.tn, joseph.haggege@enit.rnu.tn,
Guillaume.Sandou@centralesupelec.fr

Abstract—In this paper, an advanced Computer Aided Design (CAD) methodology for the Process-In-the-Loop (PIL) co-simulation and rapid prototyping of model predictive controllers is proposed and successfully implemented using the NI CompactRIO-9082 RT target and a host PC. The developed software (SW) and hardware (HW) co-design platform is based on the Control Design and Simulation (CDSim) module of LabVIEW environment and the Network Streams data communication protocol. The designed LabVIEW-based MPC algorithm as well as the dynamic model of the controlled plant are implemented as VI under the cRIO-9082 target and the host PC, respectively. This hardware model will be deployed on the CompactRIO Real-Time (RT) target within a PIL co-simulation framework. The proposed NI CompactRIO-9082 based CAD approach for prototyping and implementation of MPC algorithms is applied to the position control of a Magnetic Levitation system (MAGLEV). All obtained SW/HW co-simulation results are compared and discussed in order to improve the effectiveness of the proposed MPC co-design methodology.

Keywords—*model predictive control, computer aided design, PIL co-simulation, rapid prototyping, LabVIEW/CDSim, NI Compact RIO-9082 platform, MAGLEV system.*

I. INTRODUCTION

Model Predictive Control (MPC) is an advanced design approach that focuses on constructing and optimizing feedback controllers at each discrete-time instant [1]–[4]. Such predictive controllers can adjust the control action before a change in the output setpoint actually occurs. The ability of handling operational constraints in an explicit manner is one of the main reasons for the popularity and successful of MPC approaches in industrial applications especially for electrical motors drives, power systems, chemical process and so on [5], [6].

The implementation of model predictive controllers for complex dynamic systems requires usually non negligible software processing power, resources and hardware materials. The complexity of MPC algorithms (on-line constrained optimization, prediction, estimation, etc.) makes difficult the practical implementation of these digital control laws. Such an implementation becomes increasingly complex with the complexity of the controlled dynamic plants. The textual programming based method of MPC algorithms implementation, such as with C, C++ and VHDL languages, is a tedious and error-prone task [7]–[11]. This may present a real brake to the design of increasingly complex MPC controllers. However,

for rapid prototyping of these predictive control structures purposes, the use of high level development tools such as graphical code generators such as the LabVIEW Control Design and Simulation module (LabVIEW/CDSim) [12] and the hardware Compact Reconfigurable Input/Output (cRIO) platforms [13], [14] is a promising solution for the easy and fast MPC algorithms implementation. The Control Design and Simulation software of LabVIEW, associated to a NI multi-core cRIO RT target and a host PC, presents an efficient tool for Processor-In-the-Loop (PIL) and Hardware-In-the-Loop (HIL) co-simulations and rapid prototyping [15], [16]. These software/hardware (SW/HW) environments and resources will be used together to develop a powerful PIL co-design platform for MPC implementation framework.

The remainder of this paper is organized as follows. In Section II, the model predictive control concept is introduced. A LabVIEW/CDSim-based workflow to design and implement MPC controllers is particularly described. Section III presents the proposed NI cRIO-9082 based CAD methodology for PIL co-simulation and rapid prototyping of MPC controllers. An application to rapid control prototyping of a magnetic levitation system is investigated in Section IV. All SW/HW co-simulation results, obtained in the proposed CAD methodology with LabVIEW/CDSim and cRIO tools, are presented and discussed. Section V concludes this paper.

II. LABVIEW/CDSIM-BASED MODEL PREDICTIVE CONTROL

This section provides information about using the LabVIEW Control Design and Simulation Module to design and implement MPC controllers. Basic concepts about MPC design are firstly presented.

A. Basic concepts of MPC design

As mentioned by J.A. Rossiter in [2], MPC approach reflects human behavior whereby we select control actions which we think will lead to the best predicted outcome over some limited horizon. To make this selection, an internal model of the process in question is used. Decisions are constantly updated as new observations become available. So, an MPC algorithm consists mainly of a cost function to be minimized, operational constraints and model of the controlled plant [1]–[4]. The prediction horizon N_p , used in MPC approach, is the

number of samples in the future during which the feedback controller predicts the plant output. The control horizon N_c is the number of samples within the prediction horizon during which the predictive controller can affect the control action as depicted in Fig. 1. The value specified for this horizon must be less than the one used for the prediction horizon.

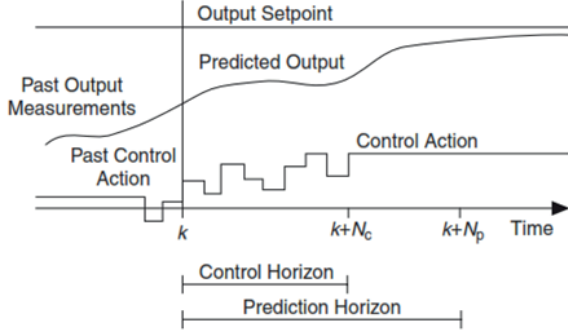


Fig. 1: Prediction and control horizons in MPC framework.

So, the main elements of the discrete-time model predictive control are depicted in Fig. 2. The plant input, the controlled output and the reference trajectory are denoted by $u \in R$, $y \in R$, and $r \in R$, respectively. The plant model is employed to calculate output predictions up to samples of prediction horizon in the future. The optimization algorithm is aimed at determining the control sequence given by $\{u(k-1+i), i=1, 2, \dots, N_c\}$. This sequence minimizes the specified cost function (1) in the MPC design formalism, subject to problem constraints on the input, change in input and output of the plant. The optimization algorithm assumes that $u(k-1+i) = u(k+N_c-1)$ for $N_c < i \leq N_p$.

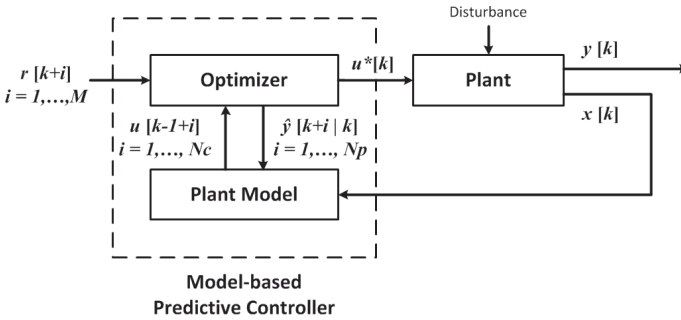


Fig. 2: Model predictive control structure.

For SISO systems, and when omitted the term related on the control action error, the optimization problem, which penalizes tracking errors and control variations and often used in the MPC framework, is given as follows [2]:

$$\min_{\Delta U} J = \sum_{i=1}^{N_p} [r(k+i) - \hat{y}(k+i|k)]^2 + \lambda \sum_{i=1}^{N_c} [\Delta u(k-1+i)]^2 \quad (1)$$

where $\Delta U = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+N_c-1)]^T$ is the vector of decision variables and $\Delta u(k) = u(k) -$

$u(k-1)$. The design parameter $\lambda > 0$, called weighting factor, is adjusted to achieve a compromise between minimizing the output tracking error and minimizing variations on the control signal. In addition, $\hat{y}(k+i|k)$ denotes the prediction of the output at instant $k+i$ on the basis of the measured state $x(k) \in R$.

From an implementation point of view, only the first element of the optimized control sequence is applied to the plant and the control input is updated at each sampling time. The optimization process is repeated at the next sampling instant, on the basis of the measured state vector $x(k)$. In addition to the weight coefficient $\lambda > 0$ (or weighting matrices in the case of MIMO systems) in the cost function, operational constraints on the control action, its rate of change and plant output signals limitations can be defined, respectively, as given by Equations (2), (3) and (4):

$$u_{\min} \leq u(k) \leq u_{\max} \quad (2)$$

$$\Delta u_{\min} \leq \Delta u(k) \leq \Delta u_{\max} \quad (3)$$

$$y_{\min} \leq y(k) \leq y_{\max} \quad (4)$$

B. MPC design with LabVIEW/CDSim

To create an MPC controller under LabVIEW/CDSim environment, the CD Create MPC Controller VI of Fig. 3 is used. This VI bases the MPC controller on a state-space model of the controlled plant.

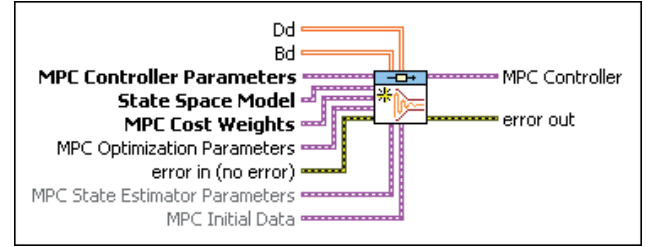


Fig. 3: LabVIEW CD Create MPC Controller VI.

The prediction and control horizons must be provided in the MPC Controller Parameters input of the CD Create MPC Controller VI. These predictive control parameters, as shown in Fig. 1, are fixed for the duration of the execution of the controller. MPC State Estimator Parameters of this VI specifies the parameters of the state estimator that the MPC Controller uses to estimate the states of the plant. This state estimator is defined by using the Noise Covariance method and Kalman filter gain. Plant model states can also be estimated by using the Discrete Observer function outside the MPC controller [12]. The MPC controller, created and implemented in LabVIEW/CDSim, calculates a sequence of future control action values such that the cost function (1) is minimized. The specified weight matrices, through the MPC Cost Weights input of the CD Create MPC Controller VI, adjust the priorities of the control action, rate of change in control action, and plant outputs.

The CD Create MPC Controller VI is also used to specify constraints for the predictive controller design. MPC constraints are specified by either the dual optimization or the

barrier function method. These handling constraints methods are manually selected to use by the polymorphic instances of such a VI. More details about these methods are given in [12]. The constraints for MPC design problem are specified in the MPC Constraints parameter input of the CD Create MPC Controller VI. All these constraints can be updated in run time [12].

Barrier type approach of MPC handling constraints specifies any constraints on the quadratic programming optimization algorithm using the barrier function method. Each parameter specifies a minimum limit, maximum limit, tolerance, or penalty on the initial or final control action $u(k)$, plant output $y(k)$, or rate of change in control action $\Delta u(k)$. When specifying only a minimum or a maximum limit for a parameter, LabVIEW assumes a constant profile for that parameter. Dual MPC constraints method specifies any constraints on the quadratic programming optimization algorithm using the dual optimization method.

Finally, the created MPC controller is now implemented either in a simulation or a real-world scenario while using the CD Implement MPC Controller VI of Fig. 4 within a timed or simulation loops [12]. This VI needs, in addition to the created MPC controller, information about profiles of the input setpoints (Control Action Reference Window) and output setpoints (Output Reference Window), disturbances (Disturbance Window) and mainly the measured output $y(k)$ of the controlled plant, to calculate the control action $u(k)$ to apply to the plant along the control horizon at time k .

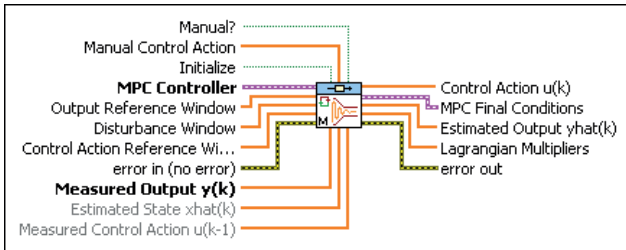


Fig. 4: LabVIEW CD Implement MPC Controller VI.

According to Fig. 4, the CD Implement MPC Controller VI returns mainly the information related to the control action necessary to react to the change in the output setpoint profile, the predicted output of the plant along the prediction horizon and the rate of change in control action. In all cases, it is possible to provide setpoint and disturbance profile information either in advance of controller execution or dynamically during this execution [12].

III. PROPOSED SW/HW CO-SIMULATION APPROACH FOR MPC PROTOTYPING

A. CAD methodology for rapid MPC prototyping

A powerful PIL co-simulation platform for MPC algorithms verification and prototyping must make easy the practical implementation of such control laws with the same hardware target and software tools [15], [16]. For this purpose, the principle of our proposed CAD methodology for rapid

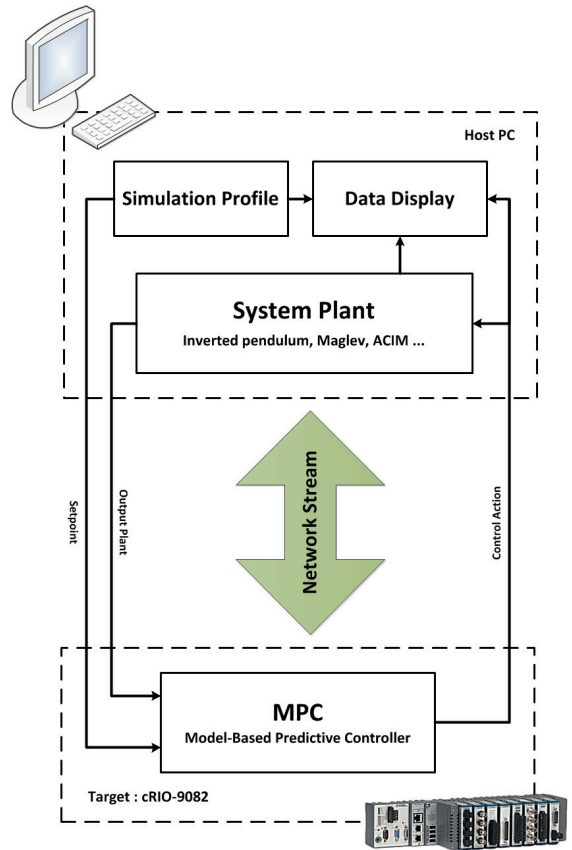


Fig. 5: Block diagram of the cRIO-9082 based PIL co-simulation methodology.

prototyping of MPC algorithms is based on the hardware setup of Fig. 5.

The developed SW/HW solution is based mainly on the use of a multi-core CompactRIO-9082 platform from National Instruments [13], [14]. Well-suited to complex processing and real-time computing, this RT target is associated to a host PC equipped with LabVIEW/CDSim environment. This advanced control platform is connected to the host PC via an Ethernet connection during prototyping phase. The cRIO-9082 device operates autonomously to execute a control LabVIEW project deployed on its RT dual-core processor.

Thanks to its powerful tools, LabVIEW software simplifies construction and prototyping of designed control systems and provides the ability to implement a variety of control algorithms, in particular those of MPC strategy. The integrated Control Design & Simulation Module of LabVIEW presents a powerful tool to implement, prototype and test such algorithms [12].

B. NI Compact RIO-9082 based implementation

The main component of the proposed PIL platform for rapid prototyping of MPC design is the NI Compact RIO-9082 target, Fig. 6. This high-performance multi-core NI cRIO-9082 system provides advanced Intel Core i7 dual-core processor with 1.33 GHz frequency speed, 32 GB nonvolatile storage and 2 GB DDR3 800 MHz RAM [13], [14]. This hardware

target provides also a built-in VGA display output for an integrated user interface and the option to use a Microsoft Windows Embedded Standard 7 (WES7) or LabVIEW Real-Time Operating System (OS).

The NI cRIO-9082 system is built around an 8-slot Spartan-6 LX150 FPGA chassis for custom I/O timing, control, and processing. This chassis contains 1 MXI-Express, 4 USB Hi-Speed, 2 Gigabit Ethernet, and 2 serial ports for connectivity and expansion. An operating temperature range of 0 to 55°C is tolerated for this multi-core RT target.



Fig. 6: Used NI multi-core cRIO-9082 platform.

The NI cRIO-9082 device offers the widest array of connectivity and expansion options available in the NI CompactRIO platform, including the high-bandwidth and low-latency MXI-Express bus for expansion using the multi-slot MXI-Express RIO chassis. The increased processing power of the NI cRIO-9082 target makes it well suited to perform the advanced processing tasks required by complex and hard applications such as rapid control prototyping and HIL co-simulation.

The LabVIEW Real-Time workflow is chosen to take advantage of deterministic execution and the highest degree of reliability in continuous operation environments [14]. However, the WES7 based one takes advantage of the extensive Windows ecosystem of software and display capabilities made possible by LabVIEW software.

IV. APPLICATION TO MPC PROTOTYPING OF A MAGLEV SYSTEM

A. Plant description and modeling

The magnetic levitation system (MAGLEV) of Feedback company [17], which is depicted in Fig. 7, is used as a process example to implement the proposed PIL co-simulation approach.

In required range of operation, the distance h of the suspended sphere is given by the infrared photo-sensor voltage y as follows [17]:

$$y = \gamma h + y_0 \quad (5)$$

where γ is a positive gain depending on the position sensor, and y_0 is the offset voltage such that $y \in [-2V, +2V]$.

The coil current is regulated by an inner control loop within the driver block. Its characteristic is linearly related to the input voltage u as follows, neglecting its high frequency dynamics:

$$i = \rho u + i_0 \quad (6)$$



Fig. 7: Feedback MAGLEV 33-006 system.

where $\rho > 0$ is the coil resistor and $i_0 > 0$ is the offset value of current.

The working excursion of u is limited between $-3V$, corresponding to a null coil current, and $+5V$ that defines the saturation value [17]–[20]. For this process, the predictive control of the suspended sphere position is investigated. The dynamics of the vertical movement of such a sphere is modeled as follows [17], [19], [20]:

$$m \frac{d^2 h}{dt^2} = mg - K \frac{i^2}{h^2} \quad (7)$$

where K is an electromechanical conversion gain depending on the MAGLEV system, m is the mass of the sphere, g is the acceleration of gravity, and i is the coil current.

According to the given sensor and current driver characteristics (5) and (6), the equation (7) can be re-written as:

$$m \frac{d^2 y}{dt^2} = \gamma mg - K \frac{(\rho u + i_0)^2 \gamma^3}{(y - y_0)^2} \quad (8)$$

Taking $x = [y \quad \dot{y}]^T$ as state vector, the following state space representation of the studied system is obtained:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \gamma g - K \frac{(\rho u + i_0)^2 \gamma^3}{m(x_1 - y_0)^2} \end{cases} \quad (9)$$

Finally, while using the values for the physical model parameters, a linear discrete-time model of the MAGLEV plant, where a zero-order-hold with 5 ms sampling period was adopted at the input of the system [19], [20], is given by the following state-space representation:

$$\begin{cases} x_{k+1} = \begin{bmatrix} 1.0108 & 0.0050 \\ 4.3185 & 1.0108 \end{bmatrix} x_k + \begin{bmatrix} -0.0142 \\ -5.6779 \end{bmatrix} u_k \\ y_k = [1 \quad 0] x_k \end{cases} \quad (10)$$

B. Implementation and practical results

In this section, the proposed PIL co-simulation approach for rapid prototyping of the MPC of MAGLEV system is described. All implementation and practical results will be given and discussed.

To implement the proposed PIL co-simulation of MPC design, according to the principle of Fig. 5, two separate VIs are developed. The first one, dedicated to the cRIO-9082 target deployment, is used to configure the MPC controller by introducing the prediction model, input and output constraints and weight parameters of the cost function (see Fig. 8). This VI

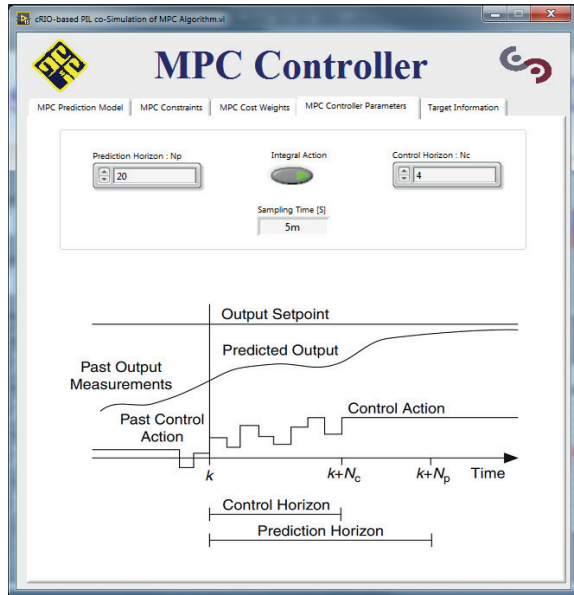


Fig. 8: Developed LabVIEW interface for the Rapid MPC prototyping: cRIO-9082 target deployment VI.

is also used to easy tune the prediction and the control horizons and extract some needed target information. The last tab in this MPC controller VI indicates the establishing communication with the host PC and the successfully downloaded the co-simulation profile. It shows also the target IP address which be used inside the host VI to create a LabVIEW Network Stream data. After setting the IP address, the co-simulation profile is configured by coupling a setpoint to its applying time.

The second VI of the developed PIL platform, as depicted in Fig. 9, is built to display the tuned responses of the cRIO-9082 based controlled system in the host PC. These hardware

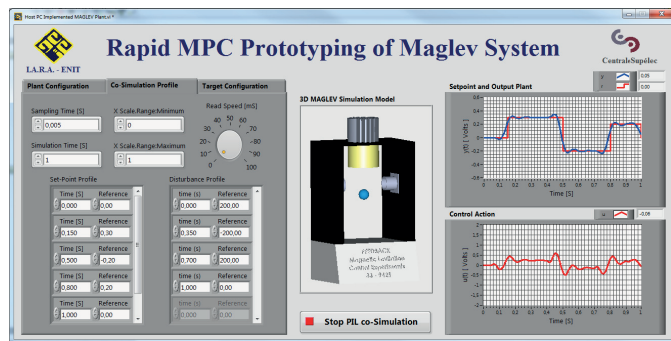


Fig. 9: Developed LabVIEW interface for the Rapid MPC prototyping: plant hardware responses in the host PC.

co-simulation results (plant output and control action) show the effectiveness of the proposed PIL methodology for rapid MPC

prototyping. A 3D model of the MAGLEV system is also built and integrated within this VI in order to show, in the real-time, the suspended sphere motion.

On the other hands, setpoint, plant output and control action signals of the hardware MPC co-design are shown in Fig. 10 and Fig. 11, respectively. All performances of the implemented MPC algorithm for MAGLEV system, in terms of tracking and robustness, are guaranteed. The specified constraints on the control action signal are also respected.

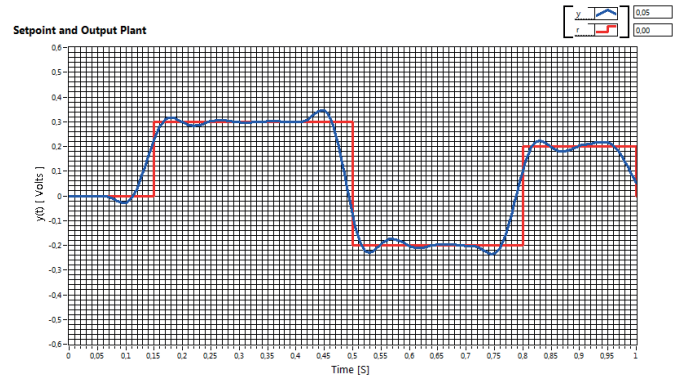


Fig. 10: Controlled position of the MAGLEV suspended sphere.

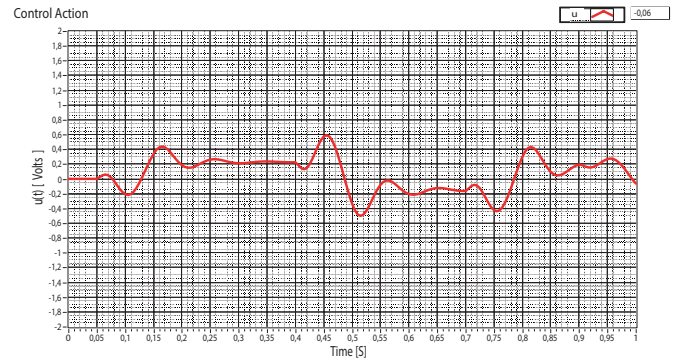


Fig. 11: Control input voltage of the MAGLEV current driver.

In order to highlight the proposed PIL co-simulation ability for choice and tuning of the MPC parameters, Fig. 12 and Fig. 13 show, respectively, the effect of varying the N_c control and N_p prediction horizons parameters on the step response of the controlled system. Values are indicated in the legend of each figure. So, from these figures we find out the usual results of the MPC strategy in terms of trajectory tracking and control law behavior.

In Fig. 13, a very large prediction horizon has a reaction early and thus leads to a stable system, but less fast. However, a too short one causes a control law that reacts late, and with important values leading to the instability of the controlled system. Fig. 12 shows that increasing the control horizon parameter further improves the system response, but at some point they no longer earn anything. The values $N_c = 4, 5$ or 8 give almost the same result. Thus, with a choice of parameters

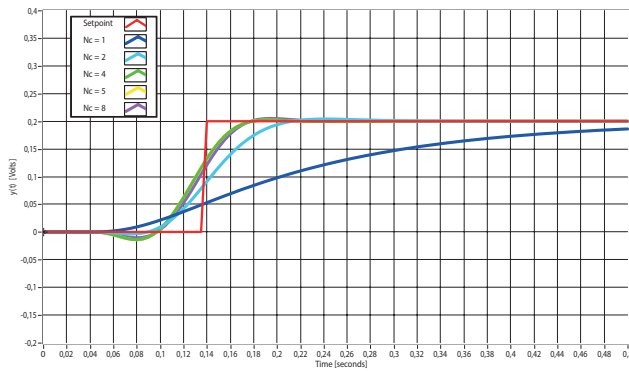


Fig. 12: Effect of varying the control horizon N_c for a fixed prediction horizon ($N_p = 20$).

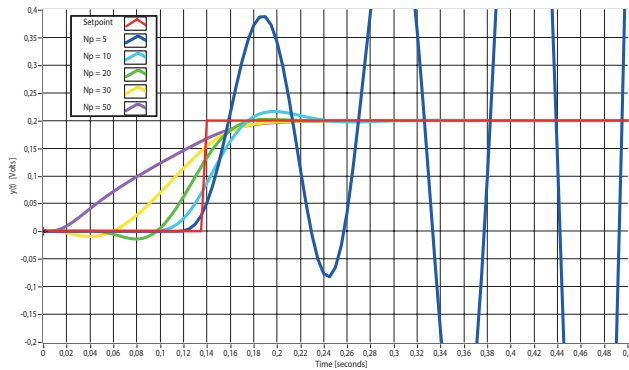


Fig. 13: Effect of varying the prediction horizon N_p for a fixed control horizon ($N_c = 5$).

$N_c = 5$ and $N_p = 20$, the PMC approach allowed to obtain a satisfactory tuning.

Clearly, this proposed LabVIEW/cRIO-based PIL platform also offers, in addition to the validation of the MPC hardware and software implementation materials, a solution for the selection and parameter settings of the MPC algorithms.

V. CONCLUSION

In this paper, a new CAD methodology for PIL co-simulation and rapid prototyping of model predictive controllers have been proposed and successfully applied to the position control of a MAGLEV system. The proposed SW/HW solution is based on the NI cRIO-9082 platform, associated to the LabVIEW/CDSim environment, and a host PC. The obtained results for such a LabVIEW/cRIO-based co-design approach are promising and open to improvement. Using such a CAD methodology for rapid MPC prototyping, users have only to develop the host PC VI for new plant and introduce the new state space model as well as the MPC parameters in the target VI. Forthcoming works deal with Hardware-In-the-Loop (HIL) co-simulation of the MPC strategy and the final implementation of the control laws on the real MAGLEV benchmark available in our laboratory. In addition, the coupling with an optimization method to tune the weighting

matrices to get the best possible performances in accordance with the specifications is also investigated.

REFERENCES

- [1] J.B. Rawlings, D.Q. Mayne, *Model Predictive Control: Theory and Design*, Nob Hill Publishing, Madison, USA, 2013.
- [2] J.A. Rossiter, *Model Based Predictive Control: A practical Approach*, Control Series, CRC Press, New York, 2004.
- [3] D. Bao-Cang, *Modern Predictive Control*, CRC Press, Taylor & Francis Group, USA, 2010.
- [4] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB*, Advances in Industrial Control, Springer-Verlag, London, 2009.
- [5] D.Q. Mayne, Model predictive control: Recent developments and future promise, *Automatica*, vol. 50, pp. 2967–2986, 2014.
- [6] S. Joe Qin, T.A. Badgwell, A survey of industrial model predictive control technology, *Control Engineering Practice*, vol. 11, pp. 733–764, 2003.
- [7] D. Bouafassa, L. Rahmani, and S. Mekhilef, Design and real time implementation of single phase boost power factor correction converter, *ISA Transactions*, vol. 55, pp. 276–274, 2015.
- [8] J. Aymes, *Implementing Adaptive Predictive Control with the TMS320C50 DSP*, EFRIE-France, Texas, 1997.
- [9] B. Mohamed, A. El Mahdy, and M. Refai, Model predictive control using FPGA, *International Journal of Control Theory and Computer Modeling*, vol. 5, no. 2, pp. 1–14, 2015.
- [10] C. Yang, D. Li, J. Zhang, and Y. Xi, Model predictive controller design and implementation on FPGA with application to motor servo system, *Control Engineering Practice*, vol. 20, pp. 1229–1235, 2012.
- [11] S. Bolognani, L. Peretti and M. Zigliotto, Design and Implementation of Model Predictive Control for Electrical Motor Drives, *IEEE Transactions on Industrial Electronics*, vol. 56, no. 6, pp. 1925–1936, 2008.
- [12] National Instruments, *LabVIEW Control Design User Manual*, 371057G-01, www.ni.com, June 2009.
- [13] National Instruments, *NI LabVIEW for Compact RIO Developer's Guide*, www.ni.com, 2013.
- [14] National Instruments, *CompactRIO NI cRIO-9081/9082: Reconfigurable Embedded Chassis with Integrated Intelligent Real-Time Controller for CompactRIO*, www.ni.com, 2011.
- [15] S. Boumazbar, S. Bouallègue, J. Haggège, Co-simulation and Rapid Prototyping of Fuzzy Supervised PID controllers based on FPGA-Nexys2 Board, *7th International Conference on Modeling, Identification and Control ICMIC'2015*, Sousse, Tunisia, December 2015.
- [16] S. Boumazbar, S. Bouallègue, J. Haggège, Co-simulation and Rapid Prototyping of Fuzzy Controllers based on FPGA Nexys2 Board, *2nd International Conference on Automation, Control, Engineering and Computer Science ACECS'2015*, Sousse, Tunisia, March 2015.
- [17] Feedback, *Magnetic Levitation Control Experiments 33-942S*, Feedback Instruments Limited, Version 7.3, 2013.
- [18] T. Bächle, S. Hentzelt and K. Graichen, Nonlinear model predictive control of a magnetic levitation system, *Control Engineering Practice*, vol. 21, pp. 1250–1258, 2013.
- [19] M. Santos, R.K.H. Galvao and T. Yoneyama, Robust Model Predictive Control for a Magnetic Levitation System Employing Linear Matrix Inequalities, *ABCMSymposium Series in Mechatronics*, vol. 4, pp. 147–155, 2010.
- [20] R.C. Fama, R.V. Lopes, A. de Paulo Milhan, R.K.H. Galvao and B.A. Della Lastra, Predictive Control of a Magnetic Levitation System with Explicit Treatment of Operational Constraints, *ABCMSymposium Series in Mechatronics*, vol. 2, pp. 1–8, 2006.